



Learn by doing: less theory, more results

# Moodle 2 for Teaching 4-9 Year Olds

Use Moodle to create quizzes, puzzles, and games  
to enhance the learning ability of your students.

## *Beginner's Guide*

**Nicholas Freear**

**[PACKT]** open source   
PUBLISHING community experience distilled

[www.it-ebooks.info](http://www.it-ebooks.info)

# **Moodle 2 for Teaching 4-9 Year Olds**

## ***Beginner's Guide***

Use Moodle to create quizzes, puzzles, and games to enhance the learning ability of your students

**Nicholas Freear**

**[PACKT]** open source   
PUBLISHING community experience distilled

BIRMINGHAM - MUMBAI



# **Moodle 2 for Teaching 4-9 Year Olds**

## ***Beginner's Guide***

Copyright © 2011 Packt Publishing

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the author, nor Packt Publishing, and its dealers and distributors will be held liable for any damages caused or alleged to be caused directly or indirectly by this book.

Packt Publishing has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, Packt Publishing cannot guarantee the accuracy of this information.

First published: October 2011

Production Reference: 1121011

Published by Packt Publishing Ltd.  
Livery Place  
35 Livery Street  
Birmingham B3 2PB, UK.

ISBN 978-1-84951-328-9

[www.packtpub.com](http://www.packtpub.com)

Cover Image by Rakesh Shejwal ([shejwal.rakesh@gmail.com](mailto:shejwal.rakesh@gmail.com))

# Credits

**Author**

Nicholas Freear

**Project Coordinator**

Michelle Quadros

**Reviewers**

Mary Cooch

Silvina P. Hillar

Maja Kuna

Kent Villard

**Proofreaders**

Aaron Nash

Steve Maguire

**Indexer**

Monica Ajmera Mehta

**Acquisition Editor**

Sarah Cullington

**Graphics**

Geetanjali Sawant

**Development Editor**

Chris Rodrigues

**Production Coordinator**

Alwin Roy

**Technical Editors**

Pallavi Kachare

Arun Nadar

**Cover Work**

Alwin Roy

# About the Author

**Nicholas Freear** got into software and educational technology through a series of happy accidents. During research for a PhD in Mechanical Engineering at the University of Birmingham he was thrown in at the deep end, learning to program in C++, using the Windows API, and programming against the interface card for an early digital camera (a "frame grabber", since you ask).

Bizarrely, this didn't put him off. In his next job, he was a programmer at a high-tech startup company helping to create products from voice recognition and speech synthesis software.

However, the World Wide Web was calling. After a character-building stint as a self-employed developer and accessibility consultant, Nicholas joined the team that was working on The Open University's next-generation e-learning environment. And so, he was introduced to Moodle and the open-source software community. Following several fruitful years, Nick joined the Institute of Educational Technology at The OU, where he got to talk to more academics, pursue his accessibility and usability interests, and work on many different education and research projects.

He blogs (<http://freear.org.uk>), contributes to the Moodle community (<http://moodle.org/user/view.php?id=93815>), and likes to talk at workshops and conferences despite his stammer.

When he's not trying to understand the mysteries of the Web, Nick likes to sing, cycle, listen to loud music, and learn about all things Chinese. Occasionally all at the same time.



# Acknowledgement

I am indebted to many people for their help during this project.

To Snehman Kohli and Sarah Cullington at Packt Publishing for coming to me with the original ideas—it's been a crazy ride!

To my coordinator, editors, and reviewers: Michelle Quadros, Sarah, Chris Rodrigues, Reshma Sundaresan, Swapna Verlekar, Arun Nadar, Pallavi Kachare, Maja Kuna, Silvina P. Hillar, Kent Villard, Rebecca Galley, and Mary Cooch – for outstanding patience.

To the Moodle community, particularly the contributed plugins curator, Anthony Borrow, Michael de Raadt, Mary Cooch, and those on the Quiz forum—for encouragement and valuable feedback on my contributed plugins.

The VLE team in Learning and Teaching Solutions at The Open University—for stimulating and nurturing my interest in free/open source software and Moodle. You guys and gals continue to do a fantastic and sometimes under-rated job.

Colleagues at the Institute of Educational Technology and around The Open University—for support, friendship, and putting interesting projects my way. This includes Will, Juliette, Chetz, Rod, and the Richards.

I owe a debt of gratitude to my parents David and Doreen for their support, and firing my interest in books.

Lastly, to the free/open source software and open-content communities: "If I have seen further it is by standing on the shoulders of giants"

— *(Isaac Newton)*

# About the Reviewers

**Silvina P. Hillar** has been teaching English since 1993. She has always had a great interest in teaching, writing, and composing techniques, and has made a lot of research on this subject.

She is an English Teacher, a Certified Legal Translator (English/Spanish), and has a Post Degree in Education (graduated with Honors).

She has been working in several schools and institutes with native English-speaking students and as an independent consultant for many international companies as an interpreter, translator, and e-learning activities developer.

She has always had a passion for technological devices concerning education. Former videos and cassettes were a must in her teaching lessons; computer was and still is present. Her brother Gastón C. Hillar designed some programs and games for her teaching. Lately, she is teaching using Moodle and the Web 2.0. She believes that one of the most amazing challenges in education is bridging the gap between classic education and modern technologies.

She has been doing a lot of research on multimedia assets which enhance the teaching and learning through VLE platforms. She tries to embed the learning of students through new resources which are appealing and innovative for them. Thus, multimedia stimulates the different thinking skills as well as multiple intelligences.

She has worked on books such as *Moodle 1.9 English Teacher's Cookbook* and *Moodle 2.0 Multimedia Cookbook*.

---

I would like to thank all the team at Packt Publishing Ltd.

I owe tremendous thanks to my wonderful six-year-old son, Nico, who despite his age has been very patient and supporting in all my projects.

---

**Maja Kuna** has been providing consultancy about technical writing, instructional design, e-learning content development, and management to corporate and public organizations for the last few years. She has consulted and worked for organizations in different verticals, including: EUMETSAT, Societa' degli Studi di Settore (SOSE), GiuntiLabs, and Piramide e-learning supporting various e-learning projects.

Maja is a certified Moodle teacher (MCCC, formerly MTC) and she has been involved as an instructor in courses on Moodle in Almere (Netherlands), Florence (Italy), and Darmstadt (Germany).

Maja holds a Master of Art in Polish Philology with specialization in teaching (Jagiellonian University in Cracow) and a postgraduate diploma in E-learning and Knowledge Management (University of Florence). She speaks English, Italian, and Polish fluently.

**Kent Villard** is a twenty-year veteran of the IT industry and currently the E-Learning Coordinator for the University of Prince Edward Island. He has been administering Moodle for almost five years. He has acted as a technical reviewer for Packt on *Moodle 1.9 for Design and Technology* and *Moodle 1.9 Teaching Techniques* among other titles.

When not administering Moodle or evangelizing the Mac platform, Kent can be found spending quality time with his beautiful wife Denise and awesome kids, Maxwell and Samantha, and watching really bad movies.

Kent lives in Cornwall, Prince Edward Island in Atlantic Canada. He can be reached at [kent.villard@gmail.com](mailto:kent.villard@gmail.com).



# www.PacktPub.com

## Support files, eBooks, discount offers and more

You might want to visit [www.PacktPub.com](http://www.PacktPub.com) for support files and downloads related to your book.

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at [www.PacktPub.com](http://www.PacktPub.com) and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at [service@packtpub.com](mailto:service@packtpub.com) for more details.

At [www.PacktPub.com](http://www.PacktPub.com), you can also read a collection of free technical articles, sign up for a range of free newsletters and receive exclusive discounts and offers on Packt books and eBooks.



<http://PacktLib.PacktPub.com>

Do you need instant solutions to your IT questions? PacktLib is Packt's online digital book library. Here, you can access, read and search across Packt's entire library of books.

## Why Subscribe?

- ◆ Fully searchable across every book published by Packt
- ◆ Copy and paste, print and bookmark content
- ◆ On demand and accessible via web browser

## Free Access for Packt account holders

If you have an account with Packt at [www.PacktPub.com](http://www.PacktPub.com), you can use this to access PacktLib today and view nine entirely free books. Simply use your login credentials for immediate access.







*To Guangyu, for believing I could do this before I did,  
Without your support and love this wouldn't have been possible.  
And to Rosie, I wrote this book for you.*



# Table of Contents

|   |           |
|---|-----------|
| <b>Preface</b>                                      | <b>1</b>  |
| <b>Chapter 1: Getting Started</b>                   | <b>9</b>  |
| Introducing Moodle                                  | 10        |
| Time for action – logging in for the first time     | 10        |
| Creating your first Moodle course                   | 13        |
| Time for action – creating a course                 | 14        |
| Exploring roles                                     | 16        |
| Creating a quiz                                     | 18        |
| Time for action – creating a quiz activity          | 20        |
| Inserting an image                                  | 24        |
| Time for action – inserting an image                | 24        |
| Installing a text filter                            | 27        |
| Time for action – installing the SimpleSpeak filter | 27        |
| Using the filter                                    | 29        |
| Time for action – using the SimpleSpeak filter      | 29        |
| Troubleshooting                                     | 31        |
| Incorporating the quiz in your teaching             | 32        |
| Summary   | 34        |
| <b>Chapter 2: Basic Math in Moodle</b>              | <b>35</b> |
| Searching for open content                          | 36        |
| Time for action – searching for open content        | 36        |
| Creating a label                                    | 39        |
| Time for action – creating a label resource         | 39        |
| Embedding a video                                   | 43        |
| Time for action – embedding video                   | 43        |
| Creating numerical questions                        | 46        |
| Time for action – creating numerical questions      | 46        |
| Creating more math questions                        | 48        |



|  |           |
|--|-----------|
| Time for action – creating more maths questions          | 49        |
| A calculated question                                    | 52        |
| Time for action – creating a calculated question         | 52        |
| Installing the Calculated Objects question type          | 55        |
| Time for action – installing a question type             | 55        |
| Creating your first calculated objects question          | 56        |
| Time for action – creating a calculated objects question | 56        |
| Summary  | 61        |
| <b>Chapter 3: Telling Stories</b>                        | <b>63</b> |
| Creating a database activity                             | 64        |
| Time for action – creating a database                    | 66        |
| Planning our database fields                             | 68        |
| Time for action – creating fields                        | 69        |
| Editing templates  | 71        |
| Time for action – editing templates                      | 72        |
| Adding an entry  | 73        |
| Time for action – adding an entry                        | 74        |
| Commenting on the database entry                         | 76        |
| Time for action – adding a story as a comment            | 76        |
| Installing a sound recorder                              | 77        |
| Audacity and LAME  | 78        |
| Time for action – installing Audacity and LAME           | 78        |
| Testing audio  | 80        |
| Time for action – performing a test recording            | 80        |
| Recording audio  | 81        |
| Time for action – recording audio                        | 82        |
| Integrating an audio file with Moodle                    | 83        |
| Time for action – enabling the multimedia filter         | 83        |
| Uploading audio files                                    | 84        |
| Time for action – uploading files                        | 85        |
| Embedding our audio file                                 | 88        |
| Time for action – embedding audio                        | 88        |
| Running your lesson                                      | 90        |
| Summary  | 91        |
| <b>Chapter 4: Spot the Difference</b>                    | <b>93</b> |
| Introducing the lesson module - SVG used as is!          | 94        |
| Creating our first lesson activity                       | 95        |
| Time for action – adding a lesson activity               | 95        |
| Finding spot-the-difference images                       | 99        |
| Time for action – finding images online on OCAL          | 99        |

---

|   |            |
|---|------------|
| Installing an image editor                            | 101        |
| Time for action – installing Inkscape                 | 102        |
| Editing the images                                    | 103        |
| Time for action – editing the first SVG image         | 104        |
| Bringing it together                                  | 109        |
| Time for action – adding images to our lesson         | 109        |
| More difficult exercises                              | 112        |
| Time for action – adding more questions               | 113        |
| Wrapping up the lesson                                | 116        |
| Time for action – adding a branch                     | 116        |
| Enrolling students                                    | 118        |
| Configuring an enrolment key                          | 118        |
| Time for action – creating and using an enrolment key | 118        |
| Enrolling via user upload                             | 120        |
| Time for action – uploading users                     | 120        |
| Summary   | 123        |
| <b>Chapter 5: Setting Homework</b>                    | <b>125</b> |
| Introducing the SIMILE timeline widget                | 126        |
| Installing a text editor                              | 127        |
| Time for action – installing Notepad2                 | 128        |
| Creating the timeline data file                       | 131        |
| Time for action – creating the XML                    | 131        |
| Troubleshooting                                       | 133        |
| Installing the Timeline Widget filter                 | 134        |
| Time for action – installing the filter               | 134        |
| Creating a follow-on activity                         | 136        |
| Time for action – adding a forum                      | 136        |
| Adding information to our XML file                    | 138        |
| Time for action – editing the XML                     | 139        |
| Embedding the timeline                                | 142        |
| Time for action – putting a timeline together         | 142        |
| Integrating the timeline with teaching                | 146        |
| Time for action – creating an assignment              | 146        |
| Summary   | 151        |
| <b>Chapter 6: Fun Games</b>                           | <b>153</b> |
| Introducing Games                                     | 153        |
| Creating a glossary                                   | 154        |
| Time for action – creating a glossary                 | 154        |
| Adding glossary entries                               | 158        |
| Time for action – adding glossary entries             | 158        |

|  |            |
|--|------------|
| Glossary auto-linking                            | 162        |
| Time for action – using auto-linking             | 162        |
| Installing the Games module                      | 163        |
| Time for action – installing the module          | 163        |
| Creating a game                                  | 166        |
| Time for action – adding snakes and ladders      | 166        |
| Adding a cryptex                                 | 170        |
| Time for action – adding a word search puzzle    | 171        |
| Hidden picture                                   | 174        |
| Time for action – adding a hidden picture        | 174        |
| Hidden picture questions                         | 177        |
| Time for action – adding questions               | 177        |
| Creating the puzzle                              | 179        |
| Time for action – adding a hidden picture game   | 179        |
| Summary  | 183        |
| <b>Chapter 7: Interactive Puzzles</b>            | <b>185</b> |
| An alternative word search                       | 186        |
| Time for action – planning the word search       | 186        |
| Creating the word search XML                     | 188        |
| Time for action – creating the XML               | 188        |
| Uploading the puzzle XML                         | 190        |
| Time for action – uploading the XML file         | 191        |
| Creating a word search puzzle                    | 193        |
| Time for action – embedding Flash                | 193        |
| Online word search generator                     | 197        |
| Time for action – using an online generator      | 197        |
| A jigsaw puzzle                                  | 199        |
| Time for action – uploading a jigsaw picture     | 199        |
| Creating the online jigsaw                       | 201        |
| Time for action – creating the jigsaw            | 201        |
| Summary  | 208        |
| <b>Chapter 8: Stories Revisited</b>              | <b>209</b> |
| Finding a book                                   | 210        |
| Time for action – finding a book                 | 210        |
| Installing the Book module                       | 212        |
| Time for action – installing the activity module | 212        |
| Creating a book                                  | 214        |
| Time for action – creating a book                | 215        |
| Improving our book                               | 219        |
| Time for action – adding custom styles           | 219        |

---

|  |            |
|--|------------|
| Adding pictures  | 223        |
| Time for action – adding pictures to our book            | 223        |
| Completing our online book                               | 225        |
| Time for action – adding the remaining parts to the book | 225        |
| Installing a dictionary                                  | 228        |
| Time for action – installing a pop-up dictionary         | 228        |
| Integrating a dictionary service                         | 230        |
| Time for action – using the dictionary                   | 230        |
| Summary  | 233        |
| <b>Chapter 9: Embedding the Web</b>                      | <b>235</b> |
| What is Scratch?   | 236        |
| Embedding Scratch projects                               | 236        |
| Time for action – embedding Scratch applets              | 236        |
| The Scratch embed filter                                 | 240        |
| Time for action – installing and using the filter        | 240        |
| RSS feeds and blocks                                     | 243        |
| Time for action – adding an RSS feed                     | 244        |
| Improving our RSS feed                                   | 246        |
| Time for action – using Yahoo! Pipes                     | 246        |
| Feeds  | 248        |
| Science simulations                                      | 249        |
| Time for action – incorporating PhET simulations         | 250        |
| HTML5 jigsaw   | 256        |
| Time for action – exploring the jigsaw                   | 256        |
| Creating your own jigsaw activity                        | 259        |
| Time for action – using an iframe                        | 260        |
| Summary  | 264        |
| <b>Chapter 10: Administration</b>                        | <b>265</b> |
| Course backup  | 265        |
| Time for action – creating a course backup               | 266        |
| Restoring a course                                       | 272        |
| Time for action – course restore                         | 273        |
| Recording and tracking progress                          | 276        |
| Time for action – making notes                           | 276        |
| Blogging   | 279        |
| Time for action – writing a blog post                    | 279        |
| Tags   | 283        |
| Gradebook  | 284        |
| Time for action – viewing grades                         | 284        |
| Community  | 286        |

|  |            |
|--|------------|
| <b>Contributed plugins</b>                           | <b>286</b> |
| <b>Summary</b>                                       | <b>287</b> |
| <b>Appendix A: Accessibility for Online Teaching</b> | <b>289</b> |
| <b>What is accessibility?</b>                        | <b>290</b> |
| <b>Why accessibility for teachers?</b>               | <b>291</b> |
| <b>Guidance for teachers and online authors</b>      | <b>292</b> |
| Perceivable  | 292        |
| Operable   | 296        |
| Usable   | 297        |
| Robust   | 299        |
| <b>Useful links</b>                                  | <b>299</b> |
| <b>Final words</b>                                   | <b>300</b> |
| <b>Appendix B: Pop Quiz Answers</b>                  | <b>301</b> |
| <b>Chapter 1</b>                                     | <b>301</b> |
| Getting Started                                      | 301        |
| <b>Chapter 2</b>                                     | <b>301</b> |
| Basic Math in Moodle                                 | 301        |
| <b>Chapter 3</b>                                     | <b>302</b> |
| Telling Stories                                      | 302        |
| <b>Chapter 4</b>                                     | <b>302</b> |
| Spot the Difference                                  | 302        |
| <b>Chapter 5</b>                                     | <b>302</b> |
| Setting Homework                                     | 302        |
| <b>Chapter 6</b>                                     | <b>303</b> |
| Fun Games  | 303        |
| <b>Chapter 7</b>                                     | <b>303</b> |
| Interactive Puzzles                                  | 303        |
| <b>Chapter 8</b>                                     | <b>303</b> |
| Stories Revisited                                    | 303        |
| <b>Chapter 9</b>                                     | <b>304</b> |
| Embedding the Web                                    | 304        |
| <b>Chapter 10</b>                                    | <b>304</b> |
| Administration                                       | 304        |
| <b>Index</b>   | <b>237</b> |

---

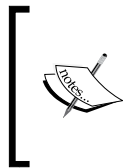
# Preface

Moodle is a virtual learning environment that is being used in more and more schools worldwide. It is ideal for teaching a younger age group as interactive lessons enable children to learn quicker and with greater ease.

*Moodle 2 for Teaching 4-9 Year Olds Beginner's Guide* will help you to adapt your existing lesson plans to online Moodle courses and will give you ideas to create new activities, quizzes, and puzzles to make the learning process fun and interactive for young children.

The interactivity of Moodle means that it is perfect for teaching younger children as they can learn by watching, listening, and doing. Learn how to create activities and quizzes that are specially adapted for younger children and are quick and easy for you to incorporate in Moodle. Other highlights include spot-the-difference exercises, games, and embedded puzzles.

Teaching young children has just got easier with the help of Moodle to create fun, interactive, and informative learning activities.



The website for the book can be found at:

<http://freear.org.uk/moodle>

The site contains links to downloads and a forum for discussions with other readers, demonstrations, updates and errata.

Note that links to individual downloads are listed in the book at the point at which they are needed. There is no single code archive.

## What this book covers

*Chapter 1, Getting Started*; after a brief introduction we step through logging into Moodle. Then we create a course and an alphabet quiz activity based around multiple-choice questions. We add in the third-party SimpleSpeak plugin, and use it to voice sounds and words for the quiz.

*Chapter 2, Basic Math in Moodle* talks about how the e-learning environment is a great home for basic and more advanced numeracy exercises to cater for a mixed ability class. We set up a math quiz using built-in question types and the contributed Calculated Objects plugin. We use Creative Commons search and learn to embed video.

*Chapter 3, Telling Stories* lets us harness the creativity and storytelling abilities of children. We create an activity with the Database module to allow the class to collect pictures. Then we set an activity where the students write a story online, inspired by their own or a classmate's picture. We also explore how to record an audio book.

*Chapter 4, Spot the Difference*, here we will learn how visual activities are a fun way to stimulate your class. We will also learn how to use an open-source, desktop image editor to modify images to form the basis of our exercise. Then we employ the built-in Lesson activity to present simple and trickier spot the difference puzzles.

*Chapter 5, Setting Homework*, here we create a visual, interactive history timeline using MIT SIMILE and linked to a Moodle Forum. And we learn about using the Assignment module for homework activities.

*Chapter 6, Fun Games*, here we will use the third-party Game module by Vasilis Daloukas to set up snakes and ladders, and hidden picture puzzles. We integrate the built-in Glossary plugin.

*Chapter 7, Interactive Puzzles*, here we integrate various open source Flash games and puzzles from Subtangent including word-search and a jigsaw.

*Chapter 8, Stories Revisited*, in this chapter we source an ex-copyright children's book from Project Gutenberg, and employ Petr Škoda's Book module to integrate it in Moodle. And we incorporate an external dictionary service.

*Chapter 9, Embedding the Web*; we incorporate various resources into our course, including activities programmed using MIT Scratch, RSS feeds, PhET science simulations, and an HTML5 jigsaw.

*Chapter 10, Administration*, here we will look at course and activity backup and restore in Moodle—for your peace of mind. We explore the Gradebook, student notes, and the Moodle community.



*Appendix A, Accessibility for Online Teaching*, here we wrap up by discussing the importance of making online learning resources accessible to those with disabilities. After a brief look at the legal framework, we illustrate the fundamental principles with practical examples. We end with links to further reading.

## What you need for this book

In order to use the activities that are presented in this book you will need:

- ◆ Access to the Internet (fairly obvious), and for some chapters the permissions to install software on your desktop or laptop computer. The software is available for recent versions of Windows (Windows XP, Vista and 7 at the time of writing), Mac OS X and Linux,
- ◆ For a few chapters you need a sound card in your computer (almost all modern computers have one), headphones or speakers and a microphone,
- ◆ A working installation of Moodle\* (installing Moodle is outside the scope of the book).
- ◆ You need an account in Moodle with the **course creator** role,
- ◆ And you need a friendly system administrator or IT support person, who is prepared to install some third-party or contributed plugins for Moodle on your behalf. (Or, if you have the master administrator account, usually called **admin**, you will be able to install plugins yourself - there are instructions throughout the book.)
  - If you only have access to a 'locked-down' installation of Moodle, with IT support people who are unwilling to install third-party plugins you will be able to complete roughly half the activities in the book.

I note above that you need a working installation of Moodle. For teaching this will need to be on a server that is accessible from the Web, so that anyone with an account can log in.

However, to learn and explore the capabilities of Moodle it can be installed on your local Windows or Mac OS X. The options are:

- ◆ Download and install a pre-built package based on Moodle and XAMPP, for Windows (<http://download.moodle.org/windows/>) or Mac OS X (<http://download.moodle.org/macosx/>). This requires basic computer knowledge.
- ◆ Install XAMPP (<http://www.apachefriends.org/en/xampp.html>) and Moodle separately (<http://download.moodle.org/>). (Note, MAMP is another solution in place of XAMPP on Mac OS X, <http://www.mamp.info/>).

## A note on versions

The activities in this book have been written to function in versions 2.0.x (2.0.0, 2.0.1...) and 1.9.x of Moodle. There are a few exceptions, which are noted in the text. In the first seven chapters, instructions are given for Moodle 2 then Moodle 1.9. For the remaining chapters the procedure is given for Moodle 2, and it is assumed that you will have learnt enough to extrapolate for version 1.9.

The activities will probably also work in Moodle 2.1.x and most should work in version 1.8.x. However, at the time of writing this has not been tested and I can't offer any guarantees (changes to this situation will be noted on the website for the book, <http://freear.org.uk/moodle>).

## Who this book is for

This book is aimed primarily at teachers of children aged 4 to 9 who wish to integrate the Moodle open-source virtual learning environment (VLE) into their teaching, and need an introduction to creating courses and activities in Moodle. It assumes a basic knowledge of information technology (for example, using office software like Microsoft Word and Excel), the Internet and the Web (for example, email and search using Google) and no prior knowledge of Moodle. The activities are designed with a blended learning approach in mind (that is, a combination of online and face-to-face). However, many could be readily adapted to other contexts.

In addition a number of the activities can be used as the basis for the teaching of older students.

## Conventions

In this book, you will find several headings appearing frequently.

To give clear instructions of how to complete a procedure or task, we use:

### **Time for action – heading**

- 1.** Action 1
- 2.** Action 2
- 3.** Action 3

Instructions often need some extra explanation so that they make sense, so they are followed with:

### ***What just happened?***

This heading explains the working of tasks or instructions that you have just completed.

You will also find some other learning aids in the book, including:

### **Pop quiz – heading**

These are short multiple choice questions intended to help you test your own understanding.

### **Have a go hero – heading**

These set practical challenges and give you ideas for experimenting with what you have learned.

You will also find a number of styles of text that distinguish between different kinds of information. Here are some examples of these styles, and an explanation of their meaning.

Code words in text are shown as follows: "The `[em]` and `[/em]` square-bracket tags denote the start and end of some emphasized text—often rendered in italics".

A block of code is set as follows:

```
[Speak] Well done! [/Speak]
[Speak] Did you know? An [em]x-ray[/em] is often used to show the
      bones in the body. [/Speak]
```

When we wish to draw your attention to a particular part of a code block, the relevant lines or items are set in bold:

```
<h2> A a </h2>
<p> ah </p>
<p>
</p>
<p> apple </p>
<p> Which of the words below start with the letter " a "? </p>
```

**New terms** and **important words** are shown in bold. Words that you see on the screen, in menus or dialog boxes for example, appear in the text like this: "Press the **Update** button to close the **Edit HTML Source** dialog. Then scroll to the end of the form and press the button to **Save and display**".



Warnings or important notes appear in a box like this.



Tips and tricks appear like this.

## Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book—what you liked or may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply send an e-mail to [feedback@packtpub.com](mailto:feedback@packtpub.com), and mention the book title via the subject of your message.

If there is a book that you need and would like to see us publish, please send us a note in the **SUGGEST A TITLE** form on [www.packtpub.com](http://www.packtpub.com) or e-mail [suggest@packtpub.com](mailto:suggest@packtpub.com).

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, see our author guide on [www.packtpub.com/authors](http://www.packtpub.com/authors).

## Customer support

Now that you are the proud owner of a Packt book, we have a number of things to help you to get the most from your purchase.

## Downloading the example code

You can download the example code files for all Packt books you have purchased from your account at <http://www.PacktPub.com>. If you purchased this book elsewhere, you can visit <http://www.PacktPub.com/support> and register to have the files e-mailed directly to you.

## **Errata**

Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you find a mistake in one of our books—maybe a mistake in the text or the code—we would be grateful if you would report this to us. By doing so, you can save other readers from frustration and help us improve subsequent versions of this book. If you find any errata, please report them by visiting <http://www.packtpub.com/support>, selecting your book, clicking on the **errata submission form** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata will be uploaded on our website, or added to any list of existing errata, under the Errata section of that title. Any existing errata can be viewed by selecting your title from <http://www.packtpub.com/support>.

## **Piracy**

Piracy of copyright material on the Internet is an ongoing problem across all media. At Packt, we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works, in any form, on the Internet, please provide us with the location address or website name immediately so that we can pursue a remedy.

Please contact us at [copyright@packtpub.com](mailto:copyright@packtpub.com) with a link to the suspected pirated material.

We appreciate your help in protecting our authors, and our ability to bring you valuable content.

## **Questions**

You can contact us at [questions@packtpub.com](mailto:questions@packtpub.com) if you are having a problem with any aspect of the book, and we will do our best to address it.



# 1

## Getting Started

*In this chapter we will start to explore Moodle, add our first course, and create our first learning activity, a quiz to help our class learn the alphabet. We will look at ways to enrich the activity with images and sound. Finally, we will discuss how to create a blended lesson incorporating the online activity.*

In this chapter we will:

- ◆ Log in to a Moodle website
- ◆ Familiarize ourselves with the Moodle system
- ◆ Introduce Moodle course formats
- ◆ Create our first course and select a format
- ◆ Learn about resource and activity modules
- ◆ Create our first activity—a quiz
- ◆ Find out about question types and add questions to the quiz
- ◆ Start to write content and search online for images
- ◆ Install the Audio Filter module
- ◆ Bring it together—format the quiz to create audio prompts
- ◆ Integrate the Moodle Quiz activity in your teaching



This book is primarily written for Moodle 2, though the activities will work with Moodle 1.9 too. Where instructions vary the Moodle 2 instructions will generally come first.

So let's get on with it...

## Introducing Moodle

Moodle is a **virtual learning environment or learning management system (VLE / LMS)**, a type of website where you as a teacher can log in and create courses and activities for your pupils, and assign grades for quizzes, and other assessed activities. It is also a place where your pupils can collaborate online, for instance through a forum or a wiki. Your school may have its own installation of Moodle, or the schools in your area may share Moodle systems maintained by a commercial partner.

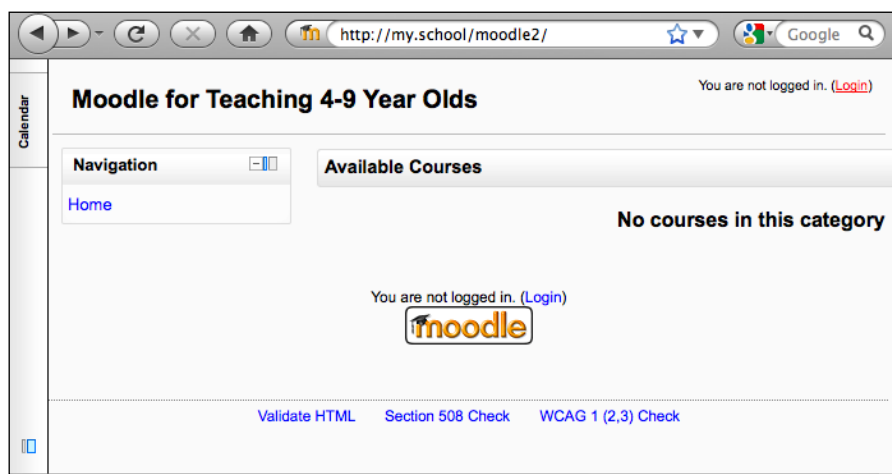
For this chapter and the ones that follow, you will need access to a pre-installed Moodle website, and a user name, and password for your account on the Moodle system. The account should give you the role of **course creator**, which will allow you to create and teach in Moodle courses. You will also need a computer with access to the Internet, speakers or headphones, and a friendly IT support person to install third-party modules on your Moodle system.

Without further ado we will launch into our first practical activity.

### Time for action – logging in for the first time

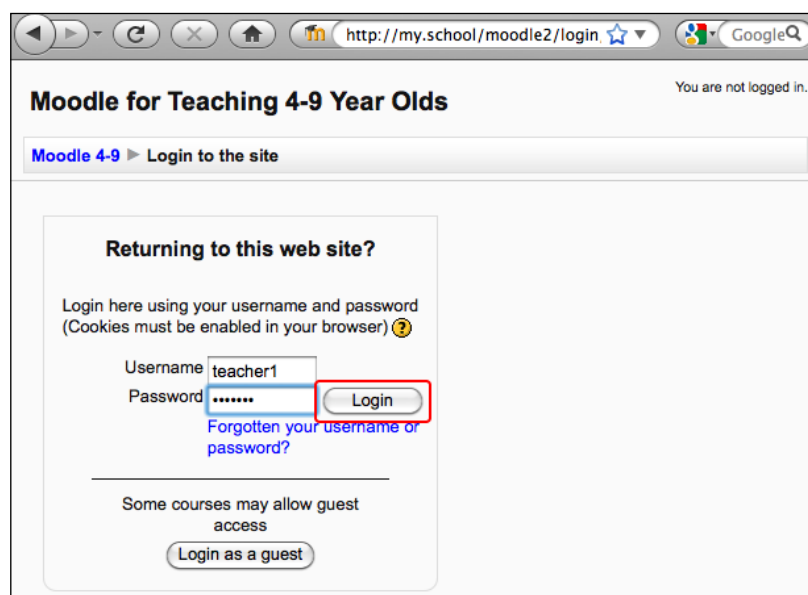
Follow these steps to log in to and explore Moodle:

1. You will need to log in to the website in order to create activities for your students. Please go to the address of your school's Moodle website now in your browser.

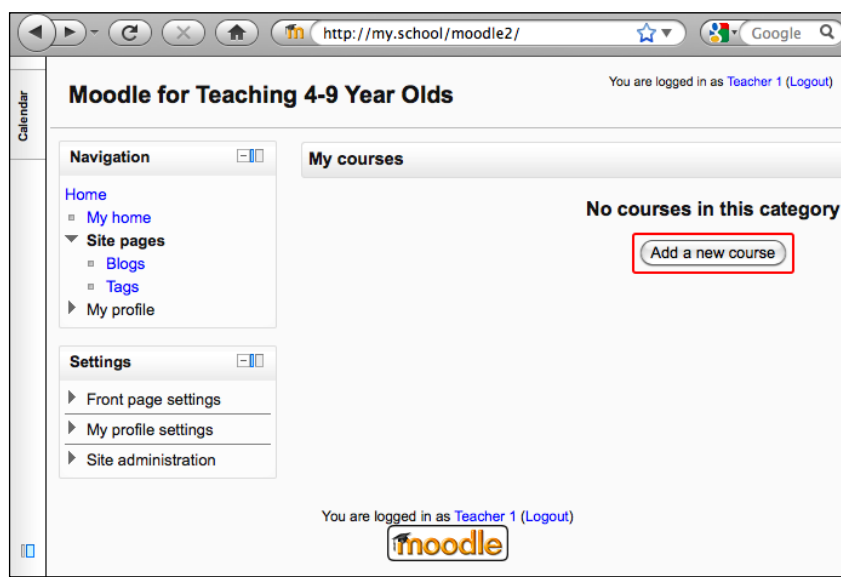




2. You will probably see a page that looks something like the next screenshot.
3. Click on the **Login** link in the top right corner. If there is no **Login** link (because of the way the web site is configured), simply add `login/` to the end of the address in your browser's address bar, for example `http://your.school/moodle/login/`—you can add this link as a bookmark in your browser.
4. You will be taken to a page like the one shown in the next screenshot. On the left is a login form. If you are using Moodle 1.9, there will be various instructions related to creating accounts on the right. Enter your username and password as shown and press the **Login** button:



5. If this is your first log in and the username and password are correct you may be taken to a page and asked to change your password. You will be taken back to the home page, which will now look something like the following screenshot. If there was a problem with your login you will see an error message on the page and for this you should talk to your IT support person.

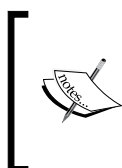


## ***What just happened?***

We employed the user account created by our IT support person and logged in to our local Moodle installation. As you can see in the previous figure, because we are logged in, the content of the home page has changed. What you find will depend on the version of Moodle.

Moodle 2 pages can contain three columns, though only two are shown in the previous screenshot. The main content, which for this page is a list of **My courses**—currently empty, is in the centre. On the left are side-blocks titled **Navigation** and **Settings**. On the far left is a **docked Calendar** block, with the text, **Calendar**, rotated 90 degrees. This block would initially be to the right of the centre block, occupying the third column. Any side-block in Moodle 2 can be **docked** (if docking is enabled by the site administrator, which it is by default).

In Moodle 1, including version 1.9, things are a little more straightforward, though less flexible. There are three columns (or two, depending on how the site is configured). Again the main content, an empty list of **My courses**, occupies the centre. The default side-blocks you would see when you first log in as a teacher are **Site Administration** to the left and **Calendar** to the right. And the side-blocks cannot be docked.



From now on in this book the descriptions and screenshots will concentrate on Moodle 2 for simplicity, with occasional references to Moodle 1.9 in brackets or as tips (asides). In this way, you will be able to work out what to do in Moodle 1.9.

A link to your user-profile and a logout link are in the top right of the page. If you do not see the **Site administration** item within the **Settings** block, or a Site administration side-block on the home page, your account may not have sufficient permissions to complete the activities in this chapter. Please ask your IT support person to make you a **Course creator**.

Expand the link **Courses** in the Site administration block, and click on **Add/edit courses**. This link takes you to the courses index page, which shows a list of course categories. There may only be one category, **Miscellaneous**. You can have as many or as few categories and sub-categories as you like in Moodle. Courses can be moved between categories at any time so don't worry unduly about categories at this stage.

We have checked that you have the sufficient privileges for the next step. Let's press on and create a course.

## Creating your first Moodle course

Now we are ready to create our first course. A course is a way of grouping activities and resources. You can then give a cohort of pupils or students access to the activities and resources. There are administrative and student-management tools associated with a course.

## Time for action – creating a course

These are the steps to follow to create a course in Moodle:

1. On the categories page click the **Add a new course** button. You will be presented with a large form, the top half of which is shown in the following screenshot. This form may be daunting at first, but once we have stepped through the important parts it will make sense. The top three fields are **Category**, **Full name**, and **Short name**.

Moodle 4-9 ► Administration ► Course categories ► Add a new course

**Navigation**

- Home
- My home
- Site pages
- My profile

**Settings**

- Category: Miscellaneous
  - Turn editing on
  - Filters
- My profile settings
- Site administration

**Edit course settings**

**General**

Category: Miscellaneous

Course full name: My first course

Course short name: MY101

Course ID number:

Course summary: This is my first course - I'll be trying out an alphabet quiz.

Path: p

Format: Weekly format

Number of weeks/topics: SCORM format, Social format, Topics format, Weekly format

Course start date: 2010

Hidden sections: Hidden sections are shown in collapsed form

News items to show: 5

Show gradebook to students: Yes

2. You can leave the category as **Miscellaneous** enter a full name such as **My First Course** and a short name, for example **MY101**. As you travel further with Moodle you may want to agree with your colleagues how to use the short name systematically. You should also enter a meaningful summary. At this stage don't worry about the buttons above the summary text area.

3. Now choose the course **Format** from the drop-down menu—there are currently six course formats built in to Moodle:
  - ☐ The first one, **SCORM** (and LAMS in Moodle 1.9) can be ignored for now.
  - ☐ The **Social** course format is useful for things like a virtual staff "room" or notice board. This leaves us with the **Topics**, **Weekly**, and **Weekly – CSS/No tables** formats.
  - ☐ The **Topics** format is ideal when you want to group items into topics, but don't have a fixed timetable in mind.
  - ☐ The **Weekly** format should be default when you do have a specific timetable (in Moodle 1.9 use the **Weekly—CSS/No table** format, which is an improvement of the **Weekly** format, for accessibility to those with disabilities).

For our first course we'll use the **Topics** format. You can then choose the number of weeks/topics—I have set five. This can be altered later. Continue, following these steps:

4. You can leave the defaults for the other fields in the **General** section of the form.
5. Moving past **Enrolments**, **Enrolment expiry notification**, and **Groups** we come to **Availability**. If this installation of Moodle is already used in classes in your school you should set availability to **This course is not available to students** for now. This will effectively put your course in an unpublished mode, where it is not visible to pupils (it will be visible to your colleagues).
6. Scroll down to the end of the form and press **Save changes**.

Congratulations, you have created your first course in Moodle!

## ***What just happened?***

We created a course by entering a title and summary, and choosing a course format. And we hopefully found that this large form is not as complicated as it first seemed. We started exploring our new course.

In Moodle 2, after the final step above you will be taken to the course main page for your new empty course.

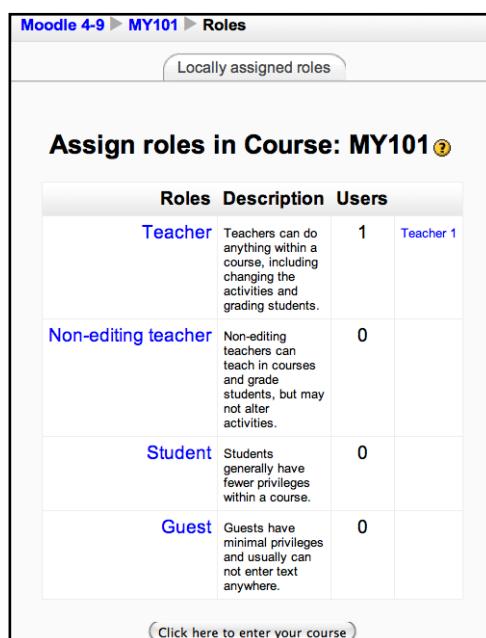
In Moodle 1.9, you will be taken to an intermediate page titled **Assign roles in Course: MY101**. At this point it is worth looking briefly at **roles**, which are common to Moodle 1.9 and 2. Then we will find out how to add course activities and content.

## Exploring roles

Roles are the system Moodle uses to grant different types of user permissions—what they can and cannot do, in different contexts or parts of the site. The context in this case is your new course, and there are four default roles in this context:

- ◆ **Teachers** can do anything in a course
- ◆ **Non-editing teachers** can teach and assign grades but not edit the course or activities
- ◆ **Students** as the name suggests can participate in a course, for example, by commenting in activities or answering quizzes
- ◆ A **Guest** may be able to read but not write in the course

Another significant context in Moodle is the site as a whole. This has the additional roles of **Course creator** (you probably have this role in the site context) and **Administrator**, a role used by IT support personnel.



Moodle 4.9 ► MY101 ► Roles

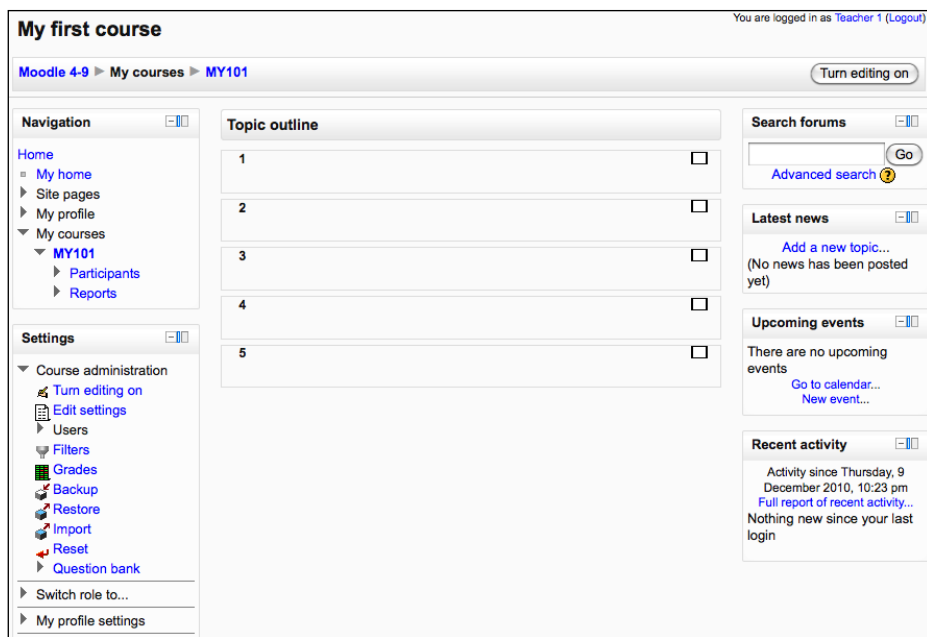
Locally assigned roles

**Assign roles in Course: MY101** ?

| Roles                               | Description   | Users                       |
|-------------------------------------|---|-----------------------------|
| <a href="#">Teacher</a>             | Teachers can do anything within a course, including changing the activities and grading students. | 1 <a href="#">Teacher 1</a> |
| <a href="#">Non-editing teacher</a> | Non-editing teachers can teach in courses and grade students, but may not alter activities.       | 0                           |
| <a href="#">Student</a>             | Students generally have fewer privileges within a course.   | 0                           |
| <a href="#">Guest</a>               | Guests have minimal privileges and usually can not enter text anywhere.                           | 0                           |

[Click here to enter your course](#)

In the figure we just saw from Moodle 1.9, you should see a single user, yourself, assigned the **Teacher** role for the course. This is sufficient while you develop and use the course. Press the button labeled **Click here to enter your course**.



As you can see in the previous screenshot, the course is displayed with a three-column layout. An introductory section with a **News forum** and five empty topics are down the center of the screen. **Navigation** and **Settings** blocks are on the left, while **Search forums**, **Latest News**, **Upcoming Events**, and **Recent Activity** blocks are on the right of the page. (In Moodle 1.9 you will probably see the **Participants**, **Search forums**, and **Administration** blocks on the left.) The **Settings** block in Moodle 2 contains a number of links—for example, **Course administration** | **Edit settings** allows you to modify the options you set when you created the course.

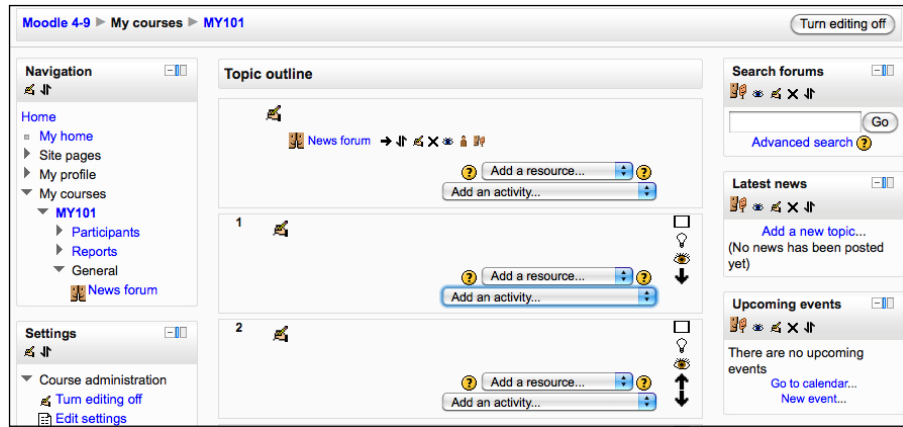


There are two useful items to note. At the top right of the page is a button labeled **Turn editing on**. This is a safety feature—during a lesson turn editing off, so that you don't mistakenly delete an activity. In Moodle 2 there is a **Switch role to...** item in the **Settings** block, below **Course administration**. This allows you to see what the course looks like if you are a pupil for example. In Moodle 1.9 there is a **Switch role to...** drop-down menu next to the **Turn editing on** button.

Press the **Turn editing on** button now.

- ◆ As the figure below shows, when editing is turned on, icons appear under the title of each side block, and for each topic in the course outline
- ◆ The arrow icons allow you to move or re-order the side-block or course topics and the **eye** icon allows you to temporarily hide a block or topic

- ◆ A **Blocks** side block appears in the bottom right of the page
- ◆ **Add a resource...** and **Add an activity...** drop-down menus appear within each topic

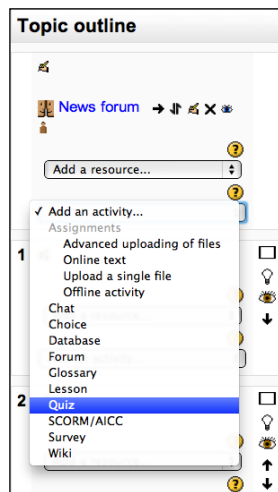


Now that we have a bare course, it is time to add an activity. We start with a quiz.

## Creating a quiz

In the last section we created our first course. Now it's time to add the first activity, a quiz to help our pupils learn the alphabet. Note that while a quiz is frequently used for assessment we will start by concentrating on it as a creative tool for learning.

In the next figure we focus on the **Add an activity** drop-down menu for topic "1"—we will look at the resources later.





The activities available by default in Moodle are:

- ◆ **Assignments:** There are four options:
  - ❑ **Single file:** Students upload a single file, for example multimedia or a Word document.
  - ❑ **Advanced uploading :** Students can optionally upload multiple files, comment next to their submissions, and teachers can provide their feedback as an uploaded file.
  - ❑ **Online text:** Students can type text directly into Moodle, with teachers providing inline feedback.
  - ❑ **Offline activity:** Teachers enter a summary and due date for an assignment separate from Moodle. The grade and comments can be recorded in Moodle.
- ◆ **Chat:** A real-time (synchronous) online text conversation, somewhat similar to Yahoo! Messenger or Skype for instance.
- ◆ **Choice:** Poll your students with a simple single-question, multiple choice quiz.
- ◆ **Database:** Design a database with arbitrary fields or columns, that you and your class can add entries to, search and view. Example uses of database include collecting nature observations, photographs, and books.
- ◆ **Forum:** A tool for pupils to discuss a topic.
- ◆ **Glossary:** A list of word definitions.
- ◆ **Lesson:** A flexible activity containing pages that can be presented in a linear order, or in an order dependent on the student's responses to questions.
- ◆ **Quiz:** An activity with questions designed and set by the teacher, which may be assessed. Questions can be of different types, for example, free text and multiple-choice.
- ◆ **SCORM/AICC:** An activity allowing the teacher to import re-usable learning objects that conform to the SCORM standards. **SCORM** stands for **Sharable Content Object Reference Model**.
- ◆ **Survey:** This module allows you to present one of the several pre-defined surveys about students' experiences of online learning.
- ◆ **Wiki:** A tool for pupils to collaboratively create documents and resources.
- ◆ **Workshop:** New in Moodle 2, this activity enables students to be graded on their work, and on an assessment of their peers' work.



As is the case in many Moodle forms, there is a help link next to the add-activity menu (by default, a question mark icon). Press it and a popup window containing a brief summary of the activity types will be displayed.

## Time for action – creating a quiz activity

We are going to use a **Quiz** as a fun, interactive means of learning the alphabet. Here's how:

1. In your new course, select **Quiz** from the drop-down menu and you will be taken to the form to add a quiz.
2. The only required field in the form is the quiz name; enter something like **Alphabet Quiz**.
3. As demonstrated in the figure below, you may want to enter an **Introduction** containing instructions for your pupils (as this activity will probably be used in class the written instructions will reinforce your verbal instructions).
4. Leave the defaults for the **Open** and **Close the quiz** fields, **Time limit**, **Attempts allowed (Unlimited)**, and **Grading method (Highest grade)**.
5. Ensure that the **New page** field is set to **Every question**. In Moodle 1.9 under **Display**, change **Questions per page** from **Unlimited** to **1**.
6. Under **Question behaviour**, press the button on the right to **Show Advanced** options. Keep **Adaptive mode** set to **Yes**, and set the **Apply penalties** field to **No**.
7. Leave the remaining sections as you find them; scroll down and press **Save and display** at the bottom of the form. As with all Moodle forms, you will be able to edit this one later should you wish.

✓ Adding a new Quiz to Topic 1 ?

**General**

Name\* Alphabet quiz

Introduction

Font family Font size Paragraph

**B** *I* U ABC x<sub>2</sub> x<sup>3</sup>

☰ ☷ ☹ ☲ ☳ ☴ ☵ ☶ ☸ ☹ ☲ ☳ ☴ ☵ ☶ ☸

☰ ☷ ☹ ☲ ☳ ☴ ☵ ☶ ☸ ☹ ☲ ☳ ☴ ☵ ☶ ☸

This is a fun quiz to help you learn letters and words.

Try to answer the question below each word.

Path: p

HTML format

Open the quiz 8 August 2010 20 35 ☐ Enable

Close the quiz 12 December 2010 20 35 ☐ Enable

Time limit ? 0 minutes ☐ Enable

Attempts allowed Unlimited

Grading method ? Highest grade

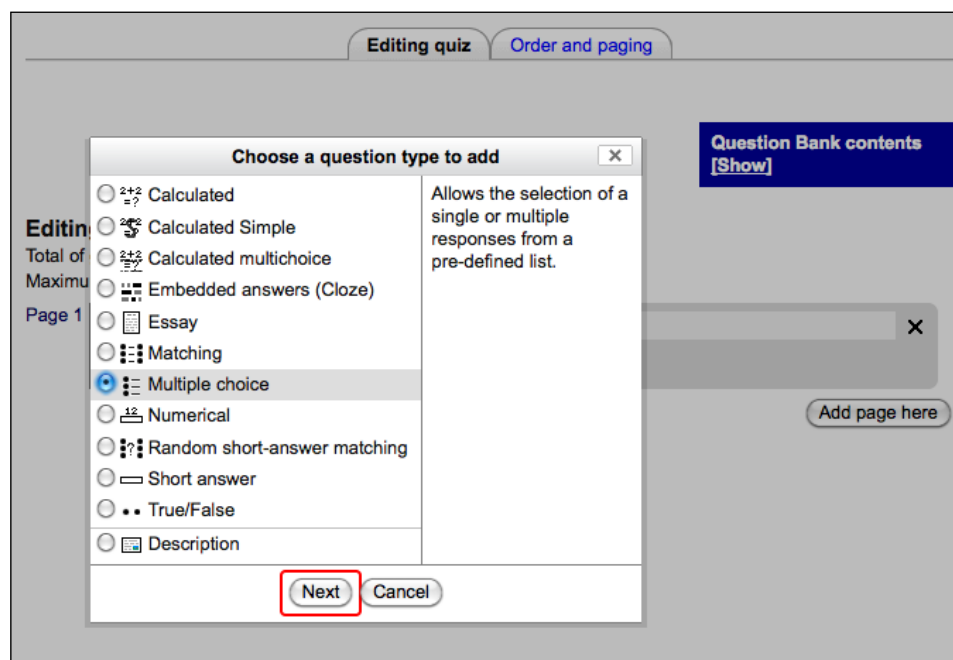
**Layout**

Question order As shown on the edit screen

New page ? Every question

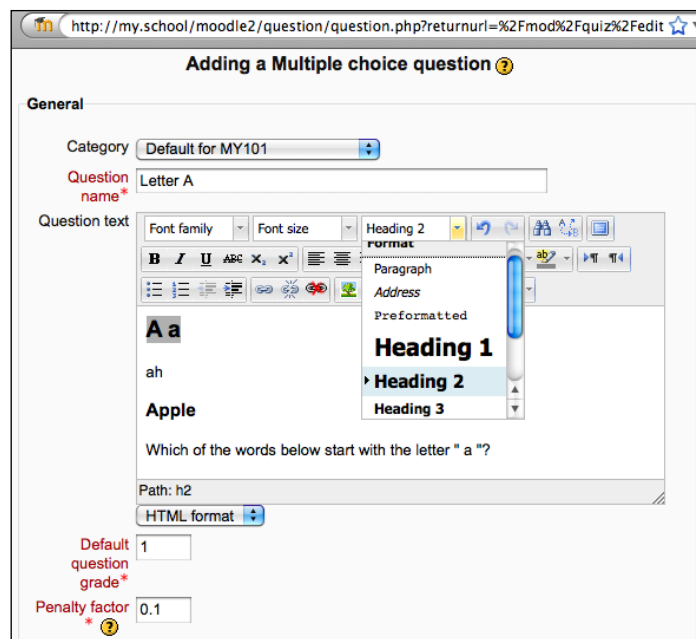
Quizzes are containers for Moodle questions—separate objects that can be reused, and what we have so far is an empty container. It's time to add some questions.

1. In Moodle 2, press the **Add a question...** button to display the dialog shown in the next screenshot. In Moodle 1.9 you will see a drop-down menu labeled **Create a new question**. Many types of questions can be created; we will use **Multiple choice** to reinforce the learning in our alphabet quiz.



2. Select **Multiple choice**, press **Next** (only in Moodle 2), and you will be taken to a new page and presented with the form shown below. You can set the category to **Default for Alphabet Quiz**, enter a name of **Letter A**, and we are ready to create our first rich content. Don't panic!

The text field under the **Question text** label is a rich or **What-You-See-Is-What-You-Get (WYSIWYG)** editor, similar to a word processor like Microsoft Word or Open Office. It consists of a text area and two or three rows of controls and buttons to set the text structure and formatting; for example, a drop-down menu to set a heading or normal text, buttons to turn text into a hyperlink, insert an image, and so on. In Moodle 1.9, you will only see the rich-editor if you use the Internet Explorer or Firefox web browsers. In other browsers, for example Safari and Google Chrome, it is replaced with a simple text-area.



To format our text (more correctly, to structure it), follow these steps:

1. In the rich-editor, first type the letters **A a** (uppercase A, lowercase a). Press *Enter* for a new line, type the approximation for the sound **ah**, press enter again, and type an example word; I used **apple**.
2. Press *Enter* for a final line break and type a question (this is a quiz after all)—**Which of the words below start with the letter 'a'?**
3. It is always useful to add headings to give your content and the page structure. Select the first line **A a** with your mouse and choose **Heading 2** in the third drop-down menu from the left.

### ***What just happened?***

In this section we added our first activity, a quiz, to our course. We discovered that activities have some common properties like a name and introductory text. There are often properties that are specific to that type of activity; for example, how to order the questions in a quiz.

We found that a quiz can contain many different types of questions, and we went on to add a multiple-choice question to our empty quiz.

Finally, we explored the use of headings in the rich-editor to structure our text. You will not normally want to use **Heading 1** (the largest), as this will be added automatically by Moodle near the top of the page, so you should generally start the hierarchy with a level 2 heading. Remember, we are adding structure, without worrying at this stage how the different elements will be styled or formatted—we'll address styling later.

We have a basic question, so now is the time to find some images to enhance the activity.

## Inserting an image

Now, you will probably want to add a picture to liven up the page. There are a number of places to search for **Open Content**—text, images, video, and so on, that you can use and re-use under license. One example, shown in the next figure, is **Wikimedia Commons**, part of the Wikipedia family of sites (<http://commons.wikimedia.org>):



### Time for action – inserting an image

Go to Wikimedia Commons now and search for **apple**. The page for apple contains various images; click on one that you like and the link will take you to an about page for the image. Right-click on the image and choose **Copy image location** from the context menu (Firefox browser). When you paste you will see a link, something like the following:

`http://upload.wikimedia.org/wikipedia/commons/f/fb/Red\_Delicious.jpg`

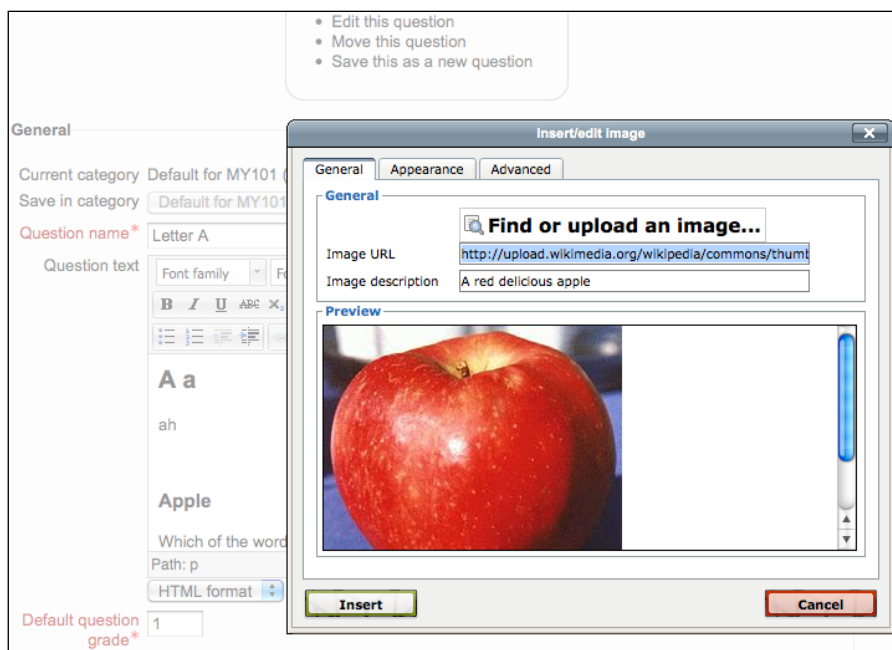


We can generate a link to a smaller image on **Wikimedia** by adding `/thumb` in the middle and a suffix like `/NNNpx-NAME.EXTENSION`, where **px** is the short form for the unit pixels. For example, to create an image that is 240 pixels wide (a reasonable size for an image to illustrate our quiz) enter the following in the address bar of your browser:

`http://upload.wikimedia.org/wikipedia/commons/thumb/f/fb/Red_Delicious.jpg/240px-Red_Delicious.jpg`.

These are the steps to add an image to our question:

1. In the Moodle editor, put your mouse cursor after the **ah** and press *Enter* to insert a new line.
2. Press the **Insert Image** button, on the bottom row of buttons. A new window will appear with the title **Insert/edit image** as shown in the next screenshot.
3. Paste the location you copied from Wikimedia Commons in the **Image URL** form-field, and add a short meaningful phrase for the **Image description / Alternate text**. I typed, **A red delicious apple**.
4. Press the **Insert** button; the window will close and you will see the image you chose in the Moodle editor.



At this stage it is instructive to view the **Hyper-Text Markup Language (HTML)** that has been created for you by the editor. Press the **Edit HTML source** button (labeled **<>** in Moodle 1.9). Most of the editor buttons are disabled, and you can see **tags** as they are called inside the angle brackets, for example `<h2>` which starts a level two heading, `</h2>` which ends it, and `<br />` for line breaks. Below, I have highlighted the `<img>` tag, which contains the address or location of our image (and inserted line breaks and truncated the `src` attribute for clarity):

```
<h2> A a </h2>
<p> ah </p>
<p>
</p>
<p> apple </p>
<p> Which of the words below start with the letter " a "? </p>
```



#### Downloading the example code

You can download the example code files for all Packt books you have purchased from your account at <http://www.PacktPub.com>. If you purchased this book elsewhere, you can visit <http://www.PacktPub.com/support> and register to have the files e-mailed directly to you.

It should be noted that by using this method, the image itself remains on the Wikimedia server—it is not copied to your Moodle system. Later, we will see an alternative way of doing things. We have made a useful digression into content editing for the **Question text**, but now we return to the job in hand—our alphabet quiz.

1. Continuing down the form, leave the **Default question grade** at **1** and adjust the **Penalty factor** to **0** (from **0.1**)—as this is not a serious assessment, we probably don't want to penalize the pupils for getting the answer wrong the first time.
2. Leave **General feedback** empty and for **Number the choices?** choose **No numbering**.
3. Now we will give our pupils two possible answers. For **Choice 1** enter a word beginning with **A**, for example **ant**, set the grade to 100 percent, and enter some feedback—I have put **Well done!** For **Choice 2** I put an incorrect answer of **bee**, a grade of 5 percent, and some feedback—**Woops, that's wrong. Please go back and try again.**
4. That is it. Scroll down to the foot of the page and press the button **Save changes**. You will be returned to the page titled **Editing Quiz**, with the new question **Letter A** appearing in the question bank on the right.



5. On the left under **Questions in this quiz**, the message will read **No questions have been added yet**. Simply tick the box against the question **Letter A** in the bottom right and press the button **Add to quiz**. The question has been added and now appears on the left.

### ***What just happened?***

We searched for an image on **Wikimedia Commons** then inserted it into our Moodle question. We viewed the **Hyper-Text Markup Language (HTML)** code that the editor created for us. And then we completed the form to create the question.

We can preview the result of our work by finding and clicking the **Preview** link in the tab bar near the top of the quiz-editing interface.

Now you could repeat the process for the other 25 letters of the alphabet—each time, saving the **Letter A** template with a new name. Simple! But first, we should take a minute. You may have spotted the flaw in what we have done so far. No? Well consider this. Any pupils who would find the alphabet quiz useful will probably not be able to simply read all the example words, like **apple**, **banana**, and so on, or follow the feedback. So, what can we do to improve matters?

## **Installing a text filter**

Adding text-to-speech functionality is one effective way to make our alphabet quiz more usable. We can achieve this courtesy of what is termed a Moodle **filter**, some computer code that transforms textual content that follows a particular pattern, and enhances it in some manner. Such a filter could use an external text-to-speech service, for instance **Google Translate** (<http://translate.google.com/#en|en|Hello+world!>—Look for the **loudspeaker** icon). Don't worry if this makes little sense yet. Our example should make things clearer.

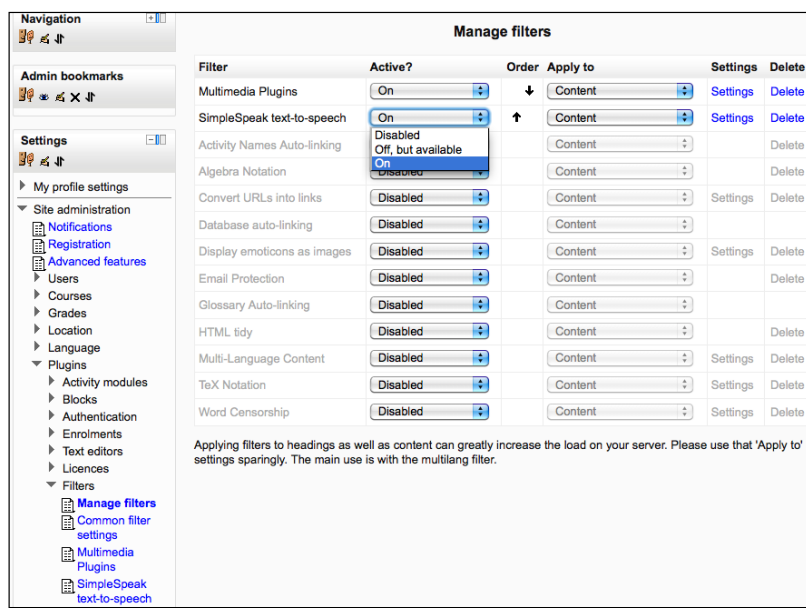
### **Time for action – installing the SimpleSpeak filter**

Unless you have **administrator** privileges on your Moodle installation, you will need the help of a friendly IT support person for this task. The installation follows these steps:

1. Visit the entry for the **SimpleSpeak** filter in the Moodle Plugins database, [http://moodle.org/plugins/view.php?plugin=filter\\_simplespeak](http://moodle.org/plugins/view.php?plugin=filter_simplespeak).
2. Review the documentation, and download the latest version of the **SimpleSpeak filter** code.
3. Unzip the code somewhere convenient, then use FTP, SSH software, or similar

to copy the directory renamed to `simplespeak` to the `filter` directory on the server. For example, you may end up with `/var/www/moodle/filter/simplespeak`.

4. The site administrator can now log in to Moodle and go to the home page. In the **Settings** block press **Site administration** and click on the **Plugins** link (**Modules** in Moodle 1.9). It will expand to show links for different types of plugins including **Filters**.
5. Click on the **Filters** link, and follow the link to **Manage filters**.
6. **SimpleSpeak text-to-speech** should be visible in the list of disabled (grayed) filters near the bottom of the page. Choose **On** in place of **Disabled** in the first drop-down menu. (In Moodle 1.9, press the closed-eye icon next to the filter name—it will become an open-eye icon to indicate that it has been enabled.) Refer to the following screenshot.
7. Click on the **Settings** link on the same row and to the right of **SimpleSpeak**. In the form which appears enter the following for the **Text to speech service URL**, `http://translate.google.com/translate_tts?q=!TEXT`. Press the **Save changes** button.



The administrator or IT support person can now hand back to you.

## What just happened?

We used the Modules and Plugins database at Moodle.org (<http://moodle.org/modules>). We or our IT support found the SimpleSpeak text-to-speech filter, which is released under the same GNU General Public License as Moodle. It was downloaded, installed, and then configured.

Note that the SimpleSpeak filter caches MP3 audio files produced by external services locally on disk, to reduce the load on those services and to improve performance.

Without further ado we will use our new filter plugin.

## Using the filter

There are two ways of making the **SimpleSpeak** filter work in Moodle—one is entering raw HTML, which for our alphabet questions may be a bit complicated. Or we can use simpler square-bracket syntax. We'll start with the latter.


### Time for action – using the SimpleSpeak filter

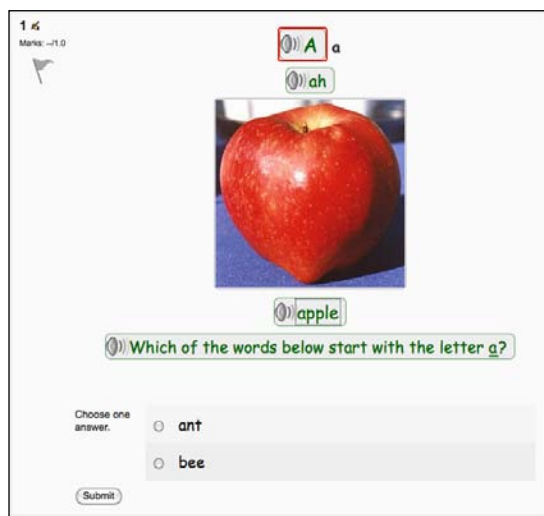
Carry out the following steps to use SimpleSpeak:

1. Go back to Moodle in your browser, and press the **My First Course** (MY101) link.
2. Press the **Alphabet Quiz** link and choose the **Edit** tab.
3. Under **Questions in this quiz** press the edit ("**hand**") icon against the **Letter A** question.
4. Scroll down to the question text. Use the rich-editor and replace the existing content with the following block of text:

```
[Speak]
; Just a comment.
letter = A
sound  = ah
image  = http://upload.wikimedia.org/wikipedia/commons/thumb/f/fb/
Red_Delicious.jpg/240px-Red_Delicious.jpg
alt    = A red delicious apple
word   = apple
phrase = Which of the words below start with the letter [em]a[/
em]?
[/Speak]
```

What does this all mean?

- ◆ The `[Speak]` and `[/Speak]` square-bracket tags indicate the start and end of some content we wish to speech-enable.
  - ◆ The next line starting `;` is a comment by (and for) the author—it will be ignored by Moodle and it won't be visible to students.
  - ◆ The remaining lines have the syntax **keyword equals value**.
  - ◆ `letter = A` will create a button labeled **A**, which when pressed will produce the audio **Say the letter, A**.
  - ◆ Similarly, `sound = ah` and `word = apple` will create buttons to speak **Say the sound, ah** and **Say the word, apple**.
  - ◆ `phrase = Which of the words ...?` creates a button to speak the given phrase as is.
  - ◆ The `[em]` and `[/em]` square-bracket tags denote the start and end of some emphasized text—often rendered in italics.
  - ◆ Finally, `image = http://...` and `alt = A red...` create an embedded image with some alternative text. There should be no line-breaks in `image = http://....`
5. When you have entered the text in the editor scrolled to the bottom of the page, press **Save changes** again. Then click the **Preview** tab near the top of the page and you should see something like the figure below. The screenshot contains the image of a red delicious apple from Wikimedia Commons.
6.  Try pressing the buttons with sound-icons and after a short pause you should hear speech.



## What just happened?

We re-visited the alphabet question that we created previously. We used a square-bracket notation that was transformed by the SimpleSpeak filter into clickable buttons. Pressing these buttons resulted in synthesized speech.

There may be a slight time-lag before the speech is audible. The length of the pause depends on the speed of your Internet connection, among other factors.

## Troubleshooting

If you don't see something like the previous screenshot there are a number of things to check:

- ◆ If you don't see the image, check the URL (`http://...`) you entered, and test your Internet connection.
- ◆ If you don't see the speak buttons, go back to editing the question. Are there any spaces between the square brackets and the keyword **Speak**? For example, the closing `[ / Speak ]`. Remove the spaces, save your question, and preview again.
- ◆ Still problems? Go back to the editor and press the **Edit/Toggle HTML source (<>)** button. You should see the square brackets, key = value and line break tags, for example:
 

```
[Speak] <br /> ; A comment. <br /> letter = A <br /> sound = ah
<br /> ... [/Speak]
```

  - ❑ In the **Toggle source** view; if you don't see line-break tags `<br />`, then insert them as required. (In this view the text will appear all on one line as above—don't worry!)
  - ❑ If there are extra line-break tags, for example `image = http://upload.wikimedia.org/ <br /> ... -Red_Delicious.jpg`, then remove them.
  - ❑ If you see any other hair-pin braces between the `[Speak]` and `[/Speak]` tags, for instance `<span style="font-style: italic;">some text</span>`, remove them but keep the text, for example **some text**. Save the question and preview again.
- ◆ If you don't hear speech when pressing a button with a sound-icon icon, ask your friendly IT support person to go to the filter settings page—they can check the speech-synthesis service.
- ◆ Ask your IT support person to check that the filter is correctly installed. You may also like to ask a question on the support forums of Moodle.org (<http://moodle.org>).

So there we have it. An interactive alphabet quiz, which provides speech prompts. How can this quiz be combined with face-to-face teaching?

## Incorporating the quiz in your teaching

Depending on your resources, you can split your class into pairs or groups of three or four, demonstrate how the quiz works, then allow the groups to practice individually. Each group will need a computer with speakers or headphones. As with many of the activities described in this book, smaller groups and individual learners can feel the benefits of self-paced learning. This is also an activity where you can ask pupils to practice with their parents.

Your class will learn important ICT skills, including following a process or workflow, and hand-eye coordination through using a mouse. And they will improve their verbal skills, and practice letter, sound, and word association.

The activity can be extended to allow the pupils to search for their own images to "complete" the quiz.



### Accessibility

Throughout the book we will be highlighting the benefits of the activities we create and how they are presented, to those with differing abilities and disabilities. The speech-enabled quiz in this chapter is "multi-modal". That is, it can be experienced through multiple senses, in this case, through sight and hearing. This is a benefit to, for example, pupils with dyslexia. The speech buttons can be operated via the keyboard as well as the mouse, for those with less manual dexterity.

Learn more about Web accessibility for online learning in *Appendix A, Accessibility for Online Teaching*.

## Have a go hero

In the previous sections we used the **SimpleSpeak** filter to enrich the questions in our alphabet quiz. Where else in the quiz activity could you use the filter to create speech buttons?

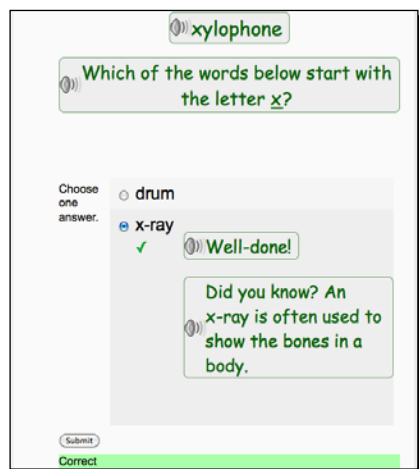
**Hint:** you may be able to use the simpler `[Speak] some text [/Speak]` syntax.

**Possible answer 1:** You could add the speak tags `[Speak] ... [/Speak]` to the feedback for each question. The feedback could also contain interesting facts. For example for **X**, a correct answer may be **X-ray**, and the feedback could be:

```
[Speak] Well done! [/Speak]
```

```
[Speak] Did you know? An [em]x-ray[/em] is often used to show the  
bones in the body. [/Speak]
```

This would result in two further speech buttons, as shown in the final screenshot below:



**Answer 2:** You could speech-enable the introductory text for the quiz. Go back to Moodle in your browser, and press the **My First Course** (MY101) link. Press the **Update** (pen) link next to our Alphabet quiz, and scroll down to the introduction text. Type `[Speak]` before the introduction and `[/Speak]` after. Remember to keep the text short—say one or two lines. Note that the `[br/]` tag can be used to provide a line-break in the text.

`[Speak]` This is a fun quiz to help you learn letters and words. `[br/]`  
Try to answer the question below each word. `[/Speak]`

## Pop quiz

1. What course format would you use for a series of lessons with a clear timetable?
  - a. Social
  - b. Topics
  - c. Weeks
2. What Moodle user-role did you need to add the activities to your course?
  - a. Student
  - b. Teacher
  - c. Course creator

3. Which answer(s) best describe a Moodle quiz?
  - a. An activity with pre-defined questions
  - b. A container that is initially empty, and to which you can add many types of question
  - c. A type of course
  - d. An interactive activity that can also present snippets of content
4. What is the purpose of a filter in Moodle?
  - a. To provide question types for a quiz
  - b. To transform input text and HTML into useful output text and other elements
  - c. To create a course format

## Summary

In this chapter we learned a lot and made a great start on our journey with Moodle.

Specifically we learned how to:

- ◆ Log in to Moodle
- ◆ Create a course
- ◆ Add a quiz activity with multiple choice questions to our course
- ◆ Make the quiz more compelling with a speech synthesis filter

We also started to learn about course formats, searching for **open content** like images, and writing content in Moodle.

Now that we've learned about the quiz activity module it's time to cover **Moodle resources** for teaching math, which is the topic of the next chapter.



# 2

## Basic Math in Moodle

*This chapter demonstrates resources and tools in Moodle for teaching math to younger children. We will be using the label and web page resources, multimedia content, and various math-related question types in the quiz to help our class practice their numbers and counting in a fun way.*

In this chapter we will:

- ◆ Use Creative Commons Search and Flickr/Blip.tv to source freely-licensed images/video to help demonstrate and illustrate numbers, addition, subtraction, and multiplication
  - Learn more about open content
- ◆ Create a course, and within it, label resources with uploaded, embedded images
- ◆ Create a web page resource, with an embedded video
  - Learn about embed code (as provided by Blip.tv, YouTube, and so on)
- ◆ Re-visit the Quiz—numeracy, number-recognition, and arithmetic using multiple choice and calculated question types
- ◆ Install the calculated objects question type
- ◆ From calculated to calculated objects—wildcards for repetition

In the previous chapter, we looked at a type of **activity module** with our alphabet quiz, and we did some searches for images on Wikimedia Commons. In this chapter, we are going to experiment with the other major type of module in Moodle, **resources** such as web pages and labels. We can use images embedded in labels to illustrate our pages and draw the interest of the class. How can we find useful material online to help us in our math lessons?

There is a lot of free or open content on the Web, which you can use to enrich your lesson material. Open content is most useful when you are given the right to use, modify, and redistribute—free in the sense of **freedom**, as opposed to no cost. **Creative Commons** is an organization that advocates for open content, mainly by developing and promoting licenses (<http://creativecommons.org/choose/>).



Big organizations which have adopted **Creative Commons** licenses include the Wikipedia family of sites, and the Open Course Ware and Open Educational Resource movements, which are collaborations between many universities including the Massachusetts Institute of Technology (MIT) in the USA and The Open University in Great Britain (<http://en.wikipedia.org/wiki/OpenCourseWare> | [http://en.wikipedia.org/wiki/Open\\_content](http://en.wikipedia.org/wiki/Open_content)). [I should declare here that I am employed by The Open University. \*\*]

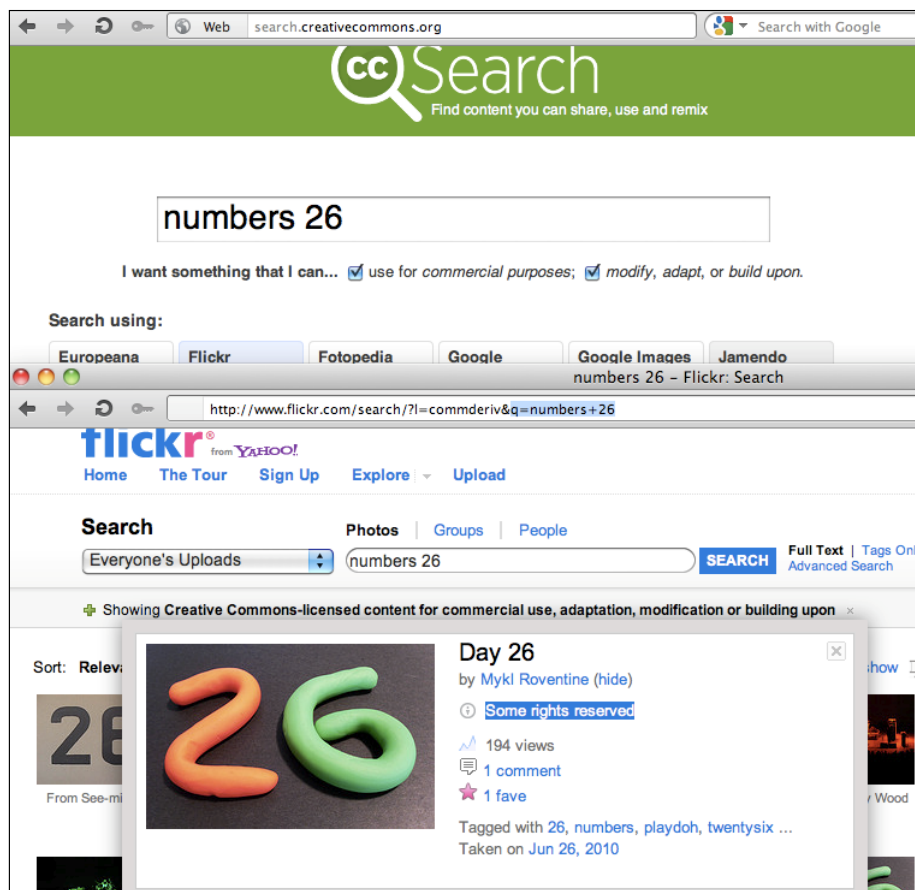
## Searching for open content

Creative Commons also provide a useful service that allows you to search for Creative Commons-licensed works. Note that this is not a search engine, but an umbrella for search facilities on other sites that allow one to search by license. So for example, you can use it to search for freely-licensed images on Google Image and Flickr, and freely-licensed video on YouTube.

### Time for action – searching for open content

Why not go to Creative Commons search now (<http://search.creativecommons.org>) and enter a search term such as `numbers typography`. Note, that if your school is state-run or public, you can probably leave the **use for commercial purposes** checkbox un-ticked, which will give you more results. Press the **Flickr** button, and you will find an attractive series of number images on Flickr, by See-ming Lee (**seeminglee**) (Creative Commons Attribution ShareAlike License, CC-by-sa).

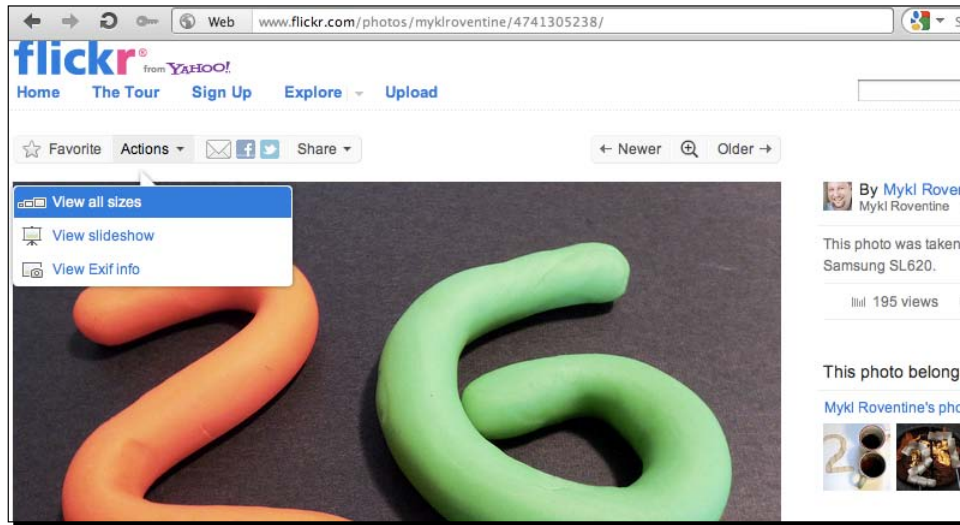
This license is probably adequate for our teaching, but not ideal for this book. So I did a further search for **numbers 26** as you can see in the next screenshot. (Contrived you ask. No, never.) This yields the image **Day 26 by Mykl Roventine**, available under the more liberal Creative Commons Attribution License (<http://flickr.com/photos/myklroventine/4741305238/>).



You can follow these steps:

1. When you hover over an image on Flickr you will see an **i** icon. Press it and an information popup will appear, as shown in the following screenshot. You will see the phrase **Some rights reserved** which indicates that the owner has assigned an open-content license to the image.
2. Click on the image to go to the page for that image. If you have an account with Flickr or Yahoo, you can log in and use the **Share this** functionality. However, we are going to proceed as if you are not logged in, and we are going to take a copy of the image to embed.

3. Click on the **Actions** button just above the image and choose **View all sizes**, as shown in the next screenshot.



4. On the available sizes page, choose a link for the size that seems most appropriate for the context—I chose **small** (240 x 180 pixels). Right-click on the resulting image, and choose **Save Image As** from the browser context menu (Firefox).
5. Save the image to a suitable location on your computer, for example **My Documents** (Windows). We will use it shortly.

You will want to keep a browser tab or window on this image in Flickr, so open a new tab or window to continue.

### ***What just happened?***

We went to the Creative Commons search facility and put in the search terms, `numbers typography` and `numbers 26`. We were able to view Creative Commons-licensed images from Flickr. And we followed links through to Flickr to save a copy of our chosen image. Next we will find out how to use it as an illustration in our course.

## Creating a label

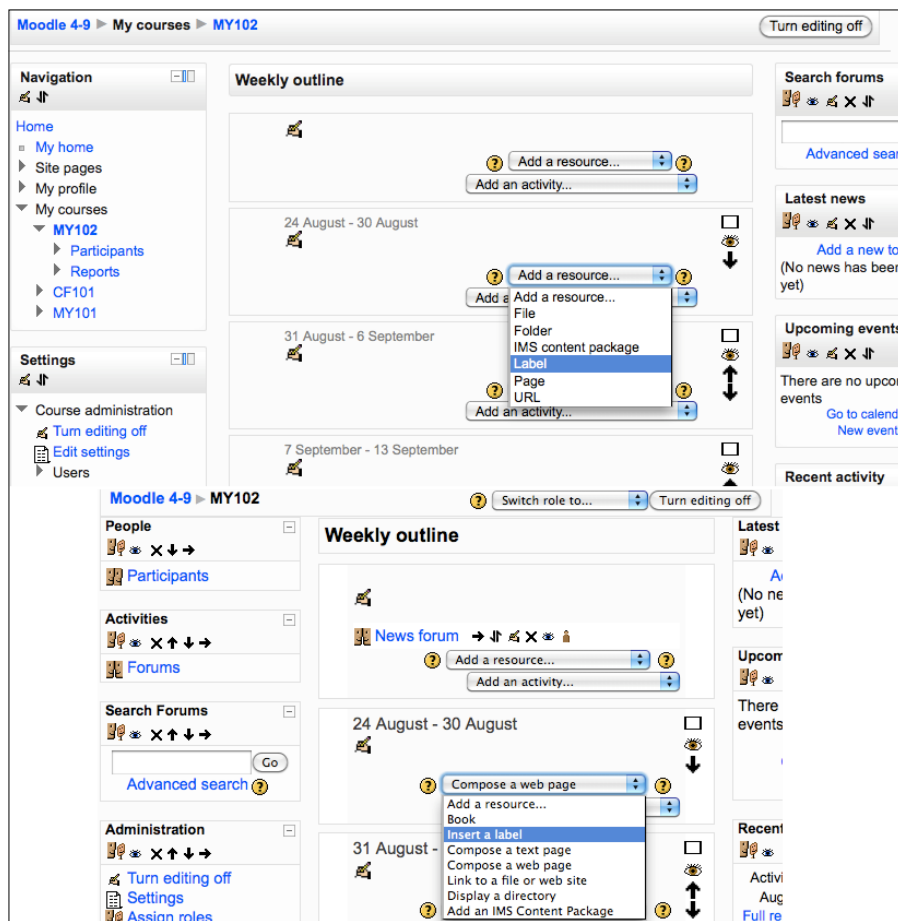
Now that we have some material, we will start embedding it in a course. Remember, the images we are going to embed will add interest to the course page, and allow our class to practice their number recognition skills.

### Time for action – creating a label resource

Follow these steps to create a label resource in Moodle 2:

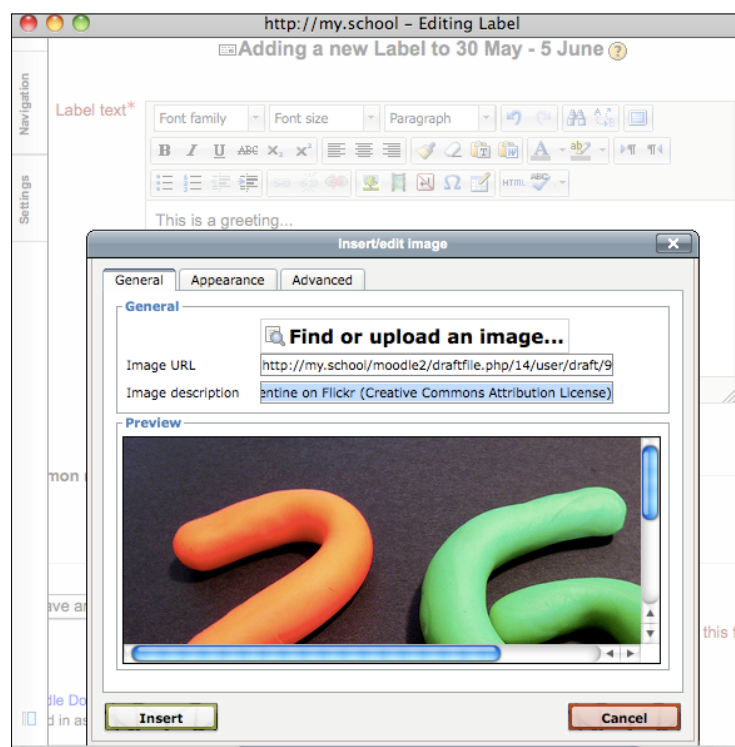
1. Log in to your Moodle site using the teacher account you used in *Chapter 1, Getting Started*. In the **Settings** side-block, expand **Site administration**, then **Courses** and click on **Add/edit courses**. Under the **Course categories** list press the **Add a new course** button.
2. In the **Edit course settings** form, leave **Category** set to **Miscellaneous**. Enter a **Course full name**, such as *Basic math*, and a **Course short name** like *MY102*.
3. This time we will use **Weekly format** as the course **Format**. You have the opportunity to set a realistic **Course start date** for this example. It can be several weeks or a month in the future.

4. Set the **Number of weeks** to a manageable number such as 5, and reduce the **Maximum upload size** to a number such as 1 MB. Scroll to the foot of the page and press **Save changes**. Skip over the **Enrolled users** page by choosing **MY102** in the breadcrumb trail near the top of the page.



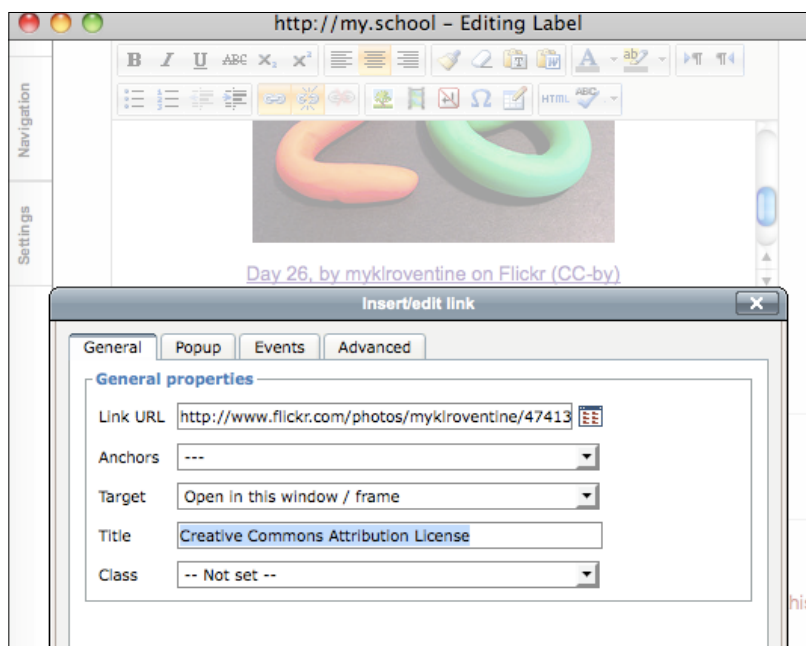
5. If it is not already active, press the **Turn editing on** button now. Go to the first week and choose **Label** from the **Add a resource...** drop-down menu. As you can see, a label does not have a title or summary, simply a **Label text** or **body**. You may want to type a greeting message for your pupils, and then press *Enter* for a new line.
6. Press the **Insert/edit image** button in the editor. You will remember that in *Chapter 1, Getting Started*, we just inserted a URL in the insert image dialog. This time we are going to upload the image that we saved from Flickr.

7. Press the button labeled **Find or upload an image...** towards the top of the **Insert/edit image** dialog, then press the **Browse...** button in the **File picker** dialog. Find the image on your computer (in Windows, try **My Documents**). Press the button to **Upload this file** and after the image has been transferred it will appear in the **Preview** pane of the **Insert/edit image** dialog.
8. Enter a suitable **Image description** text as shown in the next screenshot. For instance **Day 26, by myklorentine on Flickr** and press **Insert**.



9. After the image press *Enter* for a new line, and type a phrase, for example, **Day 26 by myklorentine on Flickr (CC-by)**.
10. Select the phrase with your mouse and press the **Insert link** button in the editor.

- 11.** Copy the URL from your other browser window, which is still on the Flickr page. Paste it into the **Link URL** field in the **Insert/edit link** dialogue (as shown in the next screenshot), and press the **Insert** button.



- 12.** Scroll down to and press the button **Save and return to course**. You will see your label with the image and link on your course main page.

## ***What just happened?***

We created a new course and this time we used the weekly course format. When you are taken to the course main page, you will see empty week-based topics down the center. We added a **Label** as our first resource. This can contain some content to be presented on the course main page. Note that a label is a flexible tool to liven up pages, link together other resources and activities for your class, and improve layout.

We inserted our image from Flickr in the label resource and added an attribution link using Moodle's rich editor. The attribution link mentions the license, which helps us conform to the terms of the Creative Commons Attribution License.

The image will add some color to the main page for our math course, and provides a useful introduction to the quiz we will create later. Now that we've explored the use of static images, we will look at embedding video in our course.



## Embedding a video

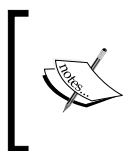
We will find further uses for images later in the chapter. In the meantime, let us suppose we wish to find resources to supplement our teaching, perhaps to help struggling pupils with key mathematical concepts. Videos are a useful supplementary resource. They can provide an alternative presentation of a concept to improve our class's understanding.

### Time for action – embedding video

Visit Google and try a search for **subtracting witte** (<http://google.com/search?q=subtracting+witte>). You should come across a video called **Subtracting Integers** by the Witte Math Program (<http://blip.tv/file/1837295>). It carries a Creative Commons Attribution License, so we are free to use it as long as we provide an attribution link. Follow the link in the search results to Blip.tv now and preview the video.

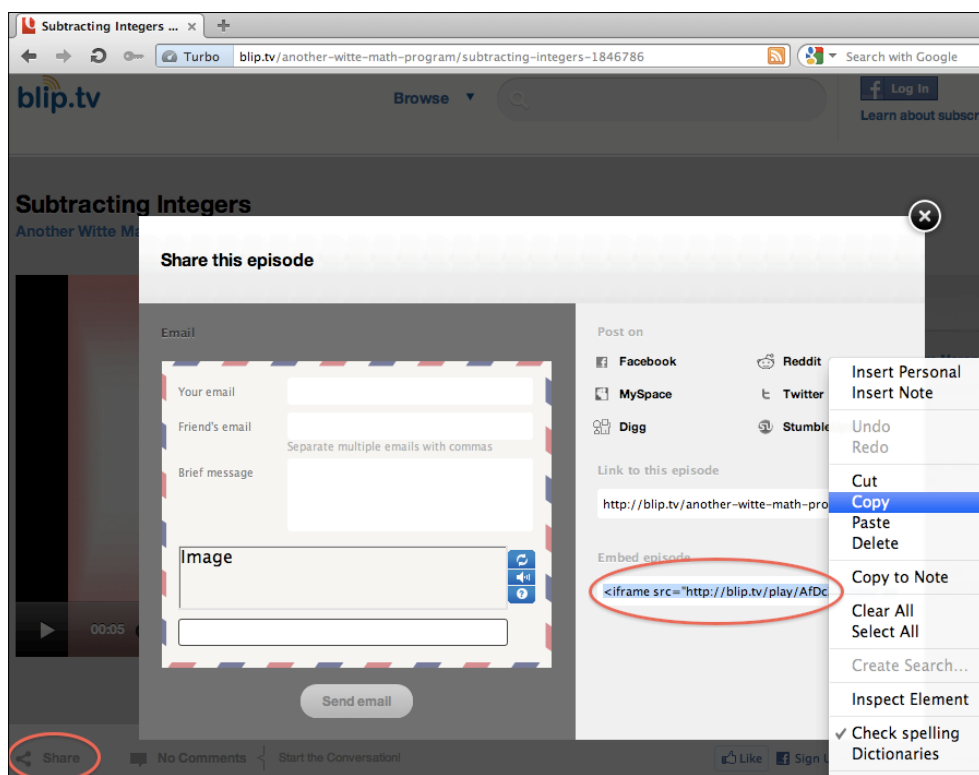
We are going to embed this video in our Moodle course:

1. Return to the course main page, and the week two topic. This time choose **Page** or **Compose a web page** (in Moodle 1.9) in the **Add a resource** drop-down menu.
2. You will be presented with a page containing an increasingly familiar form. In the **General** section, enter a **Name** for the page, perhaps **Math tutorial videos**, and a **Description** (**Summary** in Moodle 1.9) such as **Videos to add to your understanding of key concepts**.
3. Under the section labeled **Page content** (**Compose a web page** in Moodle 1.9) type the name of the video, two newlines, then the attribution **Another Witte Math Program, on Blip.tv**. As previously, use the **Insert a link** editor tool to convert the attribution text into a link to the page on Blip.tv.



Do not be too concerned about the links out of your course to external sites. Sites such as Blip.tv are fairly safe for young children, and if they follow a link for a specific video, they will see links to related videos, in this case on the topic of math. This may spark further research, and greater understanding.

4. In a new tab or window in your browser, return to the Blip.tv page, and below the video player find the **Share** link. Press it and in the resulting dialog select the **Embed episode** textbox, to highlight what is known as the HTML **embed code** for the video. This starts `<iframe src=...>`. Right-click with your mouse to bring up the context menu, and **Copy** to the clipboard as indicated in the following screenshot:



5. Return to the Moodle page to edit your resource. Press the **Toggle HTML source (<>)** button now to work with the HTML directly.

```
<h2>Subtracting Integers</h2>
<iframe
  src="http://blip.tv/play/AfDcZgI.html"
  width="550"
  height="396"
  frameborder="0"
  allowfullscreen>
</iframe>
<embed
```

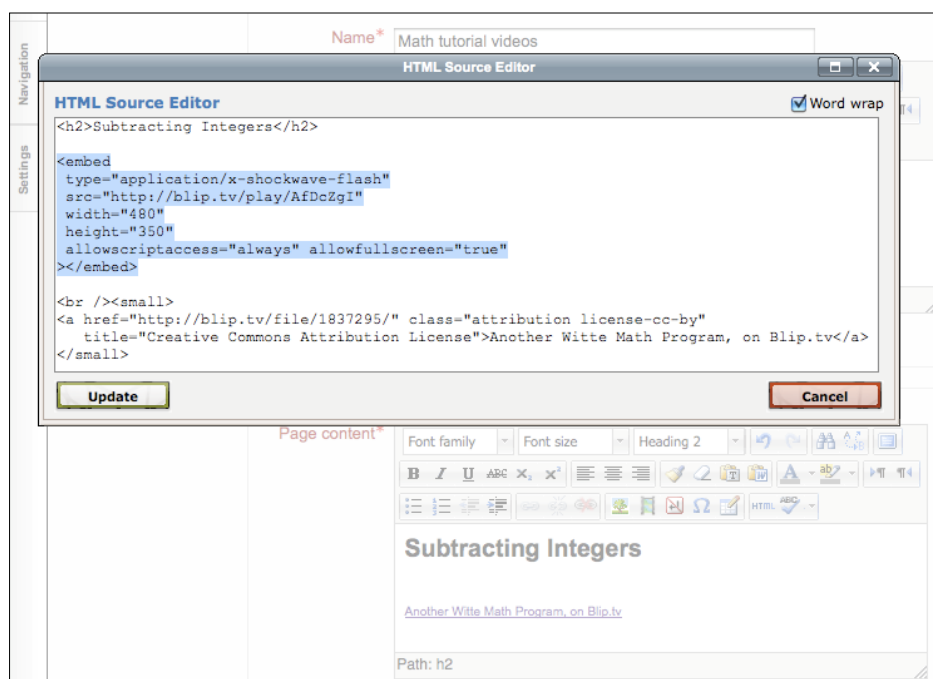
```

type="application/x-shockwave-flash"
src="http://a.blip.tv/api.swf#AfDcZgI"
style="display:none">
</embed>

<br /><small>
<a class="attribution license-cc-by" href="http://blip.tv/fi..."
  title="Creative Commons Attribution License">Another Witte
  Math Program, on Blip.tv</a></small>

```

6. Paste the embed code between the title (<h2>Subtracting Integers</h2>) and the attribution link (<br />...<a class="attribution"...">), as shown in the next screenshot (line breaks have been added for clarity). You will notice that there is an attribute named `class` on the link below the embed code, which we will find useful for styling.



7. Beneath the editor, in the section titled **Window** use the default value of **Same window**. Press the button to **Save and display**.

## What just happened?

We searched for a math-related video on Blip.tv and found its **embed code**. The embed code enables you to insert the video in your page and, as in many cases, uses the Shockwave Flash plugin for the browser. We pasted the embed code in a web page resource, in our Moodle course. You will see the video embedded in your web page, and you will be able to play it.

To troubleshoot, in Internet Explorer choose **Tools | Internet Options | Programs | Manage add-ons**. In the Firefox browser choose **Tools** from the menu, then **Add-ons** and the **Plugins** tab. In each case, check that **Shockwave Flash** is in the list of add-ons and is enabled.



The embed code for Blip.tv uses the `<iframe>` HTML element, while some other sites use the `<object>` element. Do not be unduly concerned about this. And be careful how you alter the code—if you make changes, you will need to check that the video plays in different browsers, for example Firefox, Safari, as well as Internet Explorer.

At the time of writing, many multimedia sites are making the transition to `<iframe>` embeds to take advantage of the soon-to-be standard HTML5. Never fear, embeds using `<object>` will be supported by sites such as Blip.tv and YouTube for some time to come. We touch upon an aspect of HTML5 in the *HTML5 jigsaw* section in *Chapter 9, Embedding the Web*.

## Creating numerical questions

We are going to use the Moodle quiz to develop different sorts of questions for numeracy, number-recognition, and arithmetic. These could be used as flash cards. As with many resources and activities we create in this chapter, your class can use them individually or in pairs and small groups to practice math. And after you have followed the steps below, you will find them a fairly straightforward way to create activities.

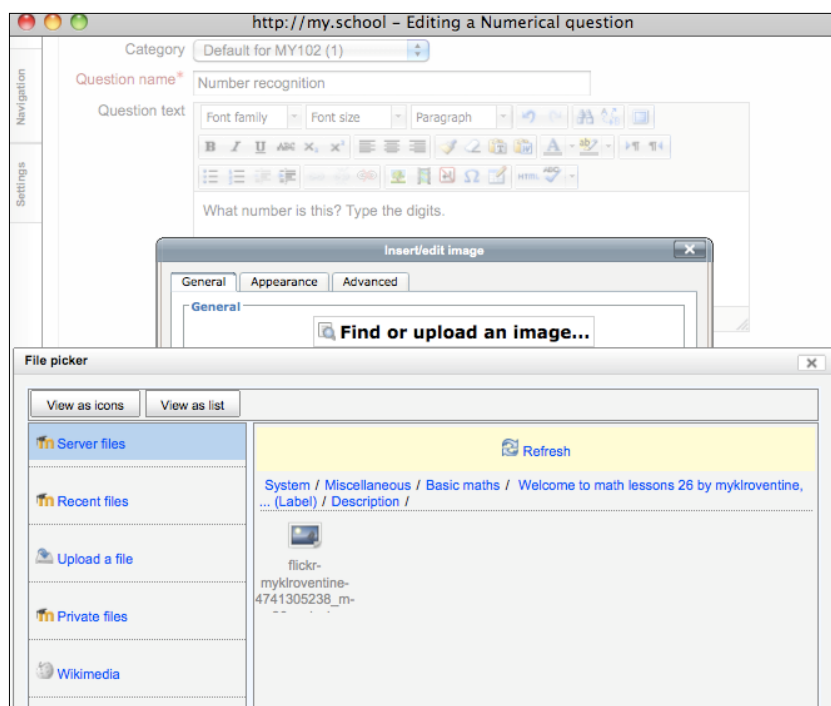
You will probably want to separate the different sort of questions into separate quizzes when you use them to teach.

### Time for action – creating numerical questions

Just follow these steps to create a quiz and some questions in Moodle 2:

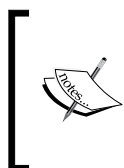
1. Go to the course main page, and choose **Quiz** from the **Add an activity...** drop-down menu. Enter a **Name** for the quiz, for example **Math quiz**, and an **Introduction**.

2. Under the **Layout** section of the form choose **Every question** from the **New page** menu (**One question per page** in Moodle 1.9) and set (**Shuffle questions**, Moodle 1.9) and **Shuffle within questions** to **Yes**. These are both designed to make it harder for pupils to copy, but they will also make the result more interesting. Scroll to the bottom and press **Save and display**.
3. You will be taken to the **Question bank** for the quiz. Next to **Page 1** press the **Add a question...** button. Select the **Numerical** radio button in the popup dialog and press **Next**. (In Moodle 1.9, under **Create a new question**, choose the **numerical question type**).
4. We will start off with number recognition. For the **Question text** I entered **What number is this? Type the digits**. Add a newline then press the editor's **Insert/edit image** button.
5. In the **Insert image** dialog, press the link labeled **Find or upload an image....** The **File picker** dialog will appear. Choose the **Server files** link on the left. Browse to the **number 26 image by mykloventine** that we used previously (Creative Commons Attribution license). This will involve following a path in the **File picker**, for example, **Basic math / Welcome to math lessons...(Label) / Description /**, as shown in the following screenshot. Click on the link for the file, then press the button labeled **Select this file**. Press **Insert** on the **Insert image** dialog.



6. Leave **General feedback** blank, and for **Answer 1** put the correct answer of 26 and a **Grade** of 100%. The **Accepted error** field is specific to the numerical question type, and can be used to allow a range of answers. Leave this blank, and under **Feedback** type *Well done!*
7. It would be encouraging for the student to give feedback if their response is close to the correct one, so for answer 2 type 25, choose a grade of perhaps 5% and in feedback enter *You're close. Try again.*
8. Repeat for answer three with the value 27. We have filled the three answers that the form provides by default. So, press the button for **Blanks for 3 More Choices**. The final answer we will cater for is the **catch-all** which for the numerical question type is denoted by \*. Type this in the answer field for answer 4, leave a **Grade** of **none** and under feedback enter a phrase like *That's wrong. Please try again.*

Note in Moodle 1.9 the process is slightly different. After step 4 above, instead of using Moodle's rich editor to insert an image, you should be able to see a list of images and files that you uploaded for the course immediately below the editor, labeled **Image to display**. Choose the number 26 image from Flickr that we used previously.



At this point it is worth noting that quizzes do not need to be graded, and you can configure the quiz so that the pupils do not see their score. This is useful where you want a quiz simply to reinforce the learning. Go to the quiz editor, choose the **Edit** tab, and in the second row of links choose **Quiz**. Instead of the default of 1 enter a grade of 0 against all the questions.

## What just happened?

We used an image saved from Flickr in a numerical question for our math quiz. We were able to tailor feedback based on how close the pupil's response was to the correct answer. This included creating a catch-all response, using the wildcard \*.

We'll move on from number recognition to sequences.

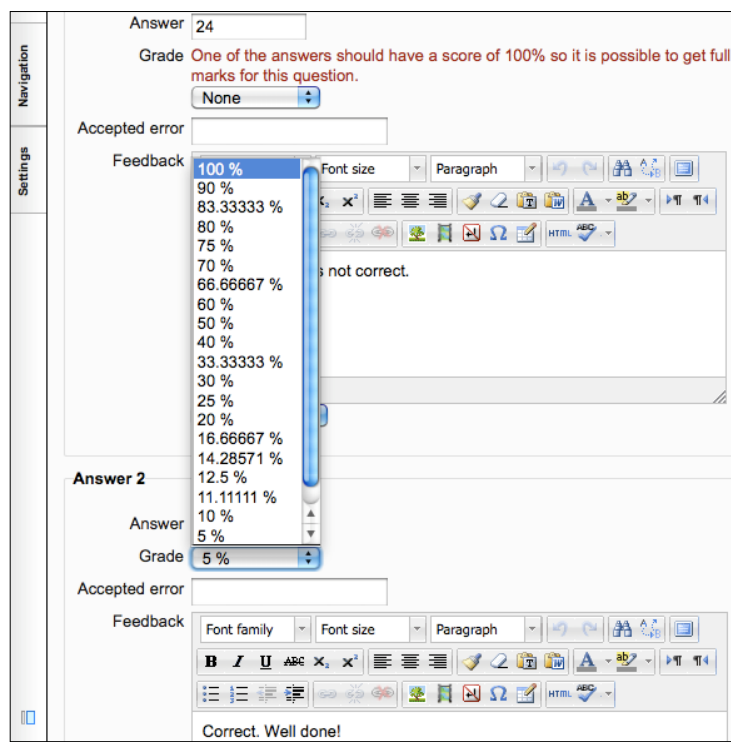
## Creating more math questions

Now that we have one question done, it is time to save more images from Flickr. You may want to find some images of numbers that form a sequence, for a "What comes next?" question. And you may want to search for photos of objects to count. I tried queries such as **number typography** and **trees**. And in Flickr, on the **all available sizes** page I chose and saved the **small** image that is 240 pixels wide.

## Time for action – creating more maths questions

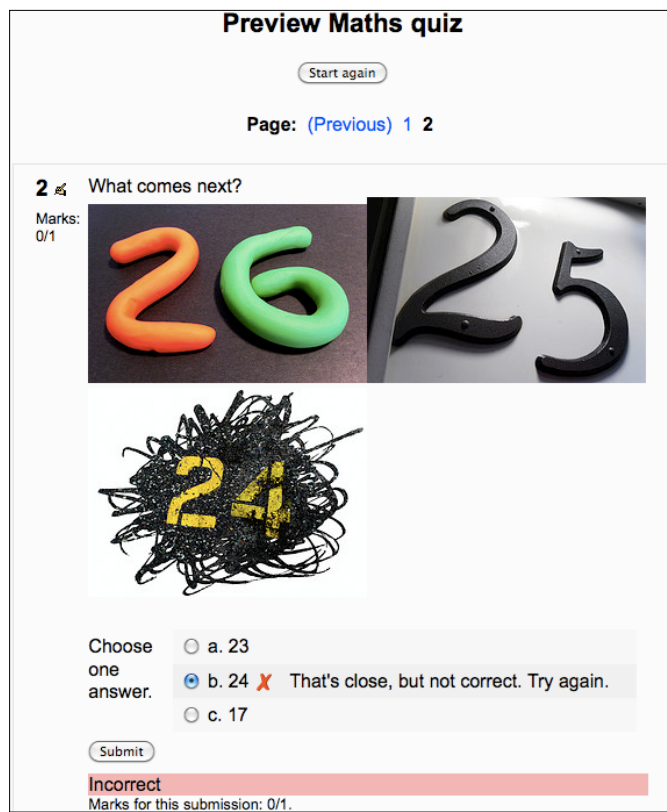
These are the steps:

1. For a sequencing question, you may want to use the multiple choice question type as we did in *Chapter 1, Getting Started*. Choose **multiple choice** now in the quiz editor.
2. In the **Question text** type a phrase like **What comes next?**, press *Enter* for a new line, then insert the images for 26, 25, and 24 in a row (I used Creative Commons images by *myklorentine* on Flickr). Note that because we have more than one image we have to insert them through the editor—the **image to display** only allows us to add one image.
3. Enter some incorrect and correct answers for **Choice 1** to **Choice 3**. Remember to set a grade of **100%** for the correct answer. As the following screenshot shows, when you try to save the question you will get a red box around **Grade** if none of the answers have a grade of 100%.



4. Save the question and in the question editor add it to your quiz. Preview the result.

Depending on the width of your browser window, you may see the images **wrapping** as shown in the following screenshot. This will make the question harder to understand. You may also have different image proportions, which look unsightly. How can we fix this?



Go back and edit the question:

1. First select the question text **What comes next?** and make it a heading by choosing **heading 3** from the third drop-down menu in the editor.
2. Then click **Edit HTML source** button. The significant parts of what you are aiming for are highlighted below (line-breaks have been added, and URLs truncated for clarity). There is an outer `<div>` element, with an attribute `style` which is equal to `min-width: 740px;`—that is, make this page element at least 740 pixels wide (3 x 240 pixels + a margin). Ensure that the height of each image `<img>` is set at 160 pixels (the first of the three was 154 pixels initially).



3. Next, create a `style` attribute on the middle image and put `margin: 0 5px;` (a shorthand way of saying, make the margins 0 top and bottom and 5 pixels left and right).
4. Finally, add an attribution link, and close your **div** element (`</div>`).

The result in the editor for the question should look something like the following HTML source code:

```
<div class="flickr image mx3" style="min-width: 740px;">
<h3> What comes next? </h3>



<br /><small>
<a title="Photos tagged 'numbers', Creative Commons Attribution
  License"
  href="http://flickr.com/photos/myklroventine/tags/numbers/"
  class="attribution license-cc-by">Photos by myklroventine, on
  Flickr</a></small>
</div>
```

## What just happened?

If you got this bit of hacking right first time, well done! Note that the `min-width` style rule will not work in older versions of the Internet Explorer browser (less than version 7). So, the question will still work fine, but it will not look quite as we wish (this is termed **graceful degradation** and is an acceptable principle to follow, especially for browsers that are at least 10 years old).

If you ran into problems, go back and try again. Check that all the elements like `DIV` are closed with a `>` (greater than) symbol, and that attribute values are surrounded by double quotation marks. Do experiment, as you won't cause any damage to your Moodle system. And do not worry, we will continue to use the rich-text editor to create content where possible, and only stray into hand-editing occasionally.

We added several images to a multiple-choice question in our example math quiz. This second question asked the student to complete the sequence of numbers. Then, we hand-edited the markup (Hyper-Text Markup Language) and inline styles (Cascading Style Sheets) to improve the appearance and readability of our question.

Next we'll look at the **Calculated** question type.

## A calculated question

As you have added questions you may have noticed the **Calculated** question type. Like the Numerical type we have tried already, this was purposefully created for mathematics. We are going to create a calculated question now, to enable our young charges to practice their arithmetic.

### Time for action – creating a calculated question

You will want to follow these steps now:

1. Choose the **Calculated** question type now.
2. Enter `Addition 1` for the **Name**, and in the **Question text** area type `What is {a} + {b}?`. The two different letters enclosed in curly braces denote wildcards, for which the system will substitute real numbers (we could have used any `{name}`).
3. Lower down in the answer section of the form you will see a field labeled **Correct answer formula** =. In this field type the formula `{a} + {b}`. Remember that the question text is used to present a question to the user, and can be formatted and styled. The correct answer formula field is used by Moodle to calculate the expected answer.
4. Set the **Grade** to 100% and against the label **Correct answer shows** select **0**, with the default format of **decimals** (that is, display no decimal places).
5. Scroll down to the bottom of the form and press the **Next page** button. Now that we have written the template or pattern for our question, it is time to choose the set of data to use.
6. You will see a form labeled **Mandatory wild cards present in answers**. The form contains a drop-down menu for each of our wild cards, a and b. The two options are use the same existing private dataset, and use a new shared dataset. Use the default of `private` for now (as yet, we do not have a dataset). Press the **Next page** button. Nearly there!

**Edit the datasets** ?

Update the datasets parameters

**Item To Add**

Param {a} 9.7

Range of Values 1.0 - 5

Decimal Places 0

Distribution Uniform

---

Param {b} 3.4

Range of Values 1.0 - 10.0

Decimal Places 1

Distribution Uniform

---

{a} + {b} 9.7 + 3.4 = 13.10  
 Min: 12.969--Max: 13.231  
 Correct answer : 13.10 inside  
 limits of true value 13.1

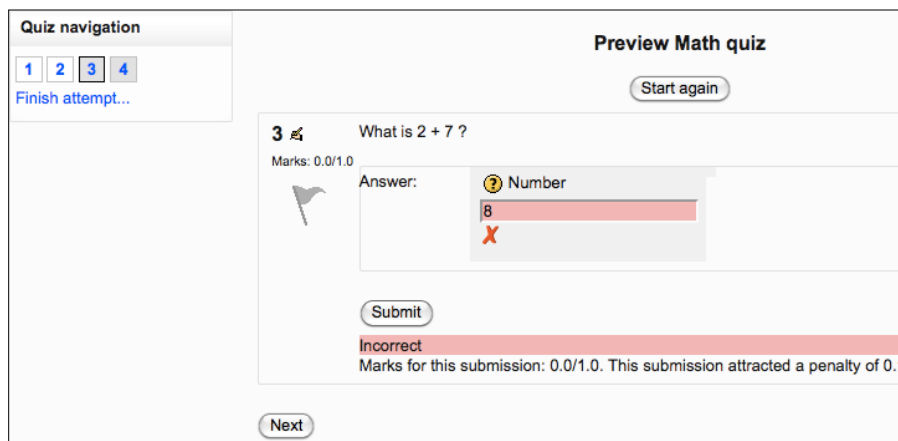
**Add**

Next 'Item to Add' ☒ reuse previous value if available  
☐ force regeneration

Get New 'Item to Add' Now

7. As the previous screenshot shows, Moodle's question engine has already created possible values for a and b. However, we need to set **Decimal Places** to 0, and it is probably best to adjust the range of values to use for each parameter. I chose 1 to 6 for parameter {a} and 4 to 10 for {b}.
8. Press the button at the top to **Update the dataset parameters**. The fields next to **Param {a}** and **Param {b}** will have changed, and will be within our new ranges.
9. Scroll down, and in the section of the form labeled **Add** leave '1' as the number of items to add in the drop-down menu. Press the button labeled **Add** next to the menu. You will see new values for a and b at the top of the page, and the first pair of values we generated in the **Item 1** section near the bottom. Press the button below to save changes.
10. You will have been taken back to the main quiz editor, and our new question **Addition 1** should appear in the question bank on the right, highlighted in green. Tick the box against the name and press the button **Add to quiz**.

- 11.** Click the **Preview** tab at the top of the quiz editor. Click the **Next** link several times to reach your new addition question. You will see the very simple question, and a field for the answer. Try a wrong answer now. You will get feedback like the one shown in the following screenshot:



And we are done.

## ***What just happened?***

We added a math question to our quiz, using the built-in **Calculated** question type. This included manipulating datasets.

There is a lot of potential in the calculated question. You may use the 5 operators  $+$   $-$   $*$   $/$   $\%$ , where  $*$  denotes multiplication, and  $\%$  is the modulo or remainder operator. When you use the  $*$  multiplication operator in the **Correct Answer Formula** field, you may want to use the entity `&times;` in the question text, as illustrated in the next table.

There are also single argument functions, for example in trigonometry (`sin`, `asin`, `cos`...), natural logarithm (`log`), and so on. There is one single argument function, `pi`, but note that as with the other functions, this requires round brackets `()` when used in the **Correct Answer Formula** field. You may be interested to know that by typing an equation in the **Correct Answer Formula** field, you are effectively writing a software program or expression—these functions correspond directly with the functions in PHP, the programming language that Moodle is written in.

| Correct Answer<br>Formula field | Example question texts                                  | Question text—View<br>HTML source | Notes  |
|---------------------------------|---|-----------------------------------|--|
| $\{a\} + \{b\}$                 | ... $\{a\} + \{b\}$ ; $\{a\}$ plus $\{b\}$              |                                   |  |
| $\{c\} * \{d\}$                 | ... $\{c\} \times \{d\}$ ; $\{c\}$ multiplied<br>by ... | $\{c\} \&times; \{d\}$            | This is the HTML<br>entity <code>&amp;times;</code> ;    |
| $\{c\} / \{d\}$                 | $\{c\}$ ; $\{c\}$ divided by...<br><br>$\{d\}$          | $\{c\} <hr /> \{d\}$              | This is the<br>"horizontal rule"<br>HTML tag.            |
| $\sin(\{a\})$                   | $\sin \{a\}$ ; What is the sine<br>of...?               |                                   | The PHP single<br>argument function,<br><code>sin</code> |
| $\{a\} + \pi()$                 | $\{a\} + \pi$ ; $\{a\} + \text{PI}$                     | $\{a\} + \&\pi;$                  | The PHP zero<br>argument function,<br><code>pi</code>    |

As we noted, there is a lot to the Calculated question type. However, when writing material for young children you will probably not make much use of the functions. What you will want is a simple means to illustrate and brighten the arithmetic exercises and quizzes you create for your class. The **Calculated Objects** question type fills this need.

## Installing the Calculated Objects question type

Calculated Objects is a third-party Moodle plugin available from [Moodle.org](http://moodle.org), and written by the author. To install this contributed plugin, you will again need the help of your friendly IT support person, as you will probably not have the required permissions.

### Time for action – installing a question type

Please pass on these installation instructions:

1. The plugins database on Moodle.org lists all third-party contributed modules and plugins. Visit the entry for the **Calculated Objects** question type at: <http://bit.ly/moodle-calcobjects> ([http://moodle.org/plugins/view.php?plugin=qtype\\_calculatedobjects](http://moodle.org/plugins/view.php?plugin=qtype_calculatedobjects)). Check that the plugin is compatible with your version of Moodle, and look for updates to these installation instructions.
2. Backup your database. (This is always good practice!)

3. Download the appropriate version of the code for your Moodle, and uncompress the archive. The resulting directory may be called something like `nfreear-moodle-qtype_calculatedobjects-SOMETHING`. Copy it and its contents to the `question/type` directory on your server and rename it `calculatedobjects`. For example on Redhat Linux you might end up with `/var/www/moodle/question/type/calculatedobjects/`
4. In your browser, visit the administrator notifications page for your Moodle site, for example, `http://my.school/moodle/admin/`. Currently there are no database tables created for this plugin (but this may change).
5. As an administrator, you can test the question type, by visiting **Front Page, Front Page Questions** (for example, `http://my.school/moodle/question/edit.php?courseid=1`), and finding **Calculated Objects** in the drop-down menu next to the create new question label.

You may now hand back to the teacher, with our thanks!

### ***What just happened?***

Our system administrator or IT support person downloaded and installed the Calculated Objects plug-in. This is a third-party contributed question type, which will facilitate the creation of colorful math quizzes to stimulate our young class.

## **Creating your first calculated objects question**

**Calculated Objects** is an extension of the Calculated question type. It allows you to create questions such as "*I have {apples} and {pears}. How many items of fruit do I have?*", where the wildcards {apples} and {pears} are used to generate wording (for example "I have 5 apples..."), and N and M images of apples and pears, respectively.

As we shall see, this question type is well suited to teaching math to younger children.

### **Time for action – creating a calculated objects question**

Let's try it out.

1. Return to Moodle, and visit the editing page for your quiz. When you click on the **Add a question...** button, **Calculated Objects** will appear in the pop-up dialog listing available question types. Choose it, and press the button labeled **Next** and you will be taken to the question editing form.

2. Leave the category as **Default for Basic Math (N)**. And you can enter a question name like **Objects addition 1**. For the question text the principles are similar to those for the calculated question type. Type a question, with wildcards surrounded by curly braces. The wild cards are from a pre-defined list of supported objects.



Currently the list is: apple, red apple, orange, pear, pineapple, walnut, coffee, cookie, tomato, car (check the documentation available via the Moodle plugins entry for any additions, [http://moodle.org/plugins/view.php?plugin=qtype\\_calculatedobjects](http://moodle.org/plugins/view.php?plugin=qtype_calculatedobjects)). The wildcards can be singular or with an 's', they should be in lowercase, and each wildcard should only appear once. We will come to how to add apples to apples shortly.

3. As you can see in the previous screenshot, I typed **I have {apples} and {+} {pears}. How many pieces of fruit do I have?** The plus sign in square brackets is required because the plus sign is not in the normal flow of the sentence. It will be used by the plugin when the question is displayed.
4. Continuing down the page, leave the default feedback empty, and for the **Correct Answer Formula** type **{apples} + {pears}**. Set the grade for this answer to 100%. Then as previously you want to set **Correct answer shows** to 0, while leaving the format as decimals.
5. Type your feedback for the correct answer, then press the **Next page** button at the foot of the page.

6. On the next page we have the opportunity to choose the dataset properties. This time, for each of the wildcards apples and pears we will use a new shared dataset.
7. Click through to the next page and review the values that have been pre-created in the dataset. You will probably need to set the decimal places to 0 as before. Press the **Add** button to add a second set of values then press **Save**.

The screenshot shows the Moodle Quiz editor interface. At the top, there are tabs for 'Info', 'Results', 'Preview', and 'Edit'. Below these are sub-tabs for 'Quiz', 'Questions', 'Categories', 'Import', and 'Export'. The main content area is divided into two panels: 'Questions in this quiz' on the left and 'Question bank' on the right.

**Questions in this quiz:**

| Order# | Question name      | Type          | Grade | Action  |
|--------|--------------------|---------------|-------|---------|
| ↓      | Number recognition | $\frac{1}{2}$ | 0     | 🔍 ⚙️ >> |
| ↑ ↓    | Numerical sequence | $1 \div 2$    | 1     | 🔍 ⚙️ >> |
| ↑      | Addition 1         | $2+2=?$       | 1     | 🔍 ⚙️ >> |

Total: 2  
Maximum grade: 10 ?  
Save changes  
☐ Show page breaks  
☐ Show the reordering tool ?  
Go

**Question bank:**

Category: Default for Basic maths (5)  
☒ Display questions from sub-categories too  
☐ Also show old questions  
☐ Show question text in the question list

The default category for questions shared in context 'Basic maths'.

Create new question ?  
Choose...  
Sort by type, name

| Action      | Question name  | Type          |
|-------------|--|---------------|
| << 🔍 ⚙️ ⌕ ✕ | <input type="checkbox"/> Addition 1                    | $2+2=?$       |
| << 🔍 ⚙️ ⌕ ✕ | <input type="checkbox"/> Addition 1                    | $2+2=?$       |
| << 🔍 ⚙️ ⌕ ✕ | <input checked="" type="checkbox"/> Objects addition 1 | $2+2=?$ 🍏     |
| << 🔍 ⚙️ ⌕ ✕ | <input type="checkbox"/> Numerical sequence            | $1 \div 2$    |
| << 🔍 ⚙️ ⌕ ✕ | <input type="checkbox"/> Number recognition            | $\frac{1}{2}$ |

Select all / Deselect all

8. When you are returned to the quiz editor, you will see our new question highlighted in the question bank. Press the **Add to quiz** button and that is it.

### What just happened?

When you press the preview button and go to question 4 you will see the question as shown in the following screenshot. At the top is the text, with numbers and the words **apples** and **pears** substituted for the wildcards. At the bottom the sum has been laid out horizontally, with pictures of apples on the right and pictures of pears on the left. The plus sign that we put in square brackets when we created the question is between the two.



Quiz navigation

1

2

3

4

Finish attempt...

Preview Math quiz

Start again

4

Marks: ~1.0

I have 4 apples and 7 pears. How many pieces of fruit do I have?

+

Answer:

Number

11

Submit

We installed the **Calculated Objects** third-party module, and created a question for our quiz using it. To create the question we used the wildcards **{apples}** and **{pears}** in the question text and in the correct answer formula. As the question text used **and** in place of the + (plus sign), we put the plus sign in square brackets next to the **and**. We used new shared datasets for the wildcards, and set decimal places to 0 when creating dataset values. And we saw how the wildcards were converted into text and images of apples and pears.

That is a good start, however we will often want to add or subtract like objects—apples with apples for instance. In this case, we differentiate using an underscore and a suffix number, for example **{apples\_1}** - **{apples\_2}**. Note, that if we use a suffix, we must use it for each wildcard. In the cases of multiplication and division, we can mix our object wildcards with single-letter wildcards.

As we have seen, Calculated Objects enables us to create visual math exercises, which makes it well suited to engaging our younger students.

## Pop quiz

Try these quick questions to test your understanding of this chapter. In each case there are up to two correct answers.

1. For which of the following is the Creative Commons Search an umbrella?
  - a. Picasa web photo sharing site.
  - b. Blip.tv video sharing
  - c. Flickr photo sharing
  - d. YouTube video sharing
2. What is the general advice about using HTML embed code?
  - a. Make any changes you see fit—if it works in your browser then it's fine.
  - b. Copy and paste what you are given without making changes.
  - c. Only make changes if you know what you are doing, and test in multiple browsers.
3. Which two quiz question types that form part of core Moodle are particularly suited to maths?

## Have a go hero

Experiment with the calculated objects question type. Try adding like objects, and try multiplication and division. Also try creating 10 to 20 questions using a shared dataset, with the calculated or calculated objects question types. What is the great strength of the calculated and calculated objects question types?

> Possible answer: like objects. The correct answer formula could be `{apples_1} + {apples_2}`.

> Possible answer: multiplication. The correct answer formula could be `{n} * {cookies}`.

> Suggested answer: the calculated question type's strength is the ability to generate an arbitrarily large dataset of random values within a range, and use them in a significant number of questions that follow the same pattern.

## **Summary**

In this chapter we learned:

- ◆ How to search for reusable content useful for supplementing math teaching, using Creative Commons, Flickr, and Blip.tv
- ◆ How to upload content to Moodle and embed content in labels and web page resources
- ◆ How to create math questions in the Quiz using three built-in question types
- ◆ How to create questions using the Calculated Objects third-party module

We also continued to learn how to use Moodle's rich-editor, and where appropriate, how to edit HTML source code.

Now that we've covered the quiz activity and math in some depth, it's time to develop some teaching aids for literacy and storytelling—which is the subject of the next chapter.



# 3

## Telling Stories

*Children have a wonderfully active imagination and enjoy telling stories orally to the people around them. It is important to harness and nurture their creativity and narrative skills from a young age. One way to make writing stories more accessible to children is to present a picture. They can build up a story based on what they see in the picture and they can imagine things like, what happened next? And, what if I was that person?*

*Of course, it is valuable for your class to write their thoughts down. They can start on paper and continue by typing online. They can also record themselves speaking or work in pairs to record each other. If you wish to take the concept in a different direction, particularly with older children, one student can present a picture, which another student can choose as the basis for their story. These ideas will start to develop valuable collaborative skills.*

In the previous chapters, we dealt thoroughly with the Quiz module for literacy and numeracy. In this chapter, we are going to return to the theme of literacy and specifically story telling. Note that story telling can be used as a broad topic and encompass areas such as history and nature. So, we will put the Quiz to one side, and introduce other activity modules.

This chapter guides you through creating a collection of pictures for and with your class, to use as the basis for story telling.

Note that you will need a microphone for this chapter. You may have a head-mounted one, or a microphone built in to your desktop or laptop computer.

In this chapter we will cover:

- ◆ One way to present a collection of images—introducing the Database module
- ◆ Creating your database; defining the fields
- ◆ Accessibility: a long description for images
- ◆ Adding your database entries
- ◆ Adding your example story, as a comment on the database entry
- ◆ Recording an MP3 audio book for your story—installing Audacity and LAME
- ◆ Integrating the audio book in Moodle—the multimedia filter
- ◆ Running your lesson—commenting online, or working in pairs

So now we have a plan, let's dive in.

## Creating a database activity

We are going to start by exploring the **Database** activity module as a means of collecting and presenting images. We can then ask our pupils to use these images as the basis for their story. Database allows you to define a set of arbitrary **fields**, which are akin to columns in a spreadsheet or table. The fields can be of different types. For example, they could be simple text, URLs (web addresses), dates, and picture uploads. After you have set up the database, if you wish your class can add entries to it.

The Database module is built in to core Moodle, so you do not have to install anything.

Following the pattern of previous chapters, we are creating a course for each chapter. This one is called simply **Story telling**, the short name is **MY103**, and this time we will revert to using the **topics** course format. As we have already mentioned, for the activities in this course we will probably be allowing students to upload images. So, it is wise to limit the size of the files that they can upload. Below the course format drop down menu you will see a field called **Maximum upload size**. Change the value from the default to 500 KB (kilo bytes), as shown in the next screenshot. (My default was 32 MB. Depending on the site configuration, yours may be different.)

Category Miscellaneous

Course full name \* Story telling

Course short name \* MY103

Course ID number

Course summary

Font family Font size Paragraph

**B** *I* U ABC

Using the database module and other resources as the basis for story telling.

Path: p

HTML format

Format Topics format

Number of weeks/topics 5

Course start date October 2010

Hidden sections 128MB 100MB 50MB 20MB 10MB 5MB 2MB 1MB

News items to show 500KB 100KB 50KB 10KB 0 bytes

Show gradebook to students

Show activity reports

Maximum upload size 128MB

After you have set the maximum upload size, carry on creating the course. Then scroll down the page and press the button to **Save changes**. When you are presented with the course main page, press the **Turn editing on** button. We are all set to create our activity.

## Time for action – creating a database

Under topic 1, choose **Database** from the **Add an activity** drop-down menu. You will be presented with a form like that shown in the following screenshot:

The screenshot shows a web browser window titled "http://my.school - Editing Database". The browser's address bar shows the URL. On the left, there is a sidebar with "Navigation" and "Settings" tabs. The main content area is a form for editing a database. The form has several sections: "Name" with the value "Story telling pictures", "Introduction" with the text "A place to upload your pictures, and tell a story.", "Path" with the value "p", "HTML format" with a dropdown menu, "Available from" with a date range of 3 October 2010, "Available to" with a date range of 3 January 2011, "Read only from" with a date range of 3 January 2011, "Read only to" with a date range of 3 January 2011, "Required entries" with a dropdown menu, "Entries required before viewing" with a dropdown menu, "Maximum entries" with a dropdown menu, "Comments" with a dropdown menu, "Require approval?" with a dropdown menu, "Grade" with a dropdown menu, "Ratings" with a dropdown menu, "Roles with permission to rate" with a dropdown menu, "Aggregate type" with a dropdown menu, and "Scale" with a dropdown menu. The "Ratings" dropdown menu is open, showing options: "No ratings", "Average of ratings", "Count of ratings", "Maximum rating", "Minimum rating", "Sum of ratings", and "Average of ratings". The "Average of ratings" option is selected. The "Scale" dropdown menu is also open, showing options: "Scale: Separate and Connected ways of knowing".

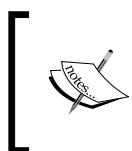
The first few fields in the previous screenshot are becoming familiar. Follow these steps to configure the activity:

1. Type a **Name** and some introductory text. Move past the next date-related fields, which can remain disabled.



2. We can leave the defaults for **Required entries**, **Entries required before viewing**, and **Maximum entries**.
3. The next field is labeled **Comments** and defaults to **No**. Up to this point the activities we have created have been more suited to individual pursuit. However, one of the strengths of Moodle is the tools it provides for collaboration and social participation. So, we will allow commenting, by setting the **Comments** drop-down field to **yes**.
4. And, we will leave **Require approval** set to **No**.
5. Leave the **Grade category** set to **Uncategorized**, and use the defaults under **Ratings**.

We will assume comments are used to present a story, or part of a story, and to give feedback.



Note that by default students can write and view comments (if commenting is enabled) in a database activity. However, they can't by default rate entries, or view ratings—**course creators** (your role), **teachers**, and **non-editing teachers** can.

After you have completed the initial settings form, press the **Save and display** button.

### ***What just happened?***

We created our database activity and configured it to allow commenting.

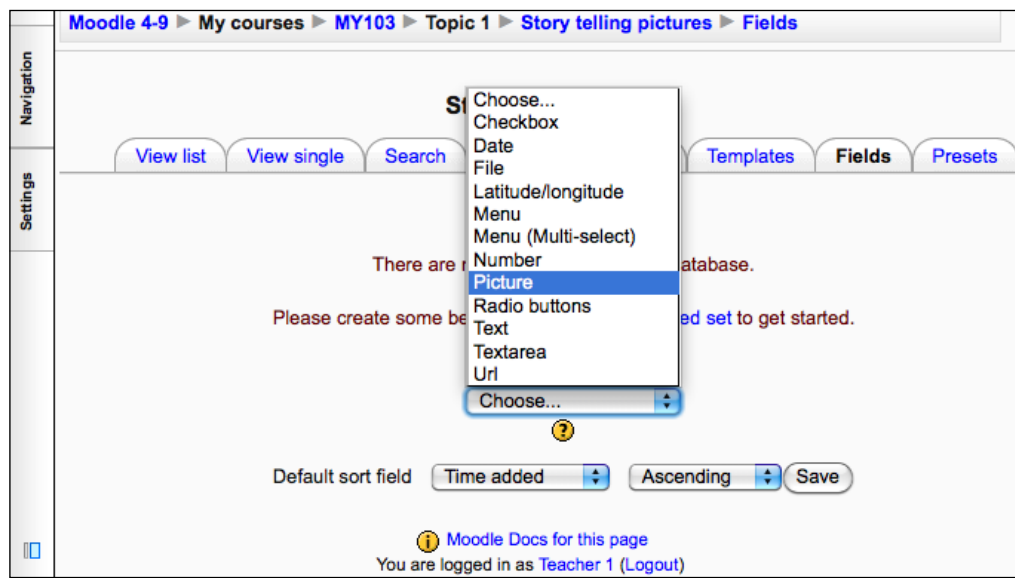
The Database module is one place where you can create social and collaborative activities. A database contains a series of entries, which equate to rows in a spreadsheet or table. And each entry can be commented on by anyone else in the course, if you the teacher configure the activity in that way.

Of course, you should consider the maturity of your class before enabling comments. You will need to explain the etiquette of online discussions, for example, that abusive comments are not allowed. Commenting may need a little moderation. Learning about the appropriate tone for comments will be a valuable lesson for your class.

We have an empty database, so the time has come to add some fields to it. This is the subject of the next section.

## Planning our database fields

After the database activity is created, you will be taken to a page like that shown in the next screenshot. This is the tab named **Fields** in the database editing and viewing interface. There is a prominent message, **There are no fields defined for this database**. There is also a link to choose a pre-defined set of fields, but in order to increase our understanding we will start off by defining our own fields.



Let us plan what fields we want for our picture database. It is useful to note that there are three pieces of meta-data, or data about the content, that the module records for each entry automatically—who created the entry, when, and when the entry was last modified.

Bearing this in mind, some reasonable fields would be:

- ◆ A **text** field: For a short caption.
- ◆ A **picture** field: This allows us to upload a file and identifies it as an image with dimensions, as opposed to some other file type.
- ◆ A **Text area**: For a longer description of the image.
- ◆ A **URL** field: For an optional attribution link. (Most of our images will presumably be provided by the pupils, but this will add some flexibility.)



Giving your class the chance to create a long description of a photograph or picture is a useful way to allow them to think creatively and laterally. It is also of particular interest for **accessibility**, that is, to try to create an equivalent experience for those who in this case, have low vision or who are blind. This may seem a bit abstract when you are creating a lesson, but bear in mind that you are teaching the writers of tomorrow's Web, and giving them the opportunity to think about such things early is worthwhile. (Note, there is more about accessibility for teachers in the *Appendix A, Accessibility for Online Teaching*.)

## Time for action – creating fields

Now that we have taken a moment to plan our database we can implement it. These are the steps you can follow to add fields:

1. Go to the **Fields** tab for our database activity. From the **Create a new field** drop-down menu choose **Text**. You will be taken to a new page with the title, **Text Field** and a form.
2. Against the **Field name** enter some text, for example, **Short caption**, and next to the **Field description** label type some notes for your benefit—these won't be visible in the form presented to pupils. Leave the **Allow autolink** check box un-ticked.
3. Finally press the **Add** button to create the database field.

The process can be repeated for the other fields specified above, although there are some twists. As can be seen in the next screenshot, for the **Picture field** we need to specify dimensions for the **single view** and **list view**. The label implies more results appear in the list view so you will probably want to specify a smaller width and height. Why not try single view dimensions of 500 x 400 pixels, and list view dimensions of 300 x 200 pixels? Also, for the picture field you will want to limit the size of the uploaded file. Following the suggestion earlier to limit uploads within the course to 500 KB we can probably leave this as the value for the picture field. You may want to come back and tweak these values later, after you have tried some uploads.

**Picture field**

Field name: Your picture

Field description: Browse your computer and upload the image that yo

Width in single view: 500

Height in single view: 400

Width in list view: 300

Height in list view: 200

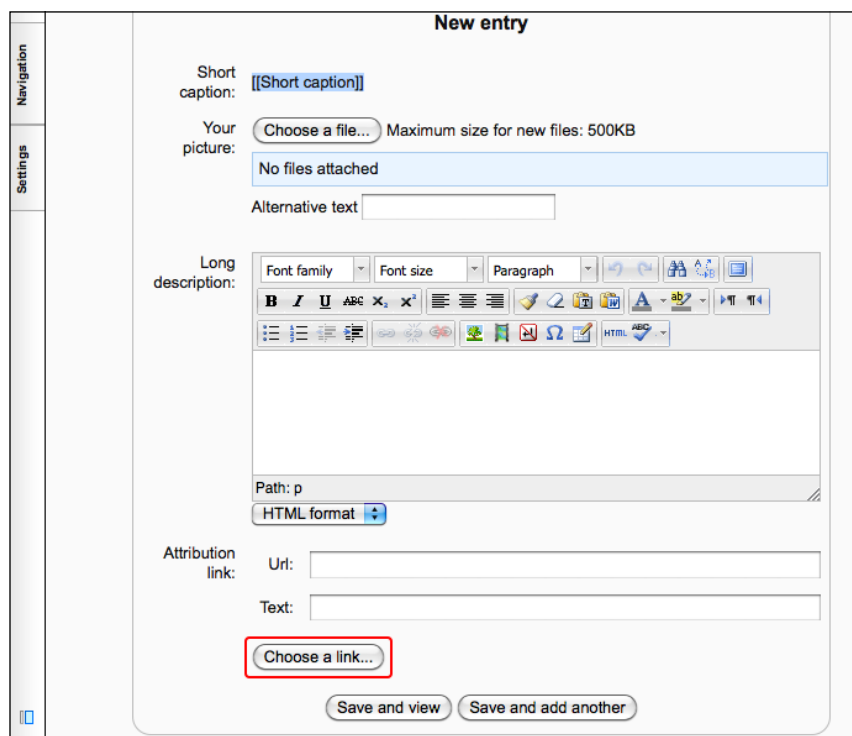
Maximum size: 500KB

**Add** **Cancel**

For the **textarea** field you can start off with the default size for the editor—60 columns wide, by 35 rows high. Naturally you can come back and adjust this later.

Finally, we have the **Url** field. In this case, we want to tick the box to **Autolink the URL**.

After you have added this field you can navigate to the tab labeled **Add entry**. And you will be presented with something like the following screenshot:



## What just happened?

We created a variety of fields in our database including text, picture upload, and URL. The **Picture** field allowed us to specify image dimensions.



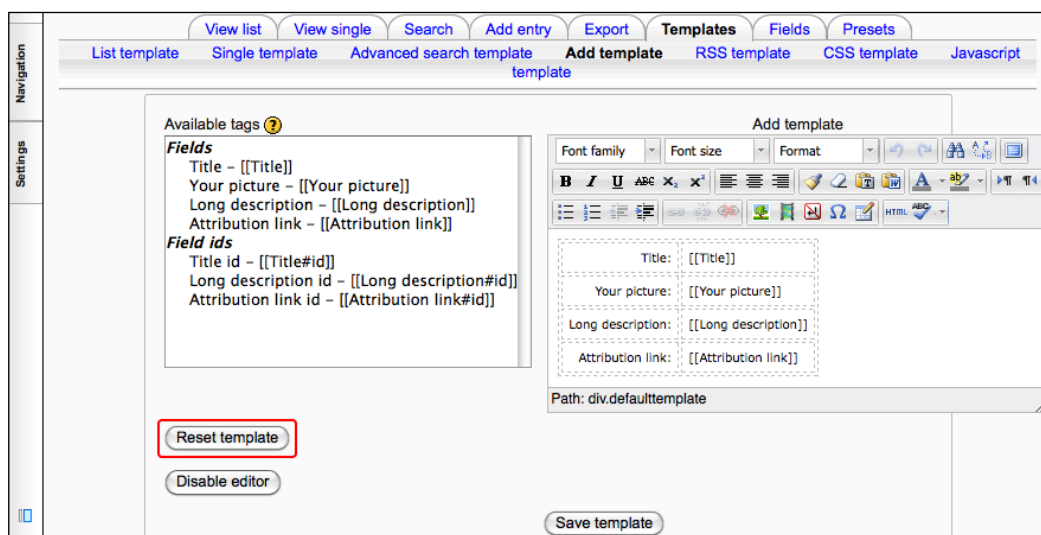
Note that the file upload form control labeled **Your picture** will look slightly different in Moodle 1.9.x and below, and of course the editor will probably be different.

If you are like me and tend to experiment, you may have an "Oh, no" moment like I have above. What is the short caption doing in square brackets **[[short caption]]**, and where is the missing field? This brings us smoothly onto database templates.

## Editing templates

Database templates are the means by which you can adjust the appearance of your **New entry** page. There are seven templates under the **Templates** tab to control the appearance of all aspects of your database activity, including search and viewing lists of entries.

As you can see in the screenshot below, the one we want is labeled **Add Template**, and it controls the appearance of the **Add entry** form. The trick when something looks wrong is to click the **Reset template** button, which forces the templating system to check the defined input fields again. Then remember to press the **Save template** button.



So far, so good, but if we want our class to upload images themselves, then a quick look at the previous screenshot would indicate that it is not very usable. Can we add any extra instructions?

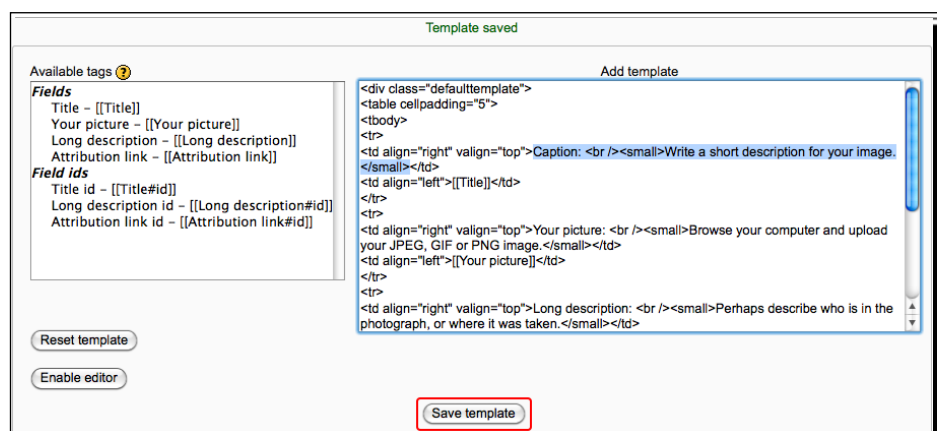
In a word, yes. Let's enhance the template for the **New entry** form.

## Time for action – editing templates

Return to the **Add template** tab under **Templates** for our database activity. We are going to use the editor to add some extra hints for each field:

1. Put your cursor at the end of the label **Short caption**, and insert a new line by hitting the *Enter* key.
2. On the new line enter your hint **Write a short description of the image.**
3. Now, make sure to click the **Save template** button.
4. Press the **Disable editor** button to the left.
5. In the resulting plain text area you will see a number of HTML tags, including `<table>`, `<tr>`, and `<td>`. These are the container elements for a table, a table row, and a cell respectively. You will also see phrases like **Short caption** in double square brackets—these are the placeholders in the template for the HTML form fields. Insert an opening tag `<small>` before the hint you have just entered (after the `<br />`) and a closing tag `</small>` after the hint.
6. Press the **Save template** button again.
7. Repeat the process for the other fields. You may want to leave the rich-editor disabled while you do this.
8. Press the **Save template** button. Your work will have resulted in something like the following screenshot.

As you can see there is a menu of **Available tags** to the left of the template editor. These are placeholders specific to the database module, not to be confused with HTML tags or elements.



## What just happened?

We edited the add entry template for the database activity to make it more usable and informative. There are also templates for viewing entries, searching, and so on. And we found that there are placeholders—names in square brackets that will be replaced by form fields when the activity is used. You can see the improved **New entry** form, with placeholders replaced and additional hints, in the following screenshot:

We have ensured that the **New entry** form is more usable. Let's move on and try it out.

## Adding an entry

The construction phase is complete, so now we come to adding a picture to our database. This will give your class the opportunity to hone their narrative, spelling, grammar, and language skills by creating a story in response to the picture.

## Time for action – adding an entry

Follow these steps to add an entry to the stories database:

1. Go to the **Add entry** tab in our database activity.
2. Find a photograph to upload. I found an attractive photograph captioned '**A Flower and A Frog**' by **dawnzy58** on Flickr (<http://flickr.com/photos/dawnzy/1120032008/>), available under a Creative Commons Attribution License. I saved it to my computer.
3. Enter the **Caption**, for example, **A Flower and A Frog**. Press the button labeled **Choose a file....** When the **File picker** dialog appears click on the link to **Upload a file**. Browse to the image and press **Upload this file** to close the File picker.
4. When you compose the **Long description** try to imagine what someone who can't see the photograph would need to know. This may include any factual information such as what the photograph contains. It may also include any thoughts or emotions that it invokes, though you can make this part of the story later on.



You may notice that the quote from the photographer, which was copied from Flickr, ends with the characters : ) . This is called an **emoticon** and is used to convey emotions in online communications. It is appropriate in less formal contexts, and can be useful for example, when you think something sensitive or personal may cause offence. It tries to take the place of the subtle eye and body communication, which is an intrinsic part of face-to-face contact. There are a number of commonly used emoticons, and there is a Moodle filter to convert the characters to an emoticon image. For example, : -D ; - ) : - ( are grin, wink, and sad respectively.



The screenshot shows the Moodle database entry form for a picture field. The form is divided into several sections:

- Navigation:** A sidebar on the left with links for 'Navigation' and 'Settings'.
- Image Preview:** A small image of a water lily flower and a green frog on a lily leaf in a pond.
- Caption:** A text field containing the URL 'flickr\_dawnzy\_1120032008\_m\_flowerfrog\_cc-by.jpg'.
- Alternative text:** A text field containing 'A Flower and A Frog'.
- Long description:** A rich text editor with a toolbar. The text area contains two paragraphs: 'I found this photograph from 2007 on Flickr. It shows a water lily flower and a green frog on a lily leaf in a pond.' and 'The photographer, dawnzy says "I was so set on getting a good shot of the flower I didn't notice the frog 😊".'
- Attribution link:** A section with a 'Url' field containing 'http://flickr.com/photos/dawnzy/1120032008/' and a 'Text' field containing 'otograph by dawnzy58, on Flickr [Creative Commons Attribution license]'. There is also a 'Choose a link...' button.
- Buttons:** At the bottom, there are two buttons: 'Save and view' (highlighted with a red box) and 'Save and add another'.

It is worth noting that the **Picture** field in the database activity incorporates a text field labeled **Alternative text**. This can contain the same short phrase that you put in the caption field—a small amount of duplication is not a problem.

5. When you have filled in the fields in the database entry press the **Save and view** button.
6. The image will be uploaded to the Moodle server, and you will be taken to the **View single** tab for the entry.

You may find that when you try to **View list** the page seems to again be broken. Try going to **templates – list template**, and hitting **Reset template** and **Save template**.

### ***What just happened?***

We found a suitable photograph, then used the add entry form to upload and describe the image in our new story telling pictures database. This is the start of a collection of images that will fire the imagination of our class.

We noted that the **picture field** incorporates an **Alternative text** field. This is an important feature for accessibility to those with disabilities.

We have an entry in our database, so we'll carry on and add comments to it.

## Commenting on the database entry

Now that we have an example picture, we will continue with the exercise by adding an example story. We do this using commenting. Commenting on another person's work is an important idea for your class to master.

### Time for action – adding a story as a comment

These are the steps to follow to comment on the entry:

1. Return to the entry you added to the **Story telling** database activity.
2. Scroll past the image and the **Long description** to the **Comment** text editor.
3. Enter the title of your story. In the Moodle 2 comment editor, surround the title with **heading 3** tags `<h3>` and `</h3>` (while in Moodle 1.9 select the text and make it a **Heading 3 using the rich-editor**).
4. Then, start typing your story in, inserting double line-breaks between paragraphs using the *Enter* key.



5. When you've finished, click on the **Save comment** link below the comment field.

### ***What just happened?***

As discussed previously, if we have comments enabled, you and your students can create the story in a comment. Potentially you and other class-mates can give feedback in a completely open manner. We presented a story by commenting on our own image.

Note that in Moodle 1.9 the comment field uses the full-fat rich editor. As you can see in the following screenshot, Moodle 2 just provides a basic text area for commenting.

This continues our example that we can show to the class to get them thinking.

Providing your class with opportunities to write descriptions and stories is important. In an increasingly multimedia world it is also instructive to try audio and video recordings. We are going to try our hand at an audio recording.

## **Installing a sound recorder**

Now that you have created your example story in Moodle, you may wish to either record an audio version of the story or some instructions for your class. Naturally, audio instructions would be in addition to written instructions. The first steps are to record the audio—this is often saved as WAVE form audio ( .wav ), which is an uncompressed file format. The **WAVE** file will be large (for example, 7 MB for 40 seconds of audio, depending on the audio bit rate and other settings). It needs to be converted to a compressed format for online use—currently Moodle supports **MP3** audio. There are two options to achieve these steps:

1. Use the basic sound recorder that is bundled as part of Windows (XP). This will create a WAVE file. Then use an online converter (Search: **online MP3 conversion**) to create the MP3 file. Note that the sound recorder does not have any editing capability, and this option will involve uploading the large WAVE files.
2. Install a desktop audio editor, such as **Audacity**, and an MP3 encoder like LAME. This will afford you full editing capabilities.

We will walk through option 2, as this is a solution that will work on various operating systems, including Windows and Mac OS X. It also provides the most scope for experimentation.

## Audacity and LAME

Audacity is free/open source desktop software that you will use to record and edit audio. It can be installed on the Windows, Mac OS X and Linux operating systems. Audacity requires LAME to convert the uncompressed audio files to smaller, compressed MP3 files for use on the Web.

### Time for action – installing Audacity and LAME

These are the steps to install Audacity on Windows Vista (similar steps will apply for Windows XP and 7):

1. Go to <http://audacity.sourceforge.net/> and follow the links to download the appropriate Audacity installer (EXE) for your version of Windows—`audacity-win-unicode-1.3.13.exe` for Windows 7 and Vista at the time of writing.
2. Click on the installer in the list of downloads from your browser. Press the **Run** button (Firefox). If you're satisfied you have the correct installer, choose **Allow** in the **User Account Control** dialog.
3. Select the language to use during installation, then click on the button labeled **OK**. Press **Next** on the **Welcome**, license information, installation destination, and additional tasks dialogs.
4. Press the **Install** button then wait as the installer runs. Click on **Next** and **Finish**.

Follow these steps to install Audacity on Mac OS X:

5. Go to <http://audacity.sourceforge.net/> and follow the links to download the Audacity universal binary (DMG)—`audacity-macosx-ub-1.3.12.dmg` at the time of writing.
6. Double-click the downloaded `.dmg` file to mount, or prepare it for installation.
7. Create a folder called `audacity` within your Applications folder.
8. Copy all the items from the mounted DMG into the `audacity` folder you created.
9. Eject or un-mount the DMG.

Here is the procedure to install LAME for Audacity on Windows (Windows 7):

10. Visit the links page on the LAME website, <http://lame.sourceforge.net/links.php> and read the terms and conditions. MP3 encoding is the subject of patents. You may be able to download and use LAME for free, but if in doubt do some research. (However, the author and publisher accept no responsibility for this.)

- 11.** Follow the link to <http://lame.buanzo.com.ar/>. If you are satisfied that you can, download the EXE installer, which will be named something like `Lame_v3.98.3_for_Audacity_on_Windows.exe` (420 KB).
- 12.** Launch the installer. When asked **Are you sure you want to run this software?**, choose **Run**.
- 13.** In Windows 7 you will see a dialog with the message **Do you want to allow the following program from an unknown publisher to make changes to this computer?** Select **Yes**.
- 14.** Click **Next** in the following step, which contains the text, **Welcome to the LAME for Audacity Setup Wizard**.
- 15.** Tick **I accept the license agreement**, then press **Next**.
- 16.** Keep the default **Destination Location**, and again press **Next**.
- 17.** The installer will do its thing. Upon seeing the text **Completing the LAME for Audacity Setup Wizard**, click **Finish**. Installation on Windows is now complete.

Follow these steps to install LAME on Mac OS X:

- 18.** Visit the LAME website, <http://lame.sourceforge.net> and read the terms and conditions on the **About** page. MP3 encoding is the subject of patents. You may be able to download and use LAME for free, but if in doubt do some research. (However, the author and publisher accept no responsibility for this.)
- 19.** If you are satisfied that you can, download the universal binary (DMG)—to do this you will probably be directed to <http://lame.buanzo.com.ar/>. The file will be named something like `Lame_Library_v3.98.2_for_Audacity_on_OSX.dmg`.
- 20.** Double-click the downloaded `.dmg` file to prepare it for installation.
- 21.** In the dialogue which pops up named **Lame Library...**, double-click on the `.pkg` installer file. Follow the on-screen instructions.
- 22.** If you have a permissions problem later, you may have to customize the **Destination**—click on the **Mac OS X** drive icon, press **choose folder...**, and change `/usr/local/lib` to (for example) `/Applications/audacity/lib`.
- 23.** Press **Choose**. Follow the on-screen instructions to complete the installation.

## ***What just happened?***

We downloaded and installed Audacity, a free audio recorder and editor on Windows and Mac OS X. And we installed the LAME encoder, which will help us convert our audio to compressed MP3.

Now we are ready to try our hand at recording.

## **Testing audio**

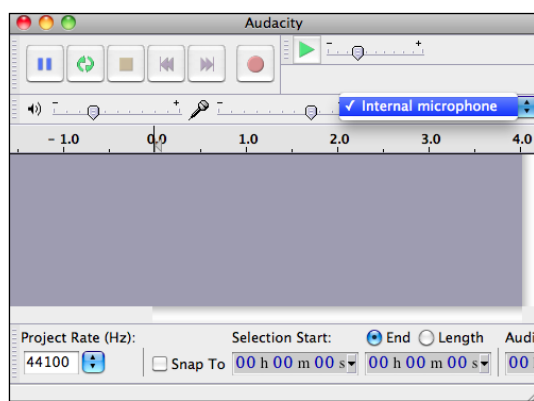
It's time to get a glass of water, clear your throat, and generally warm up your voice!

We are going to start by doing a short test recording, so that we can test recording levels and get a feel for things.

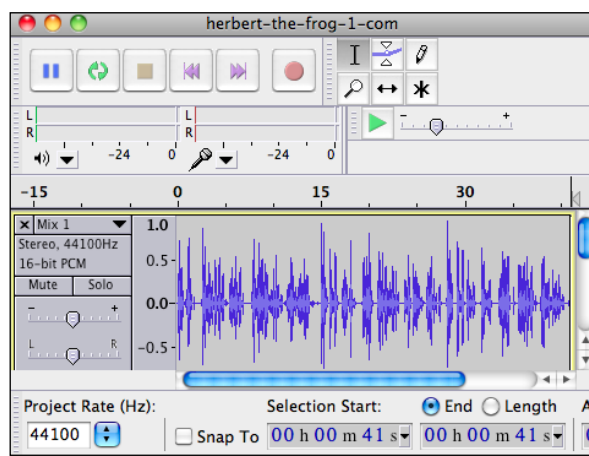
### **Time for action – performing a test recording**

Follow these steps to make a test recording:

1. Double-click on the **Audacity** application (Mac OS X) or shortcut (Windows) to launch it.
2. You will see something like the next figure. There are two elements to note. Look for the drop-down menu, which lists the internal microphone if one is present and other microphones. Select your input device now. And in the top-left you will see a series of colored buttons including ones to play, pause, and record audio—the button with the red dot is the record one.



3. If you are using a head-mounted microphone adjust it so that it is comfortable on your head. You will probably want to place a head-mounted microphone a little to one side of your mouth, to reduce the effect of vocal plosives such as "P" and "T".
4. Press record and speak a few words—this is your chance to imagine you're the sound engineer for your favorite band!
5. Press the stop button, and play it back. If the recording works, you should see and hear a result a bit like that shown in the following screenshot:



## ***What just happened?***

We installed Audacity and LAME. And we have performed a test recording to check the recording level.

If the test recording doesn't appear to work, here are some trouble-shooting steps to try:

1. Check that the microphone jack is plugged into the correct socket—on a PC this is usually the one colored in pink.
2. Check that you have selected the correct input device from the drop-down menu.
3. If recording still doesn't work and your computer has a built-in microphone try that.

Now that we have everything set up appropriately, on with the recording.

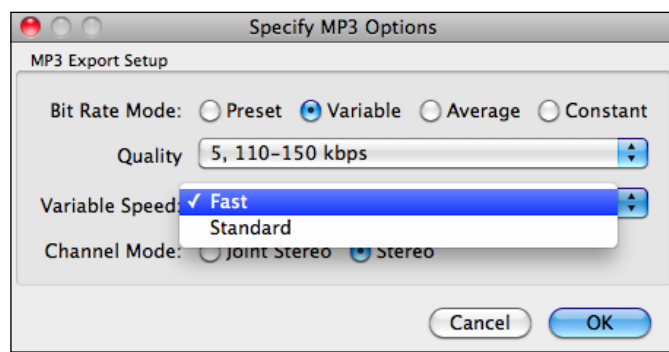
## **Recording audio**

Now that we have ensured that we can record audio let us try a full recording. We are going to record ourselves speaking the story we added to a comment previously.

## Time for action – recording audio

Follow these steps to record the story:

1. Choose **File | New** from the **Audacity** menu, to open a blank document.
2. Set up your script, which can just be the database entry with your story in your web browser, next to Audacity on your computer screen. It is then fairly straightforward to record our story.
3. When you finish, choose **File | Save Project**. You will be warned that you are saving an Audacity project file (.aup), which only Audacity can open. Do not worry, this is just a useful first step to ensure we save all the available data. Press **OK**, and choose a filename and location on your computer.
4. And now we can export our audio project to some reusable formats.
5. From the Audacity menu choose **File | Export....** In the resulting dialogue box choose **MP3 files** as the format.
6. Press the options button next to the format drop-down menu. In the dialogue box that pops up you can experiment with the MP3 options. I tried a variable **bit rate mode**, a **quality of 5, 110-150 kbps** (kilo bits per second) and a **variable speed** of fast. MP3 is a lossy compression algorithm, which means that like JPEG images, some quality can be sacrificed to produce a smaller file. Naturally the aim is to strike the appropriate balance between size and quality—as this is just speech we can aim for lower quality (a higher **quality** value). Press **OK** when you are done.



7. Type a file name and choose a location. Press **Save**.
8. You will be presented with an **Edit metadata** dialogue, and given the opportunity to fill in the **Track title** and so on. Enter as much or as little information as you wish. Nearly there!



9. Windows users can skip this step. As this is the first time we have used Audacity, the next stage on Mac OS X is a prompt to locate the LAME library we installed earlier. The default location that Audacity on Mac OS X looks in is `/usr/local/lib/audacity/libmp3lame.dylib`. I had a permissions problem with this location, and in the end put my LAME library in `/Applications/audacity/lib/audacity/libmp3lame.dylib`. When you have located LAME click on **OK**.
10. And finally, Audacity will use LAME to encode the audio as an MP3 file. You will be able to test the result in Windows Media Player on Windows, and in Audacity, Quicktime, and iTunes on either Windows or Mac OS X.

### ***What just happened?***

Having installed Audacity and LAME, we performed a test recording. After we had ensured that all was working, we recorded our story. We then stepped through the process of saving the audio, then exporting it as a compressed MP3 file.

## **Integrating an audio file with Moodle**

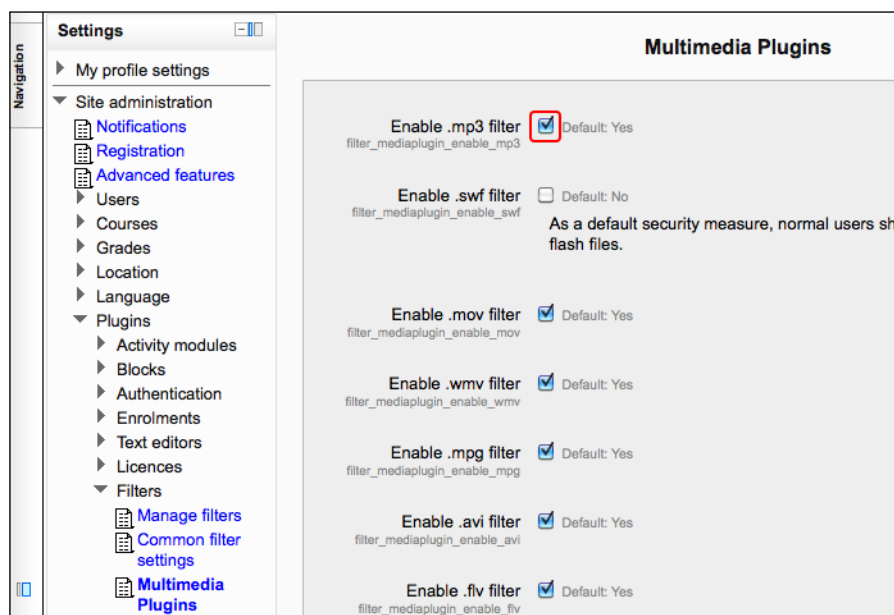
So we have an example MP3 audio file. Our next step is to upload the file to Moodle, and integrate it with our database activity. However, first we need to ensure that the multimedia filter is enabled in Moodle.

### **Time for action – enabling the multimedia filter**

One of the most useful filters packaged with Moodle is the multimedia filter. You will need to ask your friendly IT support person to enable it as you won't have permission to do this yourself. They can follow these simple steps:

1. Log in to Moodle as administrator.
2. Locate the **Site administration** section of the **Settings** side-block (the **Site Administration** block in Moodle 1.9). Expand **Plugins | Filters (Modules | Filters** in Moodle 1.9). Click on the link to **Manage filters**.
3. Ensure that **Multimedia plugins** is enabled—select **On** from the drop-down box in the **Active?** column (in Moodle 1.9 ensure the name is not grayed out and the eye icon is open).

4. Click on the **Settings** link and ensure that at least the **Enable .mp3 filter** checkbox is ticked. Scroll to the foot of the page and press the **Save changes** button.



### ***What just happened?***

The multimedia filter is a very useful plugin that is part of core Moodle. We enabled and configured the filter. The instructions above are for Moodle 2, with the minor variations for Moodle 1.9 noted in brackets.

And now, you may hand back to the teacher. Thank you!

## **Uploading audio files**

We have already uploaded image files using the rich-editor. Moodle also allows you to upload arbitrary files, and for reasons of privacy and security, these are stored separately for each course.



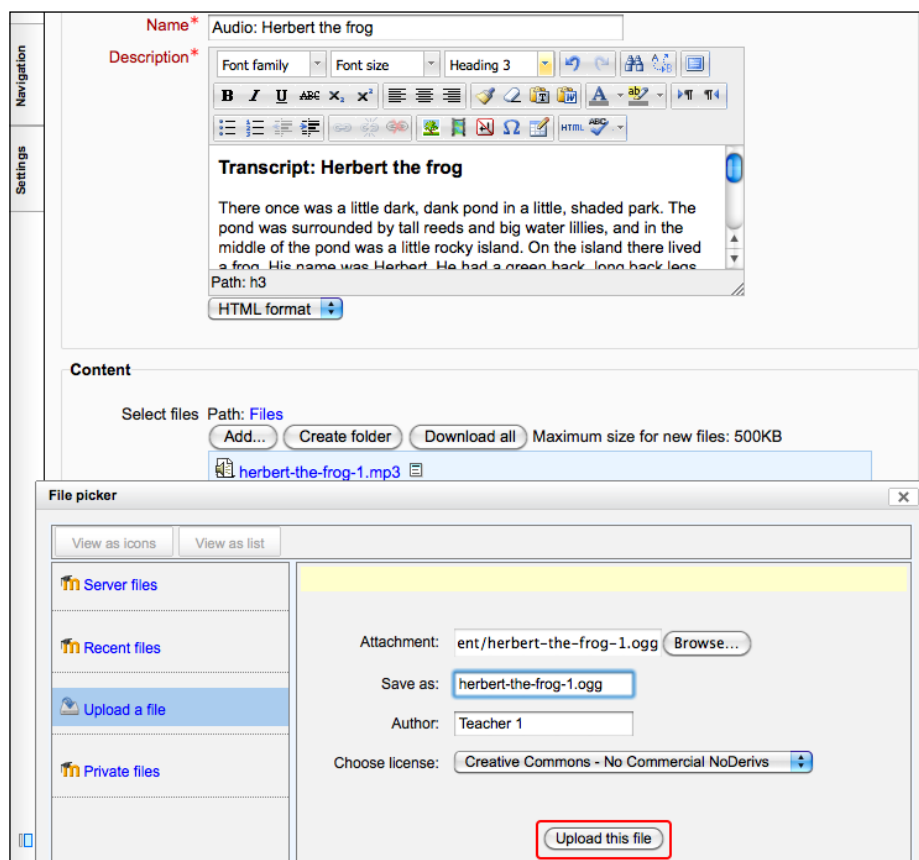
When we first came across Moodle's permissions system in *Chapter 1, Getting Started* we introduced user roles such as student, teacher, and course creator. The other side to permissions is **capabilities**. For example, there is the capability **moodle/course:managefiles**. By default, the non-editing teacher and student roles do not have this set—they cannot manage course files. The teacher and course creator roles on the other hand do have this capability set, so they can. There are approximately 230 capabilities built in to Moodle 1.9.x, giving fine grained access to all parts of Moodle and its plugins.

## Time for action – uploading files

In **Moodle 2**, follow these steps to upload your audio file:

1. In the same topic as our database (**Topic 1**), choose **File** from the **Add a resource...** drop-down menu.
2. In the form that appears give the file a **Name**. I typed **Audio: Herbert the frog**.
3. I copied and pasted the story from the **Database** entry into the **Description** rich-editor. You can prefix the **Description** with a phrase like **Transcript** (formatted as **Heading 3**), to indicate the purpose of the text.
4. In the **Content** section of the form, press the button labeled **Add...**
5. In the **File picker** dialog which appears, select **Upload a file** in the left-hand menu.
6. Press the **Browse...** button next to the **Attachment** label, and find the MP3 audio file you created in **Audacity** previously (`herbert-the-frog-1.mp3`). Press **Open** (or **OK**) to select the file.
7. Choose a **license** if you wish, or you may leave the default (**All rights reserved**). Then press the **Upload this file** button as shown in the following screenshot.

8. If you created an OGG format audio file, repeat steps 4 to 7 to upload the OGG file, as demonstrated in the next screenshot:



9. Under **Options**, keep the default **Automatic** for the **Display** drop-down menu.
10. Near the foot of the page, press the button to **Save and return to course**.
11. In the course main page, right-click with your mouse on the link for the uploaded file and choose **Copy link location** from the mouse context menu (Firefox). You will copy something like, `http://my.school/moodle2/mod/resource/view.php?id=13`.

In **Moodle 1.9** and earlier the steps are quite different.

Go to the course main page, and in the **Administration** side-block click the link **Files**. You will see a table. In the left column is the name of a file or directory, while the second and third columns contain the file size and modified date respectively. As we have already uploaded some images to our database activity module it should not be a surprise that there is a folder called **moddata**. If you click on the **moddata** directory-link, and follow the links down you should eventually reach a directory containing the image that you uploaded to your database entry. The precise numbers that make up the path (**moddata / data / 3 / 10 / 2**) will be different for you.

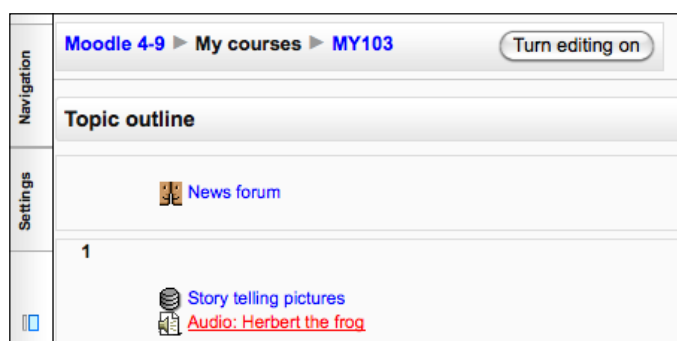
Continuing with Moodle 1.9, now that we have explored the file system for our course, proceed like this:

- 12.** It's time to go back to the **root**—follow the **Files** link in the breadcrumb trail.
- 13.** Press the button on the right, labeled **Upload a file**. You will be taken to a page with the familiar **Browse** button.
- 14.** Browse your computer for the MP3 file you created previously. When you have selected it click on **Upload this file**.
- 15.** You will be taken to the **File** index page. Hover over the link for your newly uploaded file. And in the context menu for the right mouse click press **Copy link location**.

## ***What just happened?***

We uploaded the audio file that we created in Audacity to Moodle. We found that the process was quite different for Moodle 2 and Moodle 1.9.

As we can see in the screenshot of our course main page below, **Files** are what we might term first-class objects in a Moodle 2 courses. Also note that a single **File** resource can contain multiple files, in our case the same audio track in MP3 and OGG Theora formats.



Now that we have uploaded our audio track, it's time to use it to enthrall our class in the story telling activity.

## Embedding our audio file

We are going to use the Multimedia plugin that was enabled previously to embed a player for our audio. We will be using the link that we copied to the clipboard in the last section.

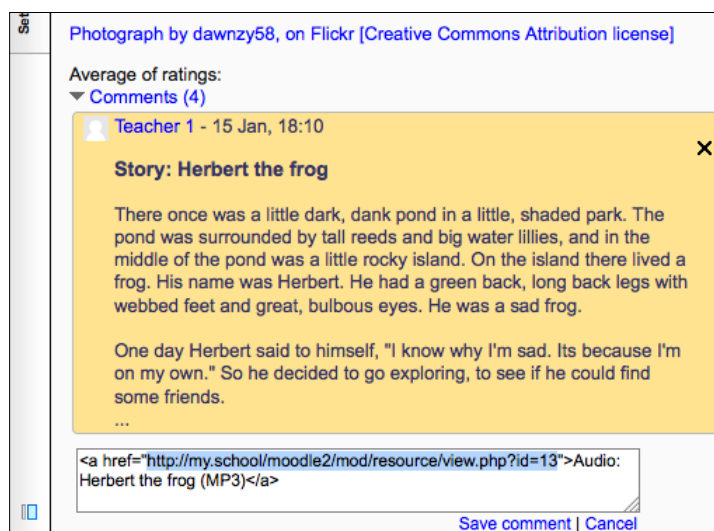
### Time for action – embedding audio

These are the steps to follow in Moodle 2 to embed an audio file in a database comment:

1. Return to your database activity, and choose the **View single** tab.
2. Scroll down and expand the **Comments** section. You will see your story in a comment, with a simple edit box below.
3. Type the following into the edit box, substituting the link you copied to the clipboard previously:

```
<a href= "http://my.school/moodle2/mod/resource/view.php?id=13"
>Audio: Herbert the frog (MP3) .</a>
```

4. Below the editor, press the **Save comment** button and view the result.

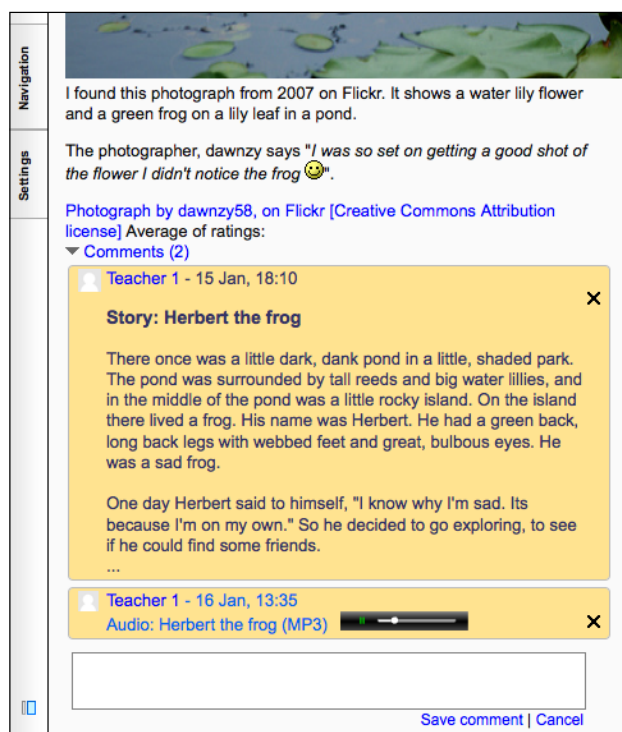


In Moodle 1.9 you should use the rich-editor on the comment field:

5. Press the **Insert web link** button in the rich-editor.
6. In the **Insert Link** dialog enter the link to the audio file, for example,  
<http://my.school/moodle/file.php/7/Herbert-the-frog-1.mp3>.
7. Click on **OK** to close the dialog. Press the **Save comment** button.

## What just happened?

In Moodle 2 you will see a link, which takes you to a page with an audio player generated by the MP3 multimedia filter. In Moodle 1.9, the multimedia filter will have automatically added an audio player after the link. (The player may use the Shockwave Flash plug-in for your browser.) Press the play button and check that it works. You will see the indicator move along the progress bar and hear the sound track as the audio file plays.



We uploaded the sound file that we recorded previously to Moodle. And we used the multimedia filter to embed an audio player for the file in our comment. The pupils will be able to read our example story, listen to the audio, and start to dream up their own stories.

## Running your lesson

So far we have used a comment to present a story—the example story from you the teacher. As explained at the start of this chapter, comments can be used in a variety of ways. You could ask your pupils to each add an entry with a picture to the database. Within the entry they can provide some narrative about the image. They can also write about their reactions to and interpretations of their class-mates' images as comments in the database. Another and quite ambitious approach would be to encourage your pupils to write the start of a story as a comment for someone else's image. Then they can explore the part-finished stories, and write their own endings for someone else's beginning, as a separate comment. You will be able to view each pupil's contributions via their profile, and rate them.

The online elements of this activity can of course be mixed with class discussion and face-to-face collaboration.

We will learn later in the book about ways to set formal assignments, in which case the picture database would simply be a starting point, with pupils not commenting or collaborating. This may be safer if you think plagiarism could be a significant problem or the comments may be misused.

However you use this activity, learning outcomes for the students may include:

- ◆ ICT: Uploading images, filling in online forms, writing appropriate comments, and feedback
- ◆ Working together: Collaborative story-telling
- ◆ Describing what you see, literacy, narrative

## Have a go hero

In this chapter, we created a database activity. We uploaded an audio file, containing our example story for the class to listen to. Older children would learn valuable skills by recording an audio version of their own story. Modify your database activity so that your class can upload an MP3 file as part of their database entry. What points and limitations would you have to consider?

The answer is in the database activity, choose the **fields** tab. Create a new **file** field. Enter a name and description. Then modify the **add entry** template. Limitations: you will probably need to increase the upload size limit for the course, from 500 KB to 1 or 2 MB per upload. It is probably worth discussing space limitations with your friendly IT support person. And, (as of Moodle 1.9.7) the MP3 filter is not applied to a file field, so you could add a link to the MP3 file in the long description. The filter would be applied to that. Other considerations: naturally you will have to gauge if your class is ready to take on audio recording and editing. And you may have to modify the format of the activity slightly.



## Pop quiz

Try this quick quiz. There may be more than one correct answer:

1. What are the purpose(s) of templates in the database module?
  - a. They control the appearance of forms to add entries.
  - b. They specify the appearance of database views.
  - c. They specify which fields are in a database.
2. If the database add entry form appears corrupted, what should you do?
  - a. Delete and re-create the database field in question.
  - b. Just put up with it.
  - c. Note any customizations made to the add template, then reset the template.
3. What is the purpose of LAME?
  - a. It is software to record and edit audio files.
  - b. It can encode audio in the compressed MP3 format.
  - c. It uploads files to the Internet.

## Summary

We learned a lot in this chapter about using the database module.

Specifically, we covered:

- ◆ What the database activity is, and the potential of commenting
- ◆ Specifying the fields for a database activity
- ◆ Editing database templates to enhance usability
- ◆ Adding entries and comments—and possible collaboration
- ◆ Recording and encoding audio files
- ◆ Uploading files, and using the multimedia filter

We also discussed emoticons and role capabilities.

Now that we have learned about database activities, and introduced collaboration it is time to explore the notion of groups and further collaborative activities. This is the topic of the next chapter.



# 4

## Spot the Difference

*In this chapter, we are going to use the Moodle lesson activity module to create spot-the-difference visual exercises. Younger children find spot-the-difference activities fun, and they are a great way to develop the visual, observational, and descriptive skills. We will also explore student enrolment.*

In this chapter we will:

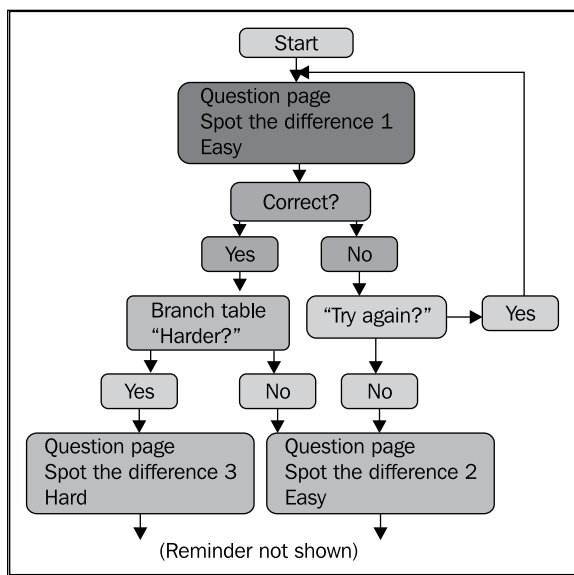
- ◆ Introduce the lesson module and its potential
- ◆ Design and create a lesson activity
- ◆ Find suitable images to use as the basis for our spot-the-difference activity
- ◆ Install the Inkscape desktop image editor on Windows
- ◆ Edit our images
- ◆ Add question pages to our lesson, and images to the question pages
- ◆ Add branch tables to our lesson
- ◆ Enroll students on our course, using different methods

So let's dive in...

## Introducing the lesson module

The *lesson* activity module shares some similarities with the *quiz* module, in that it gives the teacher the ability to set questions for the pupil to answer. However, there are some differences, notably:

- ◆ While the quiz module is often used for assessment, the lesson module as the name implies is primarily for learning. That being said, the teacher can decide whether or not to add the score from either to the *gradebook*.
- ◆ There is a wider choice of question types for the quiz and third-party question types can be added. In Moodle 1.9 and 2, a subset of the core question types is available in the lesson activity. This still affords us plenty of choices.
- ◆ The quiz module restricts the student to a linear path through a set of questions, with the outcome of one question not influencing the next. On the other hand, the lesson module allows the teacher to design in looping and branching, where the choices the pupil makes can result in different paths.



The previous diagram illustrates the flow through the first part of an example lesson activity. We will use this as the design for our spot-the-difference activity.

The flow is from top to bottom, and the four green boxes represent pages added to the activity by the teacher. There are three question pages, starting with an easy spot-the-difference question, and optionally moving to a harder question. If the pupil answers the first question correctly, as determined automatically by the system (the pink boxes), they see a branch table. This allows the pupil to choose an easier or harder question based on how well they think they coped. Alternatively, if the pupil answers the first question incorrectly then the system allows them to loop and repeat the question.

For simplicity, we are not going to consider the remaining action or page types available in the lesson module, but in brief they are as follows:

- ◆ **Cluster** – this is used to group a number of questions
- ◆ **End of cluster** – this is a hidden page that acts as a redirect to another part of the lesson
- ◆ **End of branch** – this is an optional hidden page at the end of a branch, to provide a jump to another part of the lesson

## Creating our first lesson activity

Start by creating a new course in the usual way, with a short name of **MY104**. We will launch in and add a lesson activity.

### Time for action – adding a lesson activity

To add a lesson activity, follow these steps:

1. Select **Lesson** from the **Add an activity** drop-down menu in your course.
2. The system will present a form like that shown below. Enter a lesson **Name**, for example, **Spot the difference lesson**.
3. Also in the **General** section, increase the **Maximum number of answers/branches** from 4 to 5.
4. Then under **Grade options**, set **Practice lesson** to yes – this means that the activity will not show up in the gradebook. Leave the defaults for the other items in **Grade options**.
5. In the **Flow control** section choose **Yes** from the drop-down menu for **Allow student review**. In Moodle 1.9 choose **Yes** for **Display review button**.
6. Increase the **Maximum number of attempts** from 1 to 2. You should then leave the defaults for the rest of the form fields. As ever, you will be able to return and adjust these values later should you wish.

7. Scroll to the bottom and press the **Save and display** button.

The screenshot shows the 'General' settings for a lesson in Moodle. The settings are organized into three sections: General, Grade options, and Flow control. In the General section, the 'Name\*' is 'Spot the difference lesson', 'Time limit (minutes)' is 20, and 'Maximum number of answers/branches' is 5. In the Grade options section, 'Practice lesson' is Yes, 'Custom scoring' is Yes, 'Maximum grade' is 0, 'Student can re-take' is No, 'Handling of re-takes' is Use mean, and 'Display ongoing score' is No. In the Flow control section, 'Allow student review' is Yes, 'Display review button' is Yes, 'Maximum number of attempts' is 2, 'Action after correct answer' is Normal - follow lesson path, and 'Display default feedback' is No.

| General                            |                            |
|------------------------------------|----------------------------|
| Name*                              | Spot the difference lesson |
| Time limit (minutes)               | 20                         |
| Maximum number of answers/branches | 5                          |

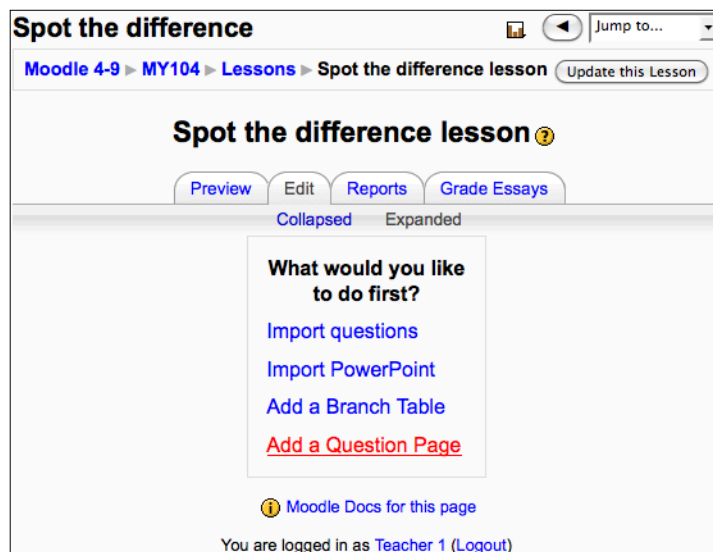
| Grade options         |          |
|-----------------------|----------|
| Practice lesson       | Yes      |
| Custom scoring        | Yes      |
| Maximum grade         | 0        |
| Student can re-take   | No       |
| Handling of re-takes  | Use mean |
| Display ongoing score | No       |

| Flow control                |                             |
|-----------------------------|-----------------------------|
| Allow student review        | Yes                         |
| Display review button       | Yes                         |
| Maximum number of attempts  | 2                           |
| Action after correct answer | Normal - follow lesson path |
| Display default feedback    | No                          |

The next page is the lesson dashboard, which comprises four tabs. The **Edit** tab will be initially selected. Under the question **What would you like to do first**, there are five options:

- ◆ **Import questions**
- ◆ **Import PowerPoint**
- ◆ **Add a content page** (labeled, more accurately, **Add a Branch Table** in Moodle 1.9)
- ◆ **Add a cluster**
- ◆ **Add a Question Page**

According to the flow chart we looked at previously, we want to start by adding a question, so click on the final link.



Then follow these steps to add a question page:

1. In Moodle 2, you will be presented with an intermediate page containing a drop-down menu labeled **Select a question type**. Choose **Multichoice** and press the **Add a question page** button. In Moodle 1.9, you are presented with the main **Add a question page** form, with a row of tabs for the available question types at the top. You should choose the **Multichoice** tab then proceed as for Moodle 2.
2. For our first question enter a **Page title** such as **Spot the difference 1**. Tick the **Multiple-answer** checkbox (in Moodle 1.9, this is at the top of the form, and in Moodle 2 under **Page contents**).
3. Next to **Page contents**, type **Take a good look at the two pictures below**. After a newline, type **How many differences can you spot? Tick the boxes that apply**. Select this line with your mouse and make it a **Heading 3**, as shown in the next screenshot.
4. In Moodle 2, you must set at least two answers, so enter placeholders for **Answer 1** and **Answer 2**. I typed **[An answer.]** and **[Another answer.]**, respectively.

5. Scroll to the bottom of the form and press the **Add a question page** button.

The screenshot shows a web-based question editor. At the top, there are tabs for different question types: Multiple Choice, True/False, Short Answer, Numerical, Matching, and Essay. The 'Multianswer' tab is selected. Below the tabs, there is a 'Page title' field with the text 'Spot the difference 1'. The 'Page contents' section has a rich text editor with a toolbar showing various formatting options. The text in the editor reads 'Take a good look at the two p' followed by 'How many differences c'. A dropdown menu is open, showing options for heading levels (Heading 1 through Heading 6), Normal, Address, and Preformatted. Below the editor, there is a 'Path' field showing 'body > h3'. At the bottom, there are fields for 'Answer 1:' and 'Response 1:'.

### ***What just happened?***

There are six types of question available in the lesson activity, and only a few of these are interesting for our spot-the-difference activity.

Using the **Multiple choice** type we can ask questions such as, how many differences can you see between the images? The answers would comprise a range of different numbers, which may be easy to guess and are not very interesting. We can also use the question type with the **Multianswer** checkbox ticked, and a phrase such as, **Tick the boxes for the differences that you can spot**. If you scroll down you will see that we can have up to five correct and incorrect answers to a multiple choice question, so this is only useful if there are three or at most four differences between the images. We can also use the **Short answer** question type. As we shall see with careful design it is possible to automatically check answers for this question type.



The multiple choice question type guides the pupil through the exercise so it is suited to the first question, while the short answer question requires more thought from the pupil so it is better for later questions.

We created a course and added a lesson activity to it, to form the basis of our spot-the-difference exercise. We configured some parameters related to the lesson flow. And after some discussion about how to apply different question types to our exercise, we added our first question page. In the next section, we will find and edit some images for the exercise.

## Finding spot-the-difference images

In order to create the spot-the-difference exercises we are going to need pairs of similar pictures. So the first question is whether we can find suitable images on the Internet. And, the next question would be, how do we edit them?

The **Open Clip Art Library (OCAL)** is a community project started by Jon Phillips and Bryce Harrington in 2004 to publish clip art. Clip art is normally used as components in other works. It can range from obviously synthesized or cartoon-like images to photorealistic images.

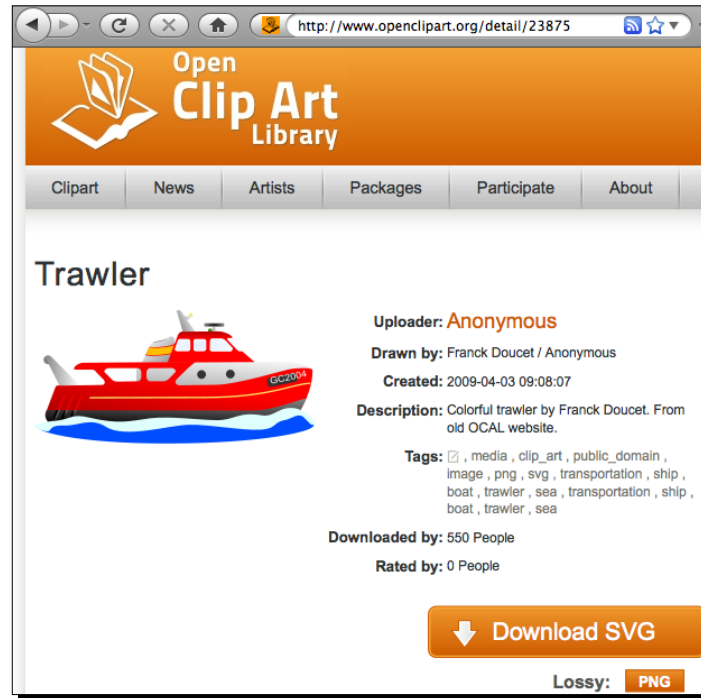
### Time for action – finding images online on OCAL

Visit the home of the Open Clip Art Library at <http://www.openclipart.org/>. We would like three or four images for this exercise, and we should be able to find them using a combination of search and browsing. Some keywords you can try are **man** (or **person**, **people**), **house**, **animal**, and **boat** (or ship).

I found the following four images, but of course you can choose whatever you think is most suitable:

- ◆ Dawg (by feraliminal)
- ◆ Man in suit (by Gerald G/ [worldlabel.com](http://worldlabel.com))
- ◆ Nicu's house (crash cghost) (by Nicu and rejon)
- ◆ Trawler (by Franck Doucet)

In the following screenshot you can see the details page for the trawler. In each case, download the SVG image and save to your hard drive. (Note, we don't want the PNG image at this stage!) I chose names like `openclipart-AUTHOR-FILE.svg` (for example, `openclipart-Gerald_G_Man_in_Suit.svg`).



There are some things worth noting about the Open Clip Art Library:

- ◆ There are a wide range of categories and a lot of interesting images in the OCAL collection.
- ◆ All Open Clip Art Library contributions are in the **Public Domain** (scroll to the bottom-left of a page on the OCAL site to view a link for the Creative Commons License). This means that the author has dedicated the work to the public domain, and does not claim any copyright on it.
- ◆ The primary format for OCAL images is **Scalable Vector Graphic (SVG)**, which is an open-standard based on XML. The individual components or layers are fairly easy to edit in the SVG image format, so as we will see it is well suited to our needs.



The **World Wide Web Consortium (W3C)** wrote the Scalable Vector Graphic standard (SVG 1.1). Vector graphics differ from raster or bitmap formats such as JPEG, GIF, and PNG in that the image file contains descriptions for components such as circles and polygons, by specifying their position, dimensions, colors, and so on. One benefit of vector graphic formats such as SVG is that they can be resized without the pixels becoming visible.

The W3C is a standards body for the Internet. It is led by Tim Berners-Lee, who wrote the first Web server and browser. It has developed many of the standards we are using in this book, including HTML, CSS, and XML (Hyper-Text Markup Language, Cascading Style Sheets, and Extensible Markup Language respectively). You can read more about the W3C's important work at <http://w3.org/>.

Scalable Vector Graphic images can be displayed natively in a range of browsers, including Mozilla Firefox, Opera, Google Chrome, and Safari (and Microsoft Internet Explorer 9, in Beta at the time of writing). And, they can be converted to the ubiquitous PNG image format.

### ***What just happened?***

We found a number of images in the Open Clip Art Library for use in our spot-the-difference exercise. And we introduced the SVG image format. We will find SVG images fairly easy to edit and well suited to this exercise.

## **Installing an image editor**

As SVG is based on XML, simple images can be edited in a text editor. However, to edit images such as the fishing boat shown previously we will want some dedicated software. Inkscape is an open source desktop SVG editor for Windows (Windows XP/Vista/7), Mac OS X, Linux, and Unix-like systems (FreeBSD).

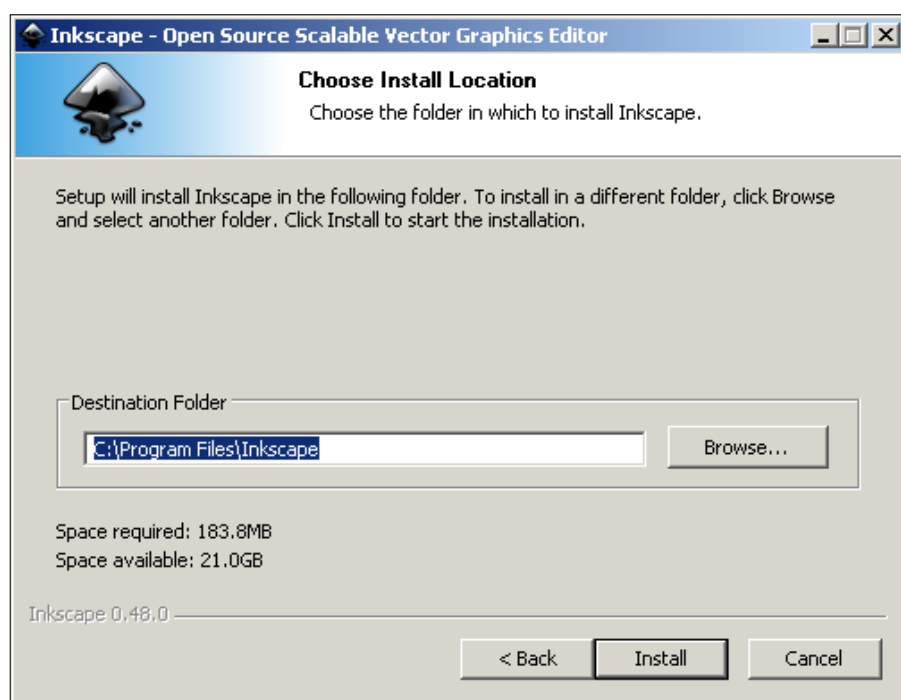


If you are already fairly comfortable with other image editing software you may be able to achieve similar results using them. However, it is not difficult to get started with Inkscape, so it's worth giving it a try. You have the choice!

## Time for action – installing Inkscape

We are going to walk through downloading and installing Inkscape on Windows. The instructions are similar for Mac OS X—look for the Universal Binary (DMG) installer file.

1. Search for Inkscape, for example using Google. You will probably be directed to `Sourceforge.net` to download the installer.
2. When prompted by your browser, save the installer file to your hard drive. Its name will follow the pattern `Inkscape-MAJOR.VERSION-MINOR.exe` (`Inkscape-0.48.0-1.exe` at the time of writing).
3. Double-click on the installer to run it. In the first dialogue window you will be asked to select the setup language from the drop-down menu. Press **OK**, then take a quick look at the GNU General Public License and press **Next**.
4. You will probably want to keep the default settings when you are prompted to choose the components and destination folder, as shown in the following screenshot. Press **Next** then **Install** to set up the program files on your computer.
5. At the final step, uncheck the **Run Inkscape** checkbox and click **Finish** to exit the install wizard.



## ***What just happened?***

In this section, we downloaded and installed Inkscape on Windows. Inkscape is an SVG editor. This will be useful to help us edit the images for our spot-the-difference exercise. And this is the topic of the next section.

## **Editing the images**

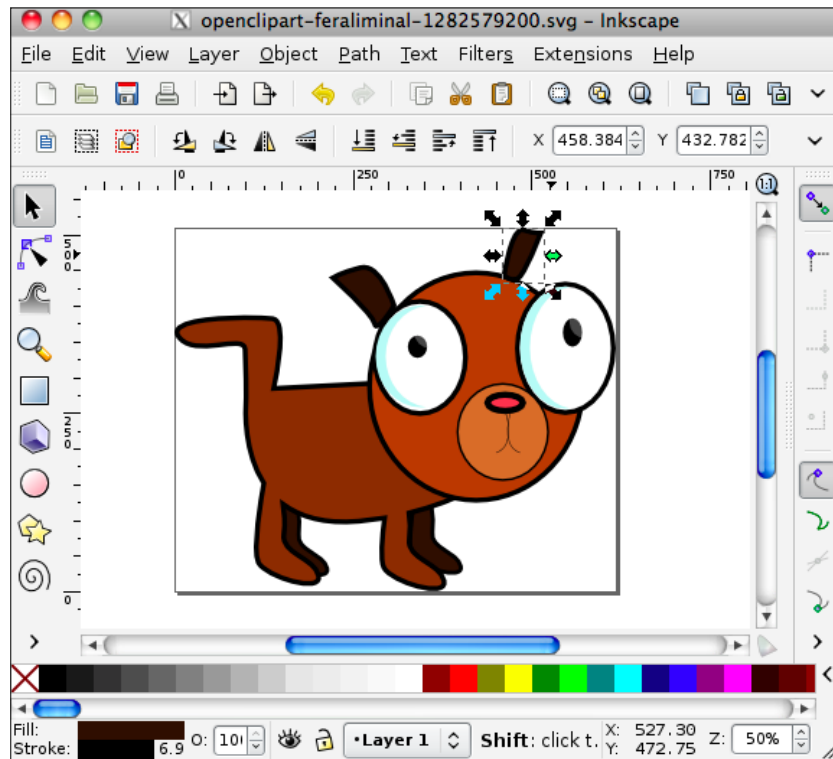
It's time to try our hand at editing the images for exercise. Let's dive in!

Find one of the SVG files on your hard drive—I will demonstrate Inkscape using the Dawg image that we downloaded previously. On Windows and Mac OS X, Inkscape will be associated with files with the `.svg` extension, so double-click on the file to launch the editor. You will be presented with a window such as the one shown in the next screenshot.

First, let's orientate ourselves:

- ◆ The name of the image is displayed at the very top.
- ◆ Below the title bar is a fairly standard menu containing items such as **File**, **Edit**, and **Help**. **File** contains familiar actions such as **New** (image), **Open** image, **Save** image, and **Quit**.
- ◆ In the middle of the window is our image.

- ◆ Around the image are a number of toolbars, including a palette below the image, and a set of tools to the left. The **Select and transform objects** tool is active initially. Click on an item in the image with your mouse, and a box and arrows will surround it. You can see that I have selected an ear in the following screenshot:



Do not worry if Inkscape seems daunting at first. The editor will become more familiar with a little use.

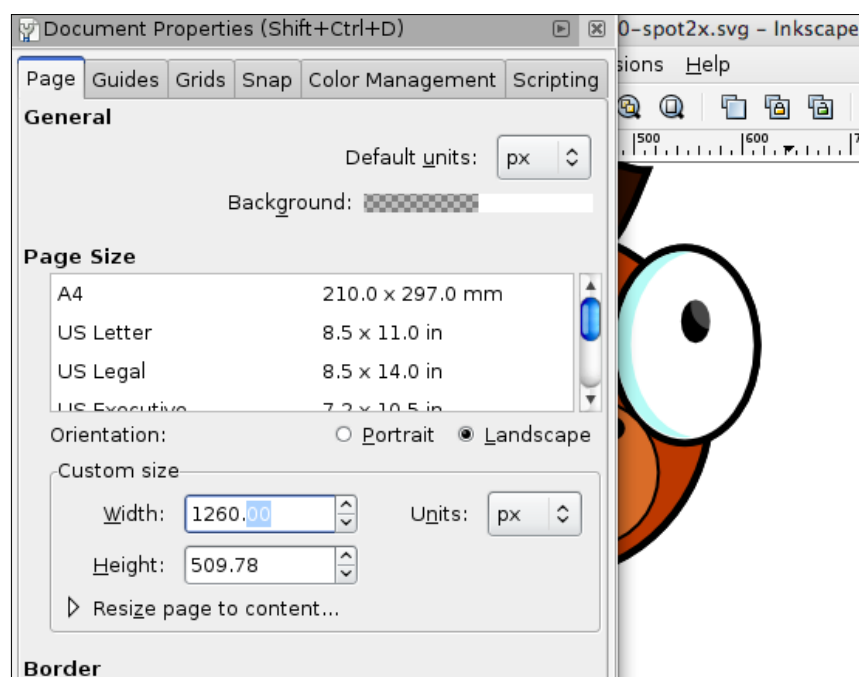
## Time for action – editing the first SVG image

We will create our pair of spot-the-difference pictures side-by-side in the same image, which will reduce the number of images that we need to manage and allow us to keep track of the differences.

Follow these steps:

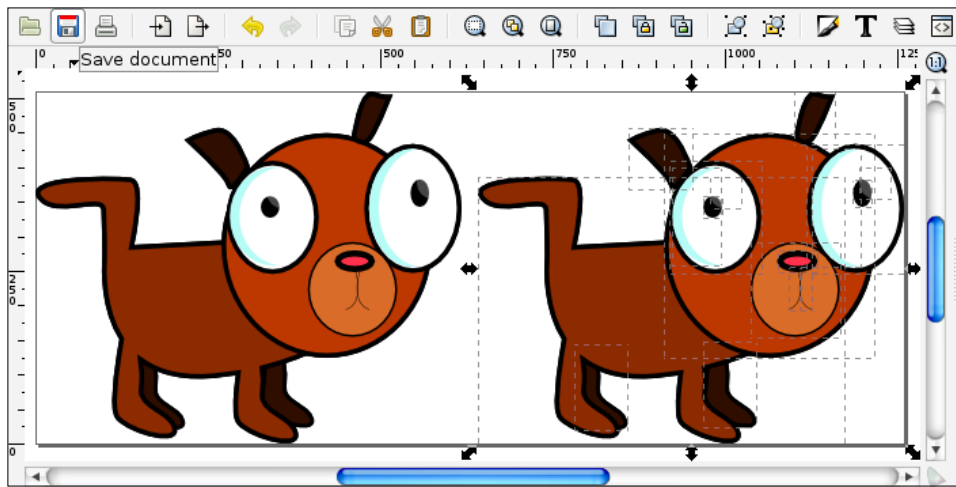
1. In the Inkscape editor menu bar choose **File | Save As** and add a suffix to the filename. I changed `openclipart-feraliminal-1282579200.svg` to `openclipart-feraliminal-1282579200-spot2x.svg`.

2. In the menu press **File | Document Properties...** and look in the resulting dialog window for the width. The original Dawg is 618.14 pixels wide, so double this number then add between 20 and 30 pixels. I ended up with a width of 1260 pixels. Press *Enter* so that the change takes effect, then close the dialog window. You will see that the shadowed box around the image has roughly doubled in width.



3. In the main Inkscape window press *Ctrl + A* to select all the layers in the image
  - Alternatively, use your mouse to drag a box over the image—make sure you select all as shown in the next screenshot.

4. Press **Ctrl + D** on the keyboard or in the menu bar press **Edit | Duplicate** to replicate all layers. Then immediately use your right arrow key to start moving the duplicate to the right. If you get something wrong don't worry—just press the green **Undo last action** (left-arrow) button in the main tool bar.

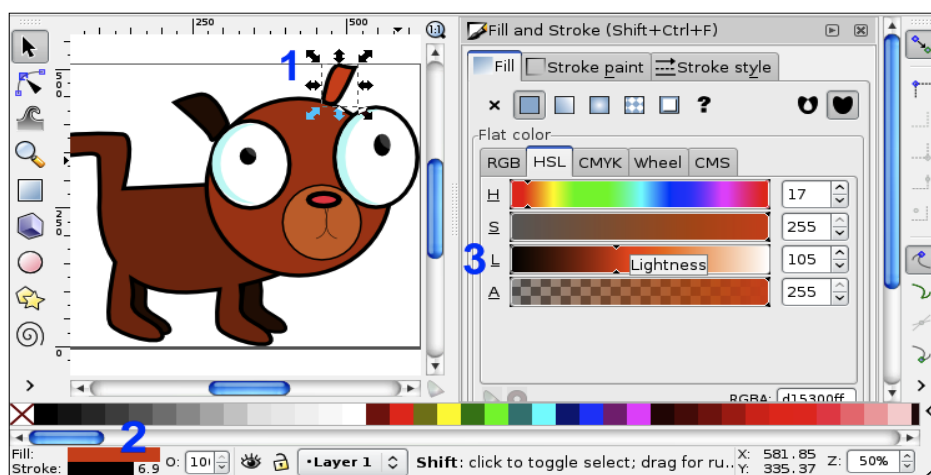


Well done! This is a good opportunity to press the **Save** button. You may wish to zoom in, so in the menu bar press **View | Zoom | Zoom In**. We will continue, by editing the copy of Dawg on the right.

5. Select one of Dawg's ears with your mouse in the new image (point 1 in the next screenshot). Scroll down to the foot of the Inkscape window and double-click on the brown rectangle next to **Fill** (point 2).
6. The **Fill and Stroke** panel will appear on the right, as shown at point 3 in the next screenshot. There are three tabs at the top of this panel, **Fill**, **Stroke paint**, and **Stroke style**. **Fill** should be selected. Below the tabs, the **Flat color** button should be selected.



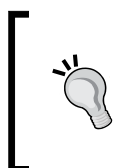
7. Beneath the buttons is a further row of tabs labeled **RGB**, **HSL**, **CMYK**, **wheel**, and **CMS**. These abbreviations denote different ways of visualizing colors and bear some explanation:
  - ❑ You may be familiar with **RGB** or **red-green-blue**.
  - ❑ The next tab, **HSL**, or **hue-saturation-luminance**, is useful, once you understand it. The hue slider allows you to set the tone or color on a continuous rainbow scale. The saturation slider controls the amount of color against the amount of gray in the fill. Luminance is the strength (or lightness) of the color and varies between black and white. Finally the **A** or **alpha** slider controls opacity and is common to all the tabs.
  - ❑ **CMYK** or **cyan-magenta-yellow-K** is the complementary color space to RGB.
8. In the **HSL** tab, move the **L** or luminance slider to the right to lighten the color.
9. Press the **Save** button at the top of the Inkscape window. And there we have our first difference!



We can go on to adjust the color of some other elements. We can also add and remove components. To remove a leg (poor Dawg), select it with your mouse and press the *Delete* or *Backspace* key. And, to duplicate an item, select it with your mouse, and press *Ctrl + D*. As before, the object will be pasted on top of the original. You may then move it using the arrow keys.

When you are editing your images, there are two questions to bear in mind:

- ◆ Can the pupil use one or at most two simple keywords to describe a difference between the pictures in an unambiguous way? Good keywords are things such as front leg, eyebrow, and curtain. Poor choices include left leg and the door lintel is different. The term left may create confusion—relative to whom? And lintel is a fairly advanced word, without an obvious replacement.
- ◆ If the pupil is asked to describe the differences from top to bottom, or possibly left to right, can they do so? This will be significant in formulating clear questions. So, ensure that objects that contain differences are clearly at different heights.



It is desirable to make the spot-the-difference reasonably accessible to any pupils who may be color blind. A useful rule is if we adjust the hue we also change the luminance or brightness of the color. This will ensure that the differences are perceivable to the maximum number of pupils. You can find out more about accessibility in the appendix.

Dawg is a fairly simple picture, and I made only three changes to this image. This makes it suitable for the first question page in our exercise.

When you are happy with the new image save the result and then:

1. From the Inkscape menu choose **File | Export Bitmap...**
2. In the resulting dialog window ensure that the **Drawing** button is selected, as we don't wish to export just a selection or the whole page. The **Bitmap size** will be set for you. The **Filename** including the destination folder will be the same as for the SVG original, except that the extension will be **.png**.
3. Press the **Export** button to create the cross-browser PNG image ready for uploading to Moodle.

## ***What just happened?***

We used Inkscape to create a copy of the original Dawg SVG image. We then explored how to select and edit components of the image. We tried editing colors and adding and removing components.

I went on to make between four and seven changes to the other three images. This will allow us to create spot-the-difference pages with a range of difficulties. The resulting exercise will provide a challenge for pupils with a range of abilities. You will end up with four SVG images and four exported PNG images. Each image file will contain a pair of pictures.

## Bringing it together

We have the images for our spot the difference exercise, so it's time to return to the lesson we started creating. But first, we need to upload our images.

### Time for action – adding images to our lesson

Follow this procedure in Moodle 1.9 and 2:

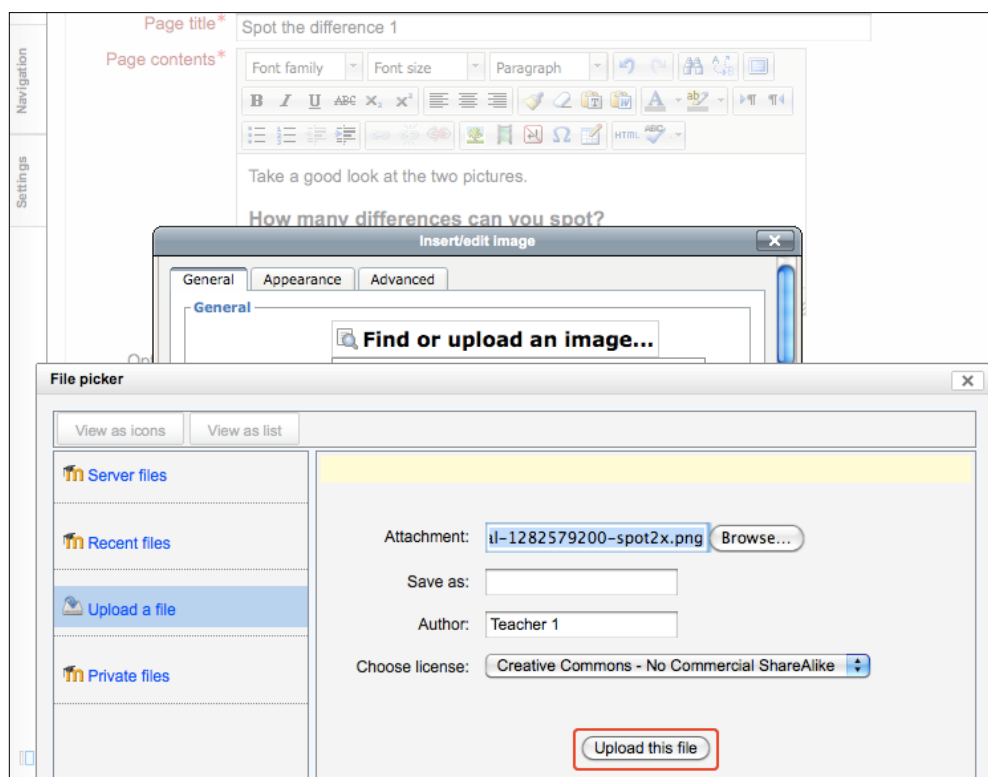
1. Go to the main page for the **MY104** course, and click on the link for the **Spot the difference** lesson.
2. You will be taken to the **Preview** tab, so choose the **Edit** tab and press the **update** icon next to the **Spot the difference 1** question.
3. Select the **Multiple choice** tab if it isn't already selected. And ensure that the **Multianswer** checkbox is ticked.
4. In the **Page contents** editor add a new line between the two existing sentences. Press the insert image icon as shown:



Continuing in Moodle 2:

5. In the **Insert/edit image** dialog press the **Find or upload an image...** button. Press the **Upload a file** button in the left menu of the **File picker** dialog.
6. Press the **Browse...** button, and find the Dawg PNG image on your computer. Returning to the **File picker** in your browser, choose a license if you wish. Press the **Upload this file** button, as shown in the next screenshot.

7. You will be returned to the **Insert image** dialog, where you should enter a suitable image description—I put **Dawg, times 2, from the Open Clip Art Library**. You may need to reduce the width and height, which you can do in the **Appearance** tab of the **Insert image** dialog. For Dawg, I halved the width from 1260 to 630 pixels and the height to 255 pixels. Press **OK** to insert the image.



These are the steps for Moodle 1.9:

8. Near the bottom of the **Insert image** dialog, press the **Browse...** button.
9. Press **Upload**, then fill in the **Alternate text** and edit the **Size** as described previously.  
Continuing, we will set the answers for our question. These points apply to Moodle 1.9 and 2:
10. Scroll down the form, and enter the three correct answers—right ear, nose, and front leg.

11. For each correct **Answer**, give a **Response**, for example **Well done**, a **Score** of 1 and for the present set the **Jump** drop-down menu to **Next page**.
12. Then enter some incorrect differences for the two final answers—I typed eye and tail. Set the **Response** to **Woops, no**, the **Score** to 0, and the **Jump** to **This page**. Note that the answers will be shuffled before they are presented to the pupil.
13. Scroll to the foot of the form and press the **Save page** button. You will be taken to a summary page. Press **Preview** to see the student's view. And give it a try.

**Multianswer:** ☒ ?

**Page title:**  
Spot the difference 1

**Page contents:**  
Trebuchet

Take a good look

What difference

Path: body » p

**Answer 1:** [Use editor: ☐ ?]  
Right ear

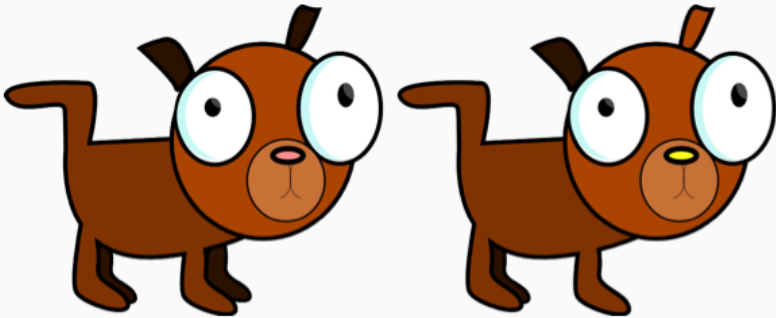
**Response 1:** [Use editor: ☐ ?]  
Correct!

**Jump 1:** Next page ? **Score 1:** 1

## ***What just happened?***

We uploaded our spot-the-difference images in PNG format to the Moodle course. We then went to the question page in the lesson, and added the first image. We went on to add correct and incorrect answers, and started to configure the lesson flow.

Take a good look at the two pictures below.



What differences can you spot? Tick the boxes that apply

|                                     |           |
|-------------------------------------|-----------|
| <input checked="" type="checkbox"/> | Eye       |
| <input checked="" type="checkbox"/> | Nose      |
| <input type="checkbox"/>            | Right ear |
| <input type="checkbox"/>            | Front leg |
| <input type="checkbox"/>            | Tail      |

Please check one or more answers

## **More difficult exercises**

We have a lesson containing a single question. To take our spot-the-difference exercise further we will want to try out different question types.

## Time for action – adding more questions

In order to take our lesson activity further we will need two more questions. Return to the lesson's **Edit** tab, scroll to the foot of the page, and click on the link, **Add a question page here**. This will add the new question after our first one. Prepare the picture named **Man in Suit** for the second question. I created four differences, so this will still be a fairly easy exercise. We can use the same **Multiple choice** question type, with the **Multianswer** checkbox ticked.

After you have added the second question, again go to the **Edit** tab. Scroll to the foot of the page and click **Add a question here**. For the third question choose the **short answer** tab below the **question type** label. Then follow these steps in Moodle 1.9 and 2:

1. Type in a **Page title**, for example **Spot the difference 3 – house**.
2. In the **Page contents** editor, type the text **Take a good look at the pictures below**. Insert a line break.
3. In the editor, press the **Insert image** icon. Choose the image based on **Nicu's house**. Enter a suitable Description/Alternate text, and adjust the width and height using the original ratio. When you are happy, press **OK**.
4. Insert another line break. Enter the phrase, **Start from the bottom and work up. Can you describe the differences?** Now, select this text and use the drop-down menu in the editor to make it a **Heading level 3**.

A simple way to express the answer is using the wild card \*. You may have come across wild cards in the Windows Explorer search. Try entering `*knob* *curtain *window* *chimney* *cloud*` in the field labeled **Answer 1**. This works, as it will match responses such as `door knob`, `curtain missing`, `extra window`, `chimney pot`, `cloud`. However, the pupil has to get the answer in precisely the correct order, bottom to top. And they are unlikely to achieve this. Can we improve matters?

In the previous steps, I deliberately didn't mention the **Use regular expression** checkbox. Regular expressions are like wildcards, but they are much more sophisticated. Follow these steps:

1. Tick the checkbox next to **Use regular expressions**. Leave the **Page title** and **Page contents** fields as they are.
2. In this situation the order of the answers matters, so scroll to **Answer 5**, and enter `.*`. This is the **catch-all** for a correct answer. In regular expressions, dot `.` means match any character and the `*` means match any number of times. Next to **Response 5**, type **Correct. Well done!** Set **Jump 5 to Next page** and change **Score 5** to 0.

3. Go to **Answer 2** and type in `--.*curtain.*i`. This looks complicated, yes? Let's break it down. `--` is a Moodle-specific extension of the regular expression syntax. It allows us to check for the presence of some desired text. As before, `.*` matches any character any number of times. And `/i` allows us to check for upper or lower-case characters. That is, it makes the regular expression case-insensitive. So the whole expression means, if the answer does not contain any characters followed by the word **curtain** followed by any characters, then the answer is incorrect. So this allows us to match words in any order. Neat, but complicated!
4. For **Response 2** enter a phrase such as **Something is missing. Try again.** Make sure that **Jump 2** is set to **This page** and **Score 2** is 0.
5. Repeat steps 3 and 4 for **Answer 3** (window) and **Answer 4** (chimney). Now we have a problem! I created five differences in the **Nicu's house** image. And for that we would need six answer fields—one for each keyword, and one catch-all. However, we only have five answer fields.
6. The workaround is to enter two keywords in **Answer 1**, which is why we left answer 1 to the end. Enter the text `--.*knob.*cloud.*i`, which is similar to the previous regular expression. However, it requires **cloud** to be after **(door) knob**. (Order does not generally matter, apart from this exception—which is a little awkward but unavoidable.)
7. As we did previously, fill in **Response 1** with **Something is missing. Try again.** Set **Jump 1** to **This page** and **Score 1** to 0. Scroll to the bottom of the form and press **Save page**.

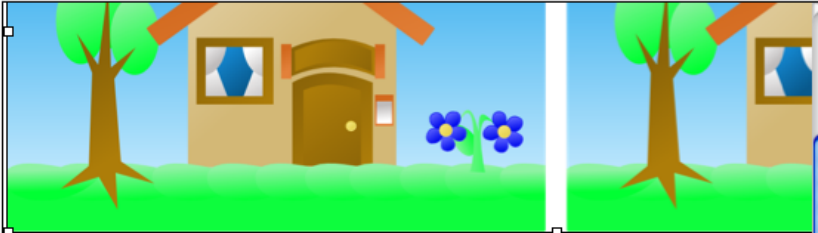
Now you will want to give our three questions a try. The third question may not work first time. If this is the case then go back and edit question 3, paying attention to the syntax of the regular expressions.



Spot the difference 3 - House

**Page contents:**

Trebuchet 1 (8 pt) Normal Lang B I U S x<sub>2</sub> x<sup>2</sup> [icons]



**Start from the bottom and work up. Can you describe the differences?**

Path: body » p » img

**Answer 1:**

--.\*knob.\*cloud.\*i

**Response 1:** [Use editor: ☐ ?] Something is missing. Try again.

**Jump 1:** This page [icon] ? Score 1: 0

**Answer 2:**

--.\*curtain.\*i

**Response 2:** [Use editor: ☐ ?] Something is missing. Try again.

### ***What just happened?***

We added two more images and two questions to our spot-the-difference activity. One was a fairly simple image, for which we used the **multiple choice** question type. And the other was a more complicated image containing five differences. For this we used the **Short answer** question type, and we learned some new regular expression syntax. This was a bit tricky, so well done!

## Wrapping up the lesson

Now that we have three questions in our lesson activity, spanning a range of difficulties, the final step is to add a branch table. As the flow diagram at the start of the chapter shows, this will allow the pupils to decide whether to try a difficult spot-the-difference exercise after they have tried a more straightforward one.

### Time for action – adding a branch

To add a branch table, go back to the **Edit** tab for the lesson activity. Then you can follow these steps:

1. Scroll down to the row of links between the first and second spot-the-difference exercises.
2. Click on the link labeled **Add a branch table**.
3. Enter a **Page title**, for example, **Your choice**.
4. Enter some explanatory text under **Page contents**. You will probably want to mark up some of the text as a bulleted or unordered list. I selected the two lines shown in the next screenshot, and pressed the **Bulleted list** icon as shown here:



5. Enter a short phrase such as **Try an easier one** for the **Description 1** field. This will be used as a label for the button. Choose **Spot the difference 2 - suit** from the **Jump 1** drop-down menu.
6. And, enter a short phrase such as **Try a harder one** for the **Description 2** field. This will be used as a label for the button. Choose **spot the difference 3 - house** from the **Jump 2** drop-down menu.
7. Scroll to the bottom of the form and press the **Add a branch table button**.

### Add a Branch Table ?

**Page title:**  
Your choice

**Page contents:**

Trebuchet 1 (8 pt) Heading 3 Lang B I U  $\times_2$   $\times^2$

**Well done for completing the first spot the difference exercise**

- If you found the first exercise quite difficult, then choose button labelled "Try an easier one".
- If you found it quite easy, then choose the button labeled "Try a harder one".

Path: body » ul

☒ Arrange Branch buttons horizontally?  
☒ Display in left menu?

**Description 1:**  
Try an easier one

**Jump 1:** Spot the difference 2 – Suit ?

**Description 1:**  
Try a ha

**Jump 2:** Spot the difference 3 – House ?

### ***What just happened?***

We added a branch table after the first question to give our pupils a choice. They will have the option to choose an easier or harder spot-the-difference exercise. And we used the editor to create a bullet list.

## Enrolling students

When you launch an online course at your school, you will need to enroll students. There are various ways to achieve this, and we will discuss two possibilities. Which you use, if either, will depend on the age and maturity of your class, and what other systems your school uses. Although as a teacher, you can configure some enrolment for your courses yourself, you should probably discuss this with fellow teachers and your friendly IT support person.

- ◆ Enrolment with an **enrolment key** is suited to older pupils who can use e-mail and can be trusted to create and remember passwords
- ◆ Enrolment via **user upload** can be used for younger pupils, and smaller groups of pupils

We will walk through both methods.

## Configuring an enrolment key

As a teacher, you can configure an enrolment key for your own courses. First, you should check with your friendly IT support person whether **Email-based self-registration** is enabled.

### Time for action – creating and using an enrolment key

To configure an enrolment key for a course in Moodle 2:

1. In the **Settings** side-block, expand **Course administration | Users**, and click on the **Enrolment methods** link.
2. There will be a table containing three enrolment methods: **Manual enrolment**, **Guest access**, and **Self enrolment (Student)**. The latter two will initially be disabled, and thus grayed out. Click on the **Edit** (hand) icon for **Self enrolment**.
3. Moodle will take you to the **Self enrolment** form. Choose **Yes** for **Allow self enrolments**.



We wish to create an enrolment key, which is similar to a password. For security, you may want to use a password generator. You can search, for example on Google, for **online password generators**. I found a useful one at <http://maord.com/>. As you can see in the next screenshot, I generated and copied an eight-character key containing letters and numbers, which is fairly strong. Then I pasted it in the **Enrolment key** field. (Tick **Unmask** to check what you have copied.)

4. Leave **Use group enrolment keys** set to **No** and **Assign role** set to **Student**.

5. Tick the checkboxes next to **Start date** and **End date**. As you will be supervising enrolment in the classroom, set the **Start date** to when you first plan to use the course site, and the **End date** to one to three days later.
6. Leave the defaults for the next three options, including **Send course welcome**. Enter a **Custom welcome message**. I typed **Welcome class! This is to the spot the difference course, MY104**.
7. When you are happy, scroll to the foot of the page and press **Save changes**. You will see that **Self enrolment** has been enabled on the **Enrolment methods** page.

**Enrolments**

Enrolment Plugins

Default role

Course enrollable ☐ No ☒ Yes ☐ Date range

Start date    ☐ Disable

End date    ☐ Disable

Enrolment duration

**Enrolment expiry notification**

Notify

Notify students

Threshold

**Groups**

Group mode

Force

**Availability**

Availability

Enrolment key  ☒ Unmask

Guest access

The location of the self-enrolment settings is slightly different in Moodle 1.9:

8. Go to the course's main page for the spot-the-difference course, **MY104**. In the course **Administration** side block, click on the **Settings** link.
9. Scroll down to the section of the settings form labeled **Enrolments**. Leave the **Site default** for the **Enrolment plugins** and **Default role** fields. Ensure that **Course enrollable** is set to **yes**.
10. Move down to the **Availability** section of the form. Ensure that **Availability** is set to **This course is available to students**.

- 11.** Set the enrolment **Start date**, **End date**, and the **Enrolment key** as detailed in the information box under the Moodle 2 instructions previously. Scroll to the end of the form and press **Save Changes**.

To use the enrolment key:

- 12.** In class, put the enrolment key on the whiteboard.
- 13.** The pupils should visit the Moodle homepage for your school; follow the links to **Login**, then to **Create new account**. After they have created a username and password, they should follow the link in the e-mail sent to them to confirm their account.
- 14.** Then the pupils can visit the homepage again, and click on the course link. It will have a key icon next to it.
- 15.** The pupils should type in the enrolment key when prompted. They will then be enrolled by the system.

### ***What just happened?***

We configured our course to use the built-in self-registration functionality. This entailed generating an enrolment key using an online service. Then we stepped through a procedure that your class can follow to create an account and enroll on the course.

### **Enrolling via user upload**

You will need the assistance of your friendly IT support person or site administrator to upload user accounts, unless they give you extra permissions. We will create a text file using a basic text editor—Notepad on Windows, or TextEdit on Mac OS X.

### **Time for action – uploading users**

To create the upload users text file, you can follow these steps:

- 1.** In Windows, go to **Start**, then find **Run**. Enter the program name, Notepad and press the **OK** button. A simple text editor will appear.
- 2.** Enter the field names on one line as shown—`username, password` and so on, with commas between each name. Press *Enter* to insert a new line.
- 3.** On the new line, start to enter the details for a pupil. You will need to decide on the convention for the username. You can choose either a pseudo-random password, or a weaker but easier to remember combination of a dictionary word and a number—think carefully about this.

4. You can use a unique e-mail address for each user, if one is available. Or, some e-mail systems (for example, a Google Mail free account and a Yahoo paid account at the time of writing) allow you to add a tag before the @ symbol. And this could be set to the username, as shown below. This will satisfy Moodle's requirement for a unique e-mail address per user.
5. You finish the line with the `maildisplay` flag set to 0 and the `shortname` for your course, MY104.
6. Repeat the previous three steps for each pupil, ensuring that the username, password, and e-mail address are unique.
7. Save the file on your computer with the extension `.csv`, for comma separated values.

```
username, password, firstname, lastname, email, maildisplay,
course1
tom, rc84hd, Tom, Jones, teacher.name+tom@my.school, 0, MY104
susan, orange32, Susan, Smith, teacher.name+susan@my.school, 0,
MY104
donald, apple21, Donald, Duck, teacher.name+donald@my.school, 0,
MY104
```

Hand the file over to your friendly IT support person and ask them to upload it to Moodle:

8. In the **Site administration** block, they should expand **Users | Authentication** and click on **Upload users**.
9. The page will allow them to upload the CSV file, and preview the results. They should click on the **Upload users** button.

### ***What just happened?***

We created a CSV text file containing user accounts and enrolment details for our pupils. And we handed the file over to the IT support person. Assuming that there were no problems, your pupils will have been added to the Moodle site if they don't already exist. And they will be enrolled on your course.

## Pop quiz

Here are some quick questions to help test your understanding of the chapter. There may be more than one correct answer. Good luck!

1. What are the benefit(s) of branch tables in the lesson module?
  - a. They allow students' paths through a lesson to diverge from a point.
  - b. They allow students' paths to converge to a point.
  - c. They present choices to the student.
2. In the regular expression syntax available in the short answer question type, what does `. *` mean?
  - a. Match a dot at least one time.
  - b. Match any character any number of times.
  - c. Match any character at least one time.
3. Which property of Scalable Vector Graphics (SVG) allows them to scale without losing detail or becoming pixelated?
  - a. That SVG is an open standard developed through the World Wide Web Consortium.
  - b. That in SVG an object is specified in terms of its attributes, like dimensions and position, instead of storing the color of areas or pixels within the image.
  - c. That SVG is based on the Extensible Markup Language (XML), which makes it more interoperable.
4. What is the purpose of an enrolment key?
  - a. They allow all the pupils in a class to share a common password.
  - b. They allow the teacher to log in on behalf of a student.
  - c. They allow a teacher to restrict entry to a course.

## Have a go hero

In this chapter we used the *lesson* activity module to create spot-the-difference exercises. Look back over the previous chapters, and think about which activities could be re-developed in the lesson module, and how.

>> Suggested answer: the alphabet quiz we created in the first chapter could make use of the lesson activity. (The maths quiz as developed by the end of Chapter 2 uses a third-party question type, which is not available in the lesson module.)



## Summary

The spot-the-difference exercises that we have created will be beneficial to your class in a number of ways:

- ◆ As visually appealing exercises they will appeal to younger students
- ◆ Students will have the opportunity to develop their visual and observational skills
- ◆ They will develop their descriptive and decision making skills

We learned a lot in this chapter about creating lesson modules. Specifically, we covered:

- ◆ Representing and planning a lesson activity as a flow diagram
- ◆ Sourcing editable, reusable vector graphics on the Open Clip Art Library website
- ◆ Installing Inkscape, a free, cross-platform Scalable Vector Graphics editor
- ◆ Using Inkscape to edit our spot-the-difference images
- ◆ Adding question pages and branch tables to our lesson
- ◆ Using multiple choice and short answer question types
- ◆ Enrolling pupils

We also discussed the powerful regular expression syntax available in short answer questions.

Now that we've learned about lessons and created a visual exercise, we're ready to look at alternative visualizations for historic and temporal information—which is the topic of the next chapter.



# 5

## Setting Homework

*In the previous chapter, we developed a visual spot the difference exercise. We are going to continue with the visual theme in this chapter. Developing an interest in history among your pupils can be a rewarding and important step in their education. And you can start on this road from an early age, when you can harness your pupil's imagination, their questioning natures, and develop their sense of empathy.*

In this chapter, we will develop an **interactive timeline widget**, potentially with the help and collaboration of our pupils. This will help your pupils visualize some historic events, and the temporal relationships between them. The timeline will contain images and links to other resources. You can use the timeline to foster discussion, which for older pupils may take place in a Moodle forum.

And we will use Moodle to set homework for our pupils. No more lugging 30 exercise books home to mark!

Specifically, we will:

- ◆ Introduce the MIT SIMILE timeline JavaScript widget [third party]
- ◆ Explore examples of timeline/temporal data
- ◆ Create the timeline data file, using a desktop text editor (Notepad2, and so on)
- ◆ Install and configure the SIMILE Timeline Filter
- ◆ Upload the data file and embed the timeline widget in a Moodle page (filter syntax)
- ◆ Engage your students—reflecting on the timeline with an assignment
- ◆ Create a Moodle uploadable assignment

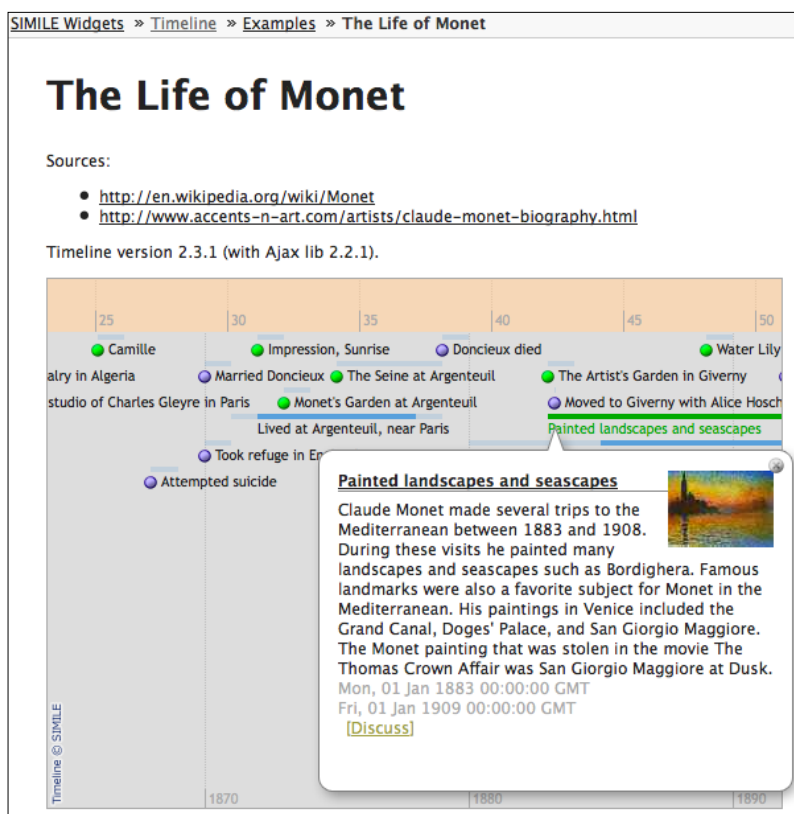
Student learning outcomes:

1. Keyboard/mouse control
2. ICT: uploading an assignment
3. History
4. Visual reasoning

So, on with the show...

## Introducing the SIMILE timeline widget

The **Massachusetts Institute of Technology (MIT)** has developed various visualization and data manipulation tools as part of the **SIMILE** project. One of these is a free/open source **timeline** JavaScript widget, which takes time-based data as input and creates an interactive timeline that scrolls from left to right and contains popup panes and links. A timeline for the life and career of Monet is as follows:



You can view more examples on the web site, <http://simile-widgets.org/timeline/>.

In order to use the timeline widget we need these components:

- ◆ The Moodle timeline filter, containing the SIMILE timeline Javascript libraries
- ◆ A timeline data file, in XML or **JSON (Javascript Object Notation)** format
- ◆ Photographs to show in the popup panes
- ◆ A web page to host the timeline

We will deal with installing the filter later, but first we must decide on the subject for our timeline. If you visit the home of SIMILE, <http://simile-widgets.org/timeline/>, you will be able to explore timelines for the assassination of John F. Kennedy, the life of Claude Monet, and other examples. Timelines can be granular to the minute or hour, as in the case of the assassination. Or they can be spread over centuries or millennia—this is currently the limit for the widget.

A suitable subject for our young audience would be significant or **important inventions**. This can encompass subjects as diverse as printing, paper, penicillin, steam, and the computer. And these inventions originate in different parts of the world, which adds an extra dimension to the subject.

Now that we have our subject, a search on Google reveals some useful links, including a list of the top 10 inventions, <http://listverse.com/2007/09/13/top-10-greatest-inventions/>. We may or may not agree with a list like this, but it acts as a useful starting point and aide memoire. There are pictures here, and more information and images on other web sites including Wikipedia.

To progress from ideas to our timeline, we need to create an XML data file. One of the easier tools to help us produce the XML is a syntax highlighting text editor, which we will install now.

If you already have a text editor on your computer that you think is suitable, skip this section.

## Installing a text editor

We have our subject, so the next step is to install some editing software to help us create our **XML timeline** file. Though most operating systems including Windows contain simple text editors, it will be helpful to install an editor that features **syntax highlighting** for various computer languages including XML.

Some general-purpose text editors are:

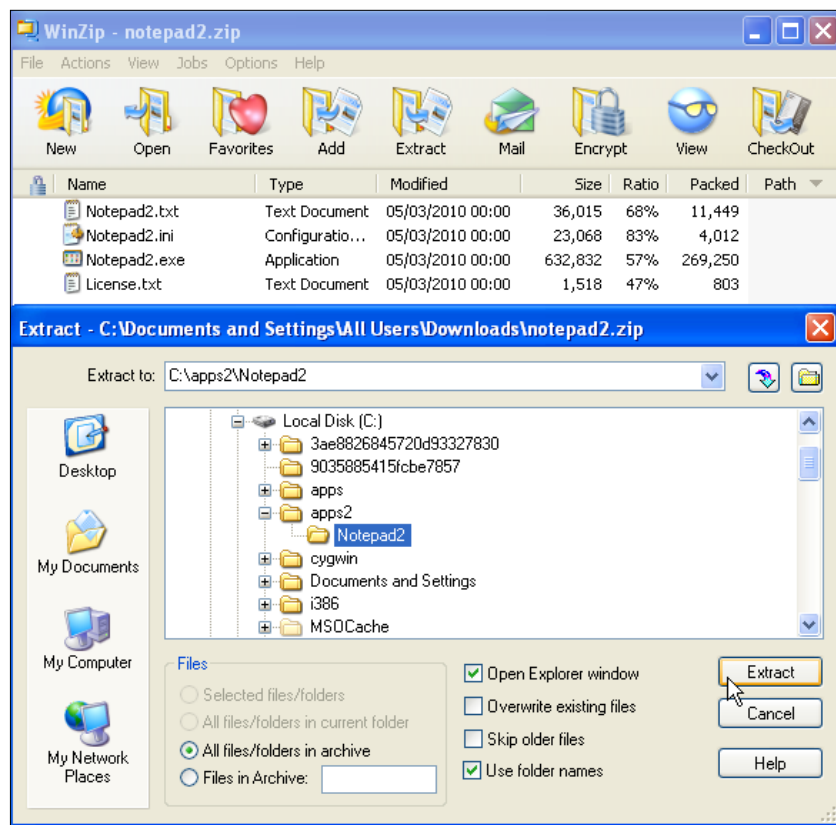
- ◆ **Notepad2** for Windows, a simple open-source editor, distributed under a BSD license (<http://opensource.org/licenses/bsd-license.php>). It is a small download, less than 300 kilobytes, from <http://flos-freeware.ch/notepad2.html>.
- ◆ **Notepad++** for Windows, distributed under a GPL license. Download it from <http://notepad-plus-plus.org>.
- ◆ **TextWrangler** for Mac OS X, distributed under a Freeware license (not open-source), <http://barebones.com/products/textwrangler/>.
- ◆ Most flavors of Linux and Unix include the **vi** or **vim** text editor. Or use **Emacs** or your favorite editor.

We will carry on and install Notepad2 for the examples in this chapter.

## Time for action – installing Notepad2

To install **Notepad2** on **Windows**, visit the web site <http://flos-freeware.ch/notepad2.html> in your browser and follow these steps:

- 1.** Under the **Downloads** section, find the link to the `notepad2.zip` file. Download it to your computer.
- 2.** Open the file **notepad2.zip** with the Unzip program available on your computer (the one built-in to Windows XP or later. Or Winzip, 7zip, or similar).
- 3.** Extract all the files in `notepad2.zip` to a directory on your hard drive, for example, `C:\apps2\Notepad2\`.

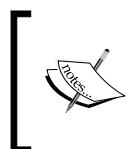


4. In your new directory, click on the file **Notepad2.exe** with your right mouse button. Choose **Create Shortcut** from the context menu that appears.
5. Name the shortcut **Notepad2**, and copy it to the clipboard.
6. Now in Windows Explorer on Windows 7, go to the directory `C:\Users\{USER}\SendTo\` (or on Windows XP, go to the directory `C:\Documents and Settings\{USER}\SendTo\`). In each case {USER} is your username. Paste a copy of the shortcut in the directory. (Note, you will probably see a shortcut for **Notepad**, the basic editor included in Windows in this directory. You may want to delete it to avoid confusion.)
7. Go to your **Desktop** and paste the shortcut there as well.

Whew! **Notepad2** is a little fiddly to set up, but don't worry, its easy to use.

## What just happened?

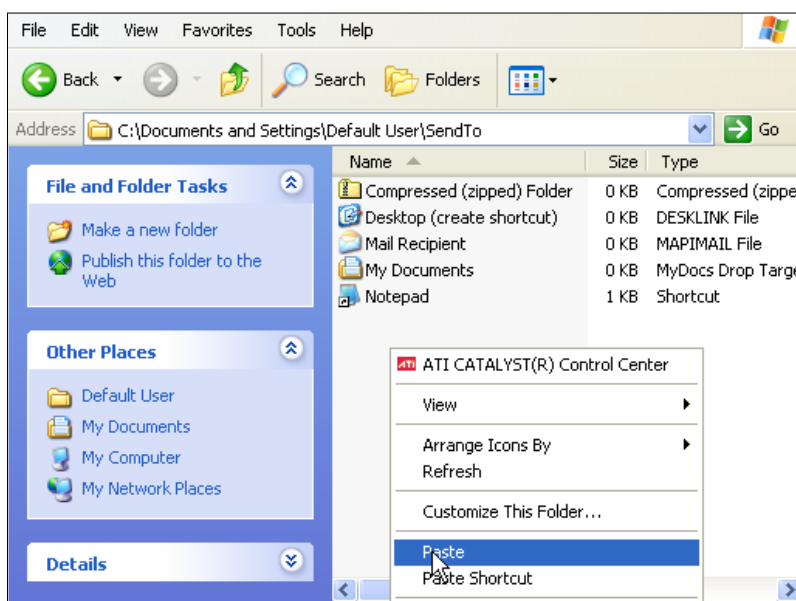
We downloaded and installed a text editor for Windows called **Notepad2**. This has a feature called syntax highlighting, which as we will see helps us to write the XML file for our timeline widget.



Note that the shortcuts we created are useful in different contexts. The **SendTo** shortcut is useful when we wish to edit a file—simply select it in Windows Explorer, right-click, and choose **SendTo | Notepad2** in the context menu. The **Desktop** shortcut is useful for creating new files.

We also found that there are many alternatives for Windows, Mac OS X, and other operating systems. These included Notepad++ for Windows, TextWrangler for Mac OS X, and vi/vim and Emacs for Linux.

We have added a shortcut for the editor to the **SendTo** menu in Windows, as shown below:



Now that we have a text editor we can create the timeline XML.



## Creating the timeline data file

We will press on with creating the XML data file for our timeline widget, and we will be also using the editor that we just installed.

### Time for action – creating the XML

Follow these steps to start creating our timeline file:

1. On Windows, go to the desktop. Browse to the shortcut for **Notepad2** and double-click to launch the editor.
2. First press the **Save** icon, browse to a suitable location on your computer, perhaps **My Documents**, and choose a name for the timeline file. It should end with the extension `.xml`. I chose `simile-invent.xml`. Press the **Save** button.
3. Returning to the editor, start typing the code that you see below. Note the use of angle brackets `<`, `>`, and `</` to mark up the start and end of **elements**, the equals sign `=` for **attributes**, and the double-quote character `"` around attribute values. This is XML, which is a little like HTML, but more general in its use, and stricter in its syntax.

```
<?xml version="1.0" encoding="utf-8"?>
<data xml:lang="en">
  <event>
    The Chinese invention...
  </event>
  <event>
    Charles Babbage...
  </event>
</data>
```

4. Finally, press the **Save** icon again.

### What just happened?

We dived into **XML** and started to create our timeline file. We started by saving the file in Notepad2 with an `.xml` extension. The editor deduces what language (XML) it is displaying based on the file extension.

Let's dissect the code that we typed:

- ◆ The line starting `<?xml ...?>` is termed as the **XML prolog** and tells the client, for example Internet Explorer, what version of XML and character encoding it is looking at. You should use **UTF-8**, which is a **Unicode** character encoding. This is widely adopted around the world, and allows you to type accented and non-Latin characters.
- ◆ The next line contains the start of a data tag, denoted `<data>`. And it contains an attribute `xml:lang="en"`. This tells us that the document is written in English (**fr** for French, **de** for German...). Well, we knew that! Still it is important to state it, so that for example a screen reader can correctly speak the document to a person who is blind.
- ◆ The next three lines contain an event element, which starts with `<event>` and ends with `</event>`. And inside is some text. Now we're getting somewhere!
- ◆ You will then see another event element, and `</data>` to close the data element. There can only be one outer-most or **root** element, in this case **data**. But there can be as many events as we wish.

This allowed us to see the syntax highlighting, as the following screenshot. As we can see, syntax highlighting helps us to distinguish between XML elements, attributes, values, text, and comments.



Text editors like **Notepad2** allow you to edit the syntax highlighting schemes, for example by adjusting the colors, or by associating a scheme with an additional file extension.

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <?xml-stylesheet href="../../simile-xml.css" type="text/css"?>
3 <data
4   xml:lang="en"
5   wiki-uri="http://my.school/moodle/mod/forum/discuss.php?d=2&wiki-title="
6   wiki-section="Important Inventions"
7 >
8 <!--
9   Source: http://listverse.com/2007/09/13/top-10-greatest-inventions/
10  Nick Freear 18 October 2010.
11 -->
12 <event
13   start="868"
14   end="1468"
15   isduration="true"
16   title="printing - chinese woodblock printing, to Johannes Gutenberg"
17   link="http://en.wikipedia.org/wiki/Johannes_Gutenberg"
18   image="http://listverse.files.wordpress.com/2007/09/press-tm.jpg?w=129&h=187"
19   icon="green-circle.png">
20 <p>The &lt;a href=
21 "http://en.wikipedia.org/wiki/Four_Great_Inventions_of_Ancient_China#Printing">Chines
22 at some point before...
23 &lt;/p>
24 </event>
25
26 </data>
```

As we will see it is useful to view the XML file as we create it in a browser. You can use most modern web browsers, including **Mozilla Firefox**, **Internet Explorer**, **Opera**, and **Safari**. Go to the menu bar in the browser, choose **File | Open...**, and browse to your XML file. Press the **Open** button.

## Troubleshooting

When you view the XML file in a browser, you may see something like the following screen. Firefox displays a scary message.



XML Parsing Error: mismatched tag Expected: </event>. Location:... Line Number 8, Column 3:...

### *What just happened?*

All modern browsers incorporate what is termed an **XML parser**. This is a program that analyses and interprets the XML. Whereas with HTML, browsers do their best to recover from badly formed markup, the XML Specification requires that XML parsers must stop at the first error and output an error message. This is called **Draconian error handling**.

The error above means that the final closing `</event>` tag is missing. Note that the line number and column are generally at the first point after the error that the software detects that something is a miss. Try working back from that point to find the error itself. And of course the visual appearance of errors will vary in different browsers.

Other common errors:

- ◆ **Not well-formed:** This may mean that there is a missing angle bracket `>` at the end of a tag, or that an attribute is not correctly quoted.

- ◆ **Undefined entity:** In order to display an angle-bracket (greater/less than symbol) or other characters correctly in the output they must be encoded as entities. There are only five **character entities** allowed in XML: `&lt;`, `&gt;`, `&amp;`, `&apos;`, `&quot;`; (less than, greater than, ampersand, apostrophe, quote). HTML allows many more character entities (for example, `&copy;` for the copyright symbol)—these must be encoded as **numerical entities**, for example, `&#169;` in XML. Refer to this list, [http://w3schools.com/tags/ref\\_entities.asp](http://w3schools.com/tags/ref_entities.asp).
- ◆ **No element found:** The final closing tag `</data>` may be missing.

As you can probably see, Web browsers serve a useful purpose in checking the **well-formedness** of the XML, before we use our timeline file in Moodle. If all goes well and with a following wind you will see output as shown in the following screenshot:



Now seems an appropriate time to install the Timeline Widget filter.

## Installing the Timeline Widget filter

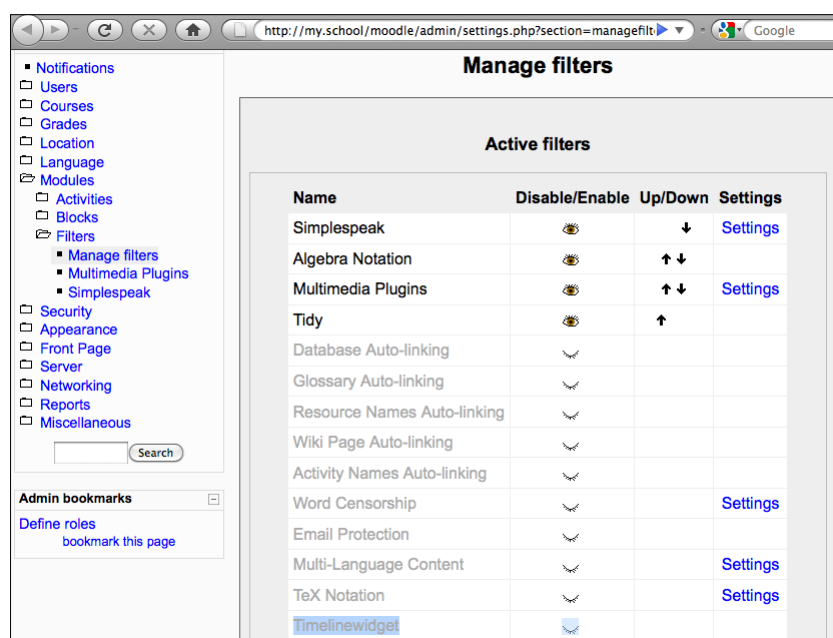
Before long we will want to embed the timeline stored in our XML file. As you will probably not have the necessary permissions yourself, this is an appropriate point to ask your friendly IT support person to install the Timeline Widget filter on your behalf. You may wish to look over their shoulder.

### Time for action – installing the filter

Ask your IT support person or system administrator to follow these steps:

1. Visit the entry for the **Timeline Widget** filter in the Moodle plugins database, [http://moodle.org/plugins/view.php?plugin=filter\\_timelinewidget](http://moodle.org/plugins/view.php?plugin=filter_timelinewidget), (if you need to search, don't confuse it with the Timeline course format).
2. Read the compatibility and installation notes. Download a Zip archive of the widget that is compatible with the version of Moodle on your server.

3. Unzip the widget code and rename the resulting directory to **timelinewidget**. Copy it to the filter directory on your server, for example, `/var/www/moodle/filter/timelinewidget/`.
4. On your local site, visit the Moodle **Site Administration** notifications page (there are currently no database changes for this filter, but it's best to check).
5. Visit the Moodle **Site Administration | Plugins (Modules) | Filters | Manage Filters** page. You will see something like the figure below. Scroll down the list of filters to the disabled / grayed out filter named **Timeline Widget (Timelinewidget)** in Moodle 1.9). Select **On** from the drop-down menu next to it (click on the closed eye to enable it in Moodle 1.9).



## What just happened?

We found and downloaded the Timeline Widget filter from the Moodle website. The filter is available under the GNU General Public License. It is written by yours truly and incorporates the MIT SIMILE Javascript libraries. Our IT support copied the code to our Moodle installation, and enabled the filter in Moodle's administrator interface and will then have handed back to you.

Job done and thank you! We have started the timeline XML and installed the filter. Now we need to make a short diversion and plan an activity that will be linked to from within the timeline. We will call it a follow-on activity.

## Creating a follow-on activity

It is possible to put a link at the foot of each popup pane on the timeline to a resource that is external to the timeline. This link will be labeled **Discuss** so it can be a discussion space like a Moodle forum, or forum discussion. Alternatively, you may want another sort of activity to flow from the timeline, particularly if you judge that your class is not mature enough to benefit from a forum. So, it can be a link to a collaborative space like a Wiki, a lesson, web page resource and so on.

If you do want a follow-on activity, then you need to create it before completing the timeline XML. It's your choice!

### Time for action – adding a forum

We will assume that you wish to present links in the timeline to a Moodle forum. To add a Moodle forum follow these steps:

1. Log in to Moodle as a teacher, and create a new course. I chose **Setting Homework** as the course title and **MY105** as the short name.
2. Go to the main page for your newly created course. Press the button labeled **Turn editing on** at the top right of the page.
3. In the first topic or week of the course, select **Forum** from the **Add an activity** drop-down menu.
4. On the new forum page, type a **Forum name**. I chose **Timeline discussion**. And enter a **Forum introduction**, as shown in the following screenshot from Moodle 2.
5. For the **Forum type** choose **Standard forum for general use** from the drop-down menu.
6. For the **Subscription mode** choose **Forced subscription** from the drop-down menu. (In Moodle 1.9 this looks a little different—next to **Force everyone to be subscribed** choose **Yes, forever**). These options will keep things easy for you and your class.

**Adding a new Forum to topic 1**

**General**

Forum name\* Timeline discussion

Forum type A single simple discussion

Forum introduction\* Let's discuss the Important Inventions timeline here. You can add extra information you find out, and propose your own inventions.

Path: body

Force everyone to be subscribed? Yes, forever

Read tracking for this forum? Yes, initially

Maximum attachment size 500KB

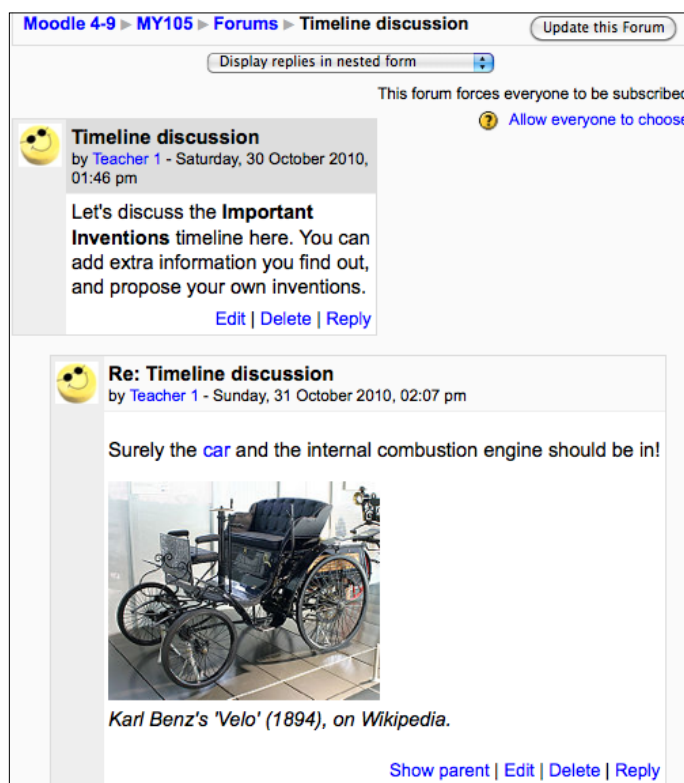
7. Scroll to the bottom of the form and press **Save and display**. Make a note of the link that you are taken to—we will need it later. My URL is, <http://my.school/moodle/mod/forum/view.php?f=10> (the URL may also end in `id=NN`, don't worry.)

As you can see below, I also wrote a post replying to my first discussion, to get the ball rolling!

### ***What just happened?***

We created our fifth course, and added a forum activity to it. We chose a standard forum, which provides the opportunity for multiple threaded discussions. If you are teaching a young class and want to make the activity simpler for you and your class you can set the forum type to **A single simple discussion** as an alternative. There will be links to this discussion in our interactive timeline, so that our pupils can comment on and collaboratively add to the timeline.

You will end up with a forum activity that looks something like the following screenshot. The photograph of the Benz Velo is by Softeis on Wikimedia Commons (<http://commons.wikimedia.org/wiki/File:Benz-velo.jpg>), licensed under a Creative Commons Attribution-ShareAlike license.



Now that we have designed a follow on activity to refer to from the timeline, it's time to return the XML timeline file.

## Adding information to our XML file

Earlier in the chapter we started creating the XML file for our inventions timeline. To bring the timeline to life for our pupils we will need to add some extra details, and we want to enrich it with images, links, and so on.



## Time for action – editing the XML

Return to the XML file in your editor. We are going to add some information that affects the whole document, and some that is specific to each event.

First we'll handle the global changes:

1. Add a line starting `<?xml-stylesheet` as shown below. You should replace `{MY_MOODLE}` with the address of your school's Moodle site. As well as the interactive timeline the filter will display a backup link to the raw XML. This stylesheet instruction will display the raw XML with formatting in most browsers.
2. Now, add two attributes to the outer `<data>` element, named `wiki-url` and `wiki-section`. Put the URL for the forum discussion from earlier (page 13) into `wiki-url`, and append `&title=` (remember the `=` at the end). Put a keyword in the `wiki-section` attribute. I chose `Inventions`.
3. You may also wish to add a comment containing your notes, at the top of your timeline file. A comment starts with `<!--` and ends with `-->`. This is what the XML file should look like at this point:

```
<?xml version="1.0" encoding="utf-8"?>
<?xml-stylesheet href="{MY_MOODLE}/filter/timelinewidget/simile-
xml.css" type="text/css"?>
<data
  xml:lang="en"
  wiki-url="{MY_MOODLE}/mod/forum/view.php?f=10&title="
  wiki-section="Inventions">
  <!--
    Source: http://listverse.com/2007/09/13/top-10-greatest-
inventions/
    by Nick Freear 18 October 2010.
  -->
  <event>
  ...
```

Now let's add our extra data to each event.

In the example below, we are looking at an **event** related to the invention of printing. We can say that this event starts with the discovery of wood block printing in China sometime before the first book in **868**. And it could end on the death of the inventor of mechanical movable type, Johannes Gutenberg in **1468**. That's 600 years summarized in a few sentences!

Here are the attributes that you will want to add to the `<event>` opening tag:

1. Add the attribute `start` date with a value of 868, and an attribute `end` date equal to 1468. Note that for events which span a few hours to a few years, you can specify a full date, for example, January 20 1700 14:15:00 GMT.
2. Add an attribute `isDuration` with a value of `true`.
3. Add a `title` attribute with a suitable value. For this event, I put `Printing: from Chinese wood block printing, to Gutenberg`.
4. You have the option to add a `link` attribute. You can provide an external URL that is the source for the main text between the start `<event>` and end `</event>` tags. I chose the link to the Gutenberg article on Wikipedia.
5. You also have the option to add a small image to the popup panes within the timeline. So, for the `image` attribute, I sourced a photograph of a Gutenberg press by **Andrew Plumb** on **Flickr**, <http://flickr.com/photos/aplumb/121285772/> (Creative Commons Attribution-ShareAlike license). Note, the image should not be too big, I chose the **small** image from Flickr, 180 x 240 pixels.

```
<event
  start="868"
  end="1468"
  isDuration="true"
  title="Printing: from Chinese woodblock..."
  link="http://en.wikipedia.org/wiki/Johannes_Gutenberg"
  image=
  "http://farm1.static.flickr.com/39/121285772_f3998d2e7a_m.jpg"
>
```

The next step is to edit the main text in between our `<event>` start and end tags, so that they contain links and perhaps are divided into paragraphs. To do this we must **encode** the HTML tags, within the XML as entities. That is, each less-than character `<` becomes a `&lt;` entity.

An example will make things clearer. The following is a HTML paragraph containing a hyperlink:

```
<p>The <a href=
"http://en.wikipedia.org/..._Ancient_China#Printing">Chinese
invention...</a>, at some point before
...
</p>
```

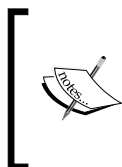
When the HTML tags are encoded this becomes:

```
&lt;p>The &lt;a href=
"http://en.wikipedia.org/..._Ancient_China#Printing">Chinese
invention...&lt;/a>, at some point
...
&lt;/p>
```

## What just happened?

We edited our timeline XML and added a number of useful global attributes. And we added further attributes to each event. Finally, we edited the body of the events to add links and generally make them richer.

We can find dates, text, and images at Wikipedia, Listverse, Flickr and any other interesting sources, including offline ones. And we will be able to bring the historic events to life for our pupils, and fire their imaginations.



You will want to check your XML in your web browser at intervals, to catch and fix any errors. There is a limit to the errors that the browser can pick up. It will generally help you spot misspelled or missing opening and closing tags. However, it will not normally tell you about missing or misspelt attributes. And, the encoded HTML may prove the trickiest to get right.

Bringing it all together, we have XML that looks something like the following:

```
<?xml version="1.0" encoding="utf-8"?>
<?xml-stylesheet href="{MY_MOODLE}/filter/timelinewidget/simile-xml.
css" type="text/css"?>
<data
  xml:lang="en"
  wiki-url="{MY_MOODLE}/mod/forum/view.php?f=10&title="
  wiki-section="Inventions">
  <event
    start="868"
    end="1468"
    isDuration="true"
    title="Printing: from Chinese woodblock..."
    link="http://en.wikipedia.org/wiki/Johannes_Gutenberg"
    image=
    "http://farm1.static.flickr.com/39/121285772_..._m.jpg"
  >
    &lt;p>The &lt;a href=
    "http://en.wikipedia.org/..._Ancient_China#Printing">Chinese
    invention...&lt;/a>, at some point
```

```
...
    &lt;t;/p>
</event>

...

</data>
```

If you've stuck with it, and completed an XML timeline file, well done! XML is tricky to hand-craft. Now, we'll move on to the fun part, namely embedding a timeline widget in our course.

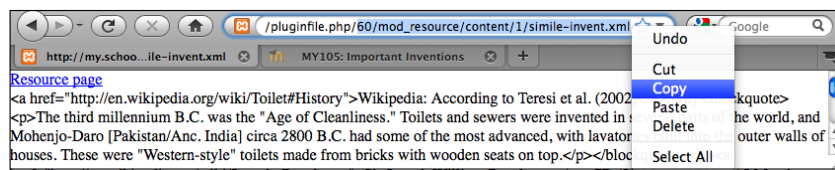
## Embedding the timeline

In the previous sections we created a timeline XML file. This is going to form part of a history lesson for our pupils.

### Time for action – putting a timeline together

Now we will find out how to embed the timeline in our course. Follow these instructions for Moodle 2 (the Moodle 1.9 variation follows):

1. Return to the **MY105** course and press the **Turn editing on** button.
2. In topic 1, choose **File** from the **Add a resource...** drop-down menu.
3. When the form appears, give the **File** object a **Name**. I typed **Important Inventions XML**. Enter a **Description**.
4. In the **Content** section of the form press the **Add...** button. On the left of the **File picker** press the **Upload a file** link.
5. In the **File picker**, press the **Browse...** button next to the **Attachment** field. Browse to and select the timeline XML file on your computer. Press **OK**, and when you return to the **File picker** press the button labeled **Upload this file**.
6. The name of your uploaded XML file will appear below the **Add...** button. Leave the **Display** option set to **Automatic**. Scroll to the foot of the page and press **Save and display**.
7. You will be taken to a page like the one shown below, with the XML content, including the HTML tags you added, all run together. Don't worry! Just select the part of the URL in your browser's address bar after `pluginfile.php/`. For example, `60/mod_resource/content/1/simile-invent.xml`. Copy this to the clipboard for the next stage.



These are the steps to follow in Moodle 1.9:

8. Return to the **MY105** course and in the **Administration** side-block click on the **Files** link.
9. Upload the XML file to our course. Then select the link for the file in Moodle's file manager and copy it to the clipboard. The URL for my file is `http://my.school/moodle/file.php/9/simile-invent.xml`.

Continue with these steps (for both Moodle 1.9 and 2):

10. Return to the course, turn on editing and in the **Add an activity** drop-down menu choose **Compose a web page**.
11. Give the page a **Name** (I wrote **Important Inventions**) and provide an introduction in the **Summary**.
12. Then in the section labeled **Compose a web page**, enter the following code:

```
[Timeline]
; A comment.
title = Important inventions timeline
dataUrl= 60/mod_resource/content/1/simile-invent.xml
date = 1870
intervalUnit = CENTURY
; How wide should the unit defined above be? In pixels.
intervalPixels= 75
[/Timeline]
```

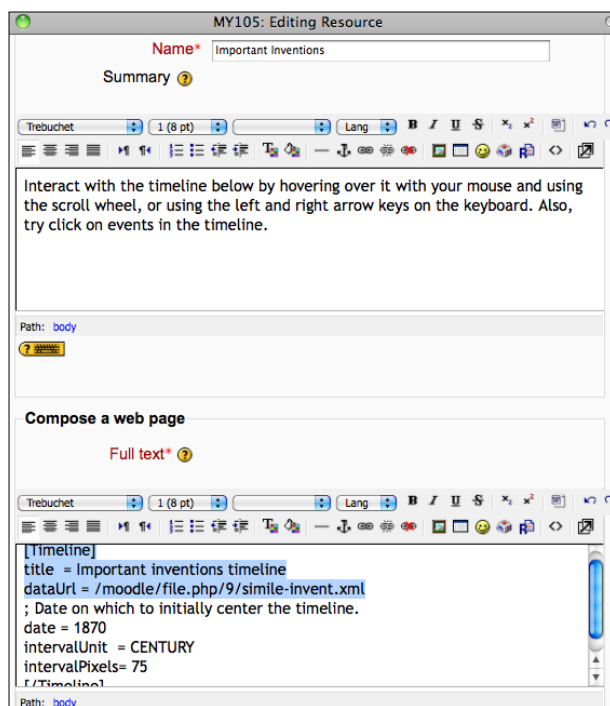
13. Note that we are using similar **square-bracket** syntax to that used way back in *Chapter 1, Getting Started*. The macro starts with `[Timeline]` and ends with `[/Timeline]`.
14. Comments are only visible to you—they live on their own lines and start with a semi-colon `;`. To separate properties from their values use an equal sign `=`.
15. The `title` property contains a short summary of the timeline.
16. The `dataUrl` is the link to the XML file you uploaded previously. Note the mix of the lower and upper-case letters (termed camel-case).

- 17.** The `date` property specifies the point on which you wish to center the timeline initially. I used 1870, and you can tweak this value later.
- 18.** The value for `intervalUnit` is always in capitals. It can vary between **minute** and **millennium**, and includes decade. It is always singular. I chose `CENTURY` for this timeline (if I required a smaller interval for a timeline, for instance an hour, then the date specified above would need to be more specific, for example, January 20 1700 14:15:00 GMT).
- 19.** And finally, `intervalPixels` specifies how wide you would like the unit you specified, in pixels. I entered 75.

Note that only `dataUrl` is **required**. If you omit `title`, `date`, and so on, then the following defaults will be used:

```
[timeline]
title = My timeline
date = 1900
intervalUnit = DECADE
intervalPixels = 35
[/timeline]
```

You can see the **Compose a web page** form in Moodle 2 as follows:



Of course, you can enter other text around the timeline definition. I provided the links to my sources including **Wikipedia** and **Listverse**. The completed resource form will look something like the next screenshot.

Finally, scroll down to the end of the form and press the button **Save and display**. And admire your handiwork!

## What just happened?

We uploaded the timeline XML file to our course, and then created a web page resource. In the web page we created a macro that pointed to our uploaded file, and specified various other properties.

When you save and view the web page you will see something like the following screenshot. We can see the years 1000 to 2700 at the foot of the timeline. Blue lines on the timeline indicate the span of events, and we can see the title of each event. You can scroll left and right with the mouse scroll-wheel, or the left and right arrow keys on your keyboard. Clicking on a text results in a popup box like the one shown. In the popup, we can see the image we specified, the text, and any HTML mark up like the hyperlinks. At the bottom of the popup is a link labeled **Discuss**, which will take you to the discussion forum we specified.

The photograph and text below concerns the development of the computer, from Charles Babbage to the Colossus computer built during World War II. (The photograph and text are sourced from **Wikipedia**, [http://en.wikipedia.org/wiki/Colossus\\_computer](http://en.wikipedia.org/wiki/Colossus_computer).)

**Moodle 4-9 > MY105 > Resources > Important Inventions** Update this Resource

from the Chinese compass, to the printing press, to Gutenberg

Important inventions timeline

**Sources**

- Listverse.com /.. Top
- Inventors.about.com
- Wikipedia.org

**The Computer: from Charles Babbage, to Colossus**

**Image** © SHOLE

**Wikipedia: Charles Babbage**, FRS (26 December 1791 – 18 October 1871)[2] was an English mathematician, philosopher, inventor, and mechanical engineer who originated the concept of a programmable computer.[3] He began in 1822 with what he called the **difference engine**.

The **Colossus machines** were electronic computing devices used by British codebreakers to help read encrypted German messages during World War II.

Tue, 01 Jan 1822 00:00:00 GMT  
Sat, 01 Jan 1944 00:00:00 GMT  
[\[Discuss\]](#)

[http://localhost/moodle/mod/forum/view.php?f=10&title=Inventions/The\\_Computer\\_from\\_Charles\\_Babbage\\_to\\_Colossus](http://localhost/moodle/mod/forum/view.php?f=10&title=Inventions/The_Computer_from_Charles_Babbage_to_Colossus)

In the following sections, we will explore how to integrate the timeline activity into lessons.

## Integrating the timeline with teaching

In the previous sections we built up the XML data file for our timeline, and learned how to embed it in a Moodle resource using the Timelinewidget filter.

There are a variety of ways of embedding the timeline activity in your teaching. One way would be to present the timeline and discuss it with your pupils. Then you may wish to set an assignment.

### Time for action – creating an assignment

Let us explore how we would create a homework assignment using Moodle. We will ask our pupils to upload a file to Moodle. Return to the main page for the **MY105** course, turn on editing and follow these steps:

1. In topic **2**, go to the **Choose an activity** drop-down menu. Select the **Assignments | Upload a single file** option. You will be taken to the **Editing Assignment** page.
2. In the form, enter an **Assignment name**. I typed **Homework: exploring inventions**.
3. Write a **Description**. As you can see in the following screenshot, there are instructions to upload a Word document. You may also like to provide a link to the free **OpenOffice** software that can also create and edit Word doc files (<http://www.openoffice.org>).
4. Next, you may wish to change the **Grade** using the drop-down menu. I reduced the grade from 100 to 50.
5. Adjust the **Available from** date and **Due date** as you see fit.
6. I left the default for the **Prevent late submissions** drop-down menu. That is, **No**.
7. For the drop-down menus **Allow resubmitting** and **Email alerts to teachers**, I changed from the default to **Yes**.
8. Finally, I reduced the **Maximum size** for uploads from 1 MB (limited by the course setting), to **500 KB**. The instructions specify **small images**, but you may have to increase this limit later. You can see how the assignment form looks for Moodle 2 in the following screenshot:



**General**

Assignment name\* Homework: exploring inventions

Description\* ?

Trebuchet 1 (8 pt) Normal Lang B I U S x x<sup>2</sup>

In class, we have looked at a timeline of some important inventions.

Your homework is to find out about and write about **another invention** that you think is important.

You should upload a Word document, which can contain small images, quotes and links Internet. (Create the document using MS Word, or the free [OpenOffice](#).)

Path: body » p

Grade ? 50

Available from 29 October 2010 11:20 Disable

Due date 7 November 2010 11:20 Disable

Prevent late submissions No

**Upload a single file**

Allow resubmitting ? Yes

Email alerts to teachers ? No

Maximum size ? 500KB

**Common module settings**

Group mode ? No groups

9. Scroll down to the bottom of the form and press the button to **Save and display**.

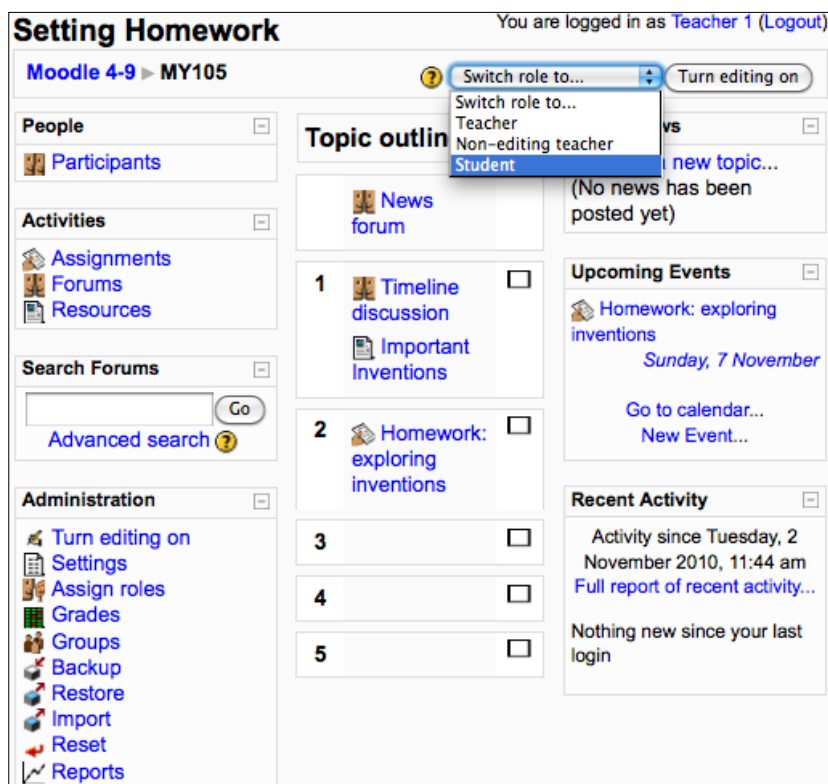
### ***What just happened?***

We created an assignment or homework activity for our class. We instructed the pupils to upload a document with their own ideas for an important invention. When you pressed **Save and display**, you would have been taken to a page that displays the details and status of the assignment. Across the top we see the usual Moodle breadcrumb trail, and on the right is a button to **Update this Assignment**. Below on the right is a link labeled **No attempts have been made on this assignment**. We can see the description of the assignment, and further down the available and due dates.

Obviously not much has happened in this activity yet. So, is there any way we can try it out?

In a word, yes. In Moodle 2, look in the **Settings** side block. Below **Course administration**, expand the item **Switch role to...**, choose **Student** from the list of roles, and hey presto! You will now be impersonating a student or pupil. At the foot of the page you will see a link **Return to my normal role**.

In Moodle 1.9, you will need to return to the main page for the **MY105** course, and look in the top-right corner of the page. You will see a drop-down menu labeled **Switch role to....** Choose **Student** from the selection.



If as a student, you return to the assignment page, you will see a form below the summary of the assignment. There is a button labeled **Upload a file** (in Moodle 1.9 you will see a file upload field, labeled **Browse...** and a submit button labeled **Upload this file**). Go on, try it out with a test Word document. You will find that any files you upload will not register in the number of submitted assignments. This is natural, given that you're not really a student!

Time to switch back. Find one of the **Return to my normal role** links that we found earlier.

**Moodle 4-9 ► MY105 ► Assignments ► Homework: exploring inventions**

In class, we have looked at a timeline of some important inventions.

Your homework is to find out about and write about **another invention** that you think is important.

You should upload a Word document, which can contain small images, quotes and links Internet. (Create the document using MS Word, or the free [OpenOffice](#).)

**Available from:** Friday, 29 October 2010, 11:20 am  
**Due date:** Sunday, 7 November 2010, 11:20 am

Upload a file (Max size: 500KB)

You are logged in as [Teacher 1: Student](#) ([Return to my normal role](#))

**MY105**

This exercise must have given you some ideas about how to use Moodle to set assignments for your pupils to extend what they have been learning in class.

## Pop quiz

To improve your understanding, try these quick multiple choice questions (there may be more than one correct answer):

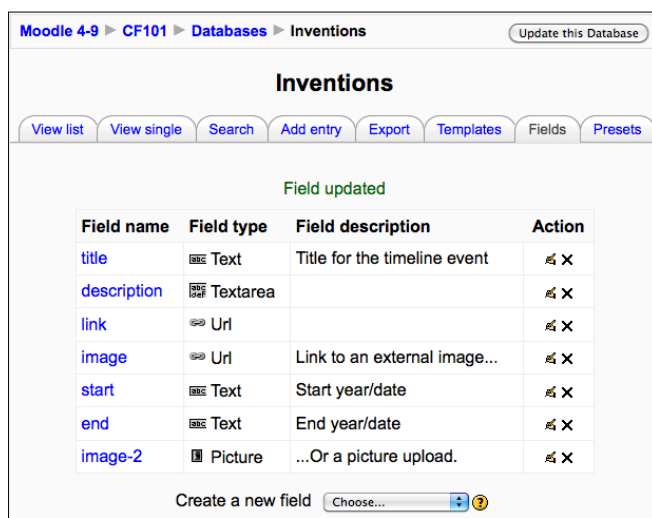
1. Earlier in the chapter we created a particular type of forum. What was it?
  - a. Single discussion
  - b. Multiple discussion
  - c. Nested discussion

2. What did we use to check the timeline XML file for errors?
  - a. Some parsing software specifically for XML
  - b. A web browser
  - c. A careful eye
3. In order to explore our assignment, what did we do?
  - a. Create a dummy student account in Moodle
  - b. Ask a colleague to be a student
  - c. Impersonate a student

## Have a go hero – Generating a timeline dynamically

In this chapter we created **static** XML to use as the source for a Javascript timeline widget. This was somewhat laborious, and you may have found it error-prone. Look back over the activities from the previous chapters. Is there a module that we can use to help us generate the timeline data dynamically? Hint: look at the core modules.

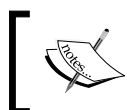
The answer is the **Database** module we looked at in *Chapter 3, Telling Stories*, can be used. As shown in the following screenshot, the names of fields will have to match the event attributes that we are familiar with—**title**, **link** and **image**. And they must all be in lower case (strictly speaking, camel-case, for example **isDuration**. No spaces). Note that the body of the event is called **description**, and the **start** and **end** dates are just **Text** fields. This gives us the freedom to add events that are far in the past, though it means we'll have to be careful about formatting.



Using this dynamic mode, there is the opportunity to collaborate with your pupils to develop the timeline.

There is an alternative syntax for the filter. Out go `title` and `dataUrl`, and in come `source` and `dataId`:

```
[Timeline]
source = mod/data
dataId = 4
date = 1870
intervalUnit = CENTURY
intervalPixels = 75
[/Timeline]
```



Note that at the time of writing, linking to the Database module is an experimental mode for the Timeline Widget filter, that is compatible with Moodle 1.9, but not with Moodle. Check the Readme file for updates.

## Summary

In this chapter, we learned a lot about creating a timeline widget to help our pupils visualize and discuss historic events. We used inventions as an example. And we used Moodle to set an assignment.

Specifically, we covered:

- ◆ What a SIMILE Timeline looks like
- ◆ Installing a syntax highlighting text editor, namely Notepad2
- ◆ Creating and checking the timeline XML
- ◆ Adding discussion forums to Moodle
- ◆ Embedding the JavaScript timeline
- ◆ Creating assignments

Now that we've learned about serious stuff like XML and history it's time to look at something lighter. Games are the subject of the next chapter.



# 6

## Fun Games

*In the previous chapter we explored a visual timeline of historical events. We are going to continue the visual theme in this chapter, and add in word and verbal reasoning. We will add and configure some fun games such as snakes and ladders for our course. These games will allow our pupils to practice their vocabulary, verbal, and reasoning skills in a fun and stimulating context.*

Specifically, we are going to:

- ◆ Introduce the Game activity module
- ◆ Create a word-source for the games—a Moodle Glossary
- ◆ Install the Game module
- ◆ Add a snakes and ladders game activity using our glossary
- ◆ Create a word search or Cryptex puzzle
- ◆ Create a hidden picture puzzle using a glossary and the question bank

So let's dive in...

### Introducing Games

Much of this chapter is centered round the contributed **Game** activity module, written and maintained by **Vasilis Daloukas** (<http://moodle.org/mod/data/view.php?rid=1196>). The module allows you to present games like snakes and ladders, crosswords, and Sudoku to your class.

Each of the games requires a source of clues or questions and words. In the case of snakes and ladders, each correctly guessed word or question results in the player moving forward through the game. Unless the player hits a snake of course, in which case they move backwards! In the case of crossword and Cryptex, the words are used directly in the grid for the game. As you can see in the table below, a Moodle Glossary activity can be used as the source for most of the games and various question types can be used for some of the games. The table is derived from one on the MoodleDocs site, at [http://docs.moodle.org/en/Game\\_module#Available\\_games](http://docs.moodle.org/en/Game_module#Available_games) (licensed under the GPL):

| Game                | Glossary | Questions    |              |              |
|---------------------|----------|--------------|--------------|--------------|
|                     |          | Short answer | Multi choice | True / False |
| Hangman             | X        | X            |              |              |
| Crossword           | X        | X            |              |              |
| Cryptex             | X        | X            |              |              |
| Millionaire         |          |              | X            |              |
| Sudoku              | X        | X            | X            | X            |
| The hidden picture  | X        | X            | X            | X            |
| Snakes and Ladders  | X        | X            |              |              |
| Book with questions | X        | X            | X            | X            |

## Creating a glossary

The **Glossary** activity is useful in itself, for example, you and your pupils can define terms relating to a subject. And it can be used in concert with other plugins. In this section, we will learn how to create a glossary to use as the source of words for the fun games we will create in later sections, using the **Game** module. Glossary is an activity module that is built into core Moodle. So there is nothing to install.

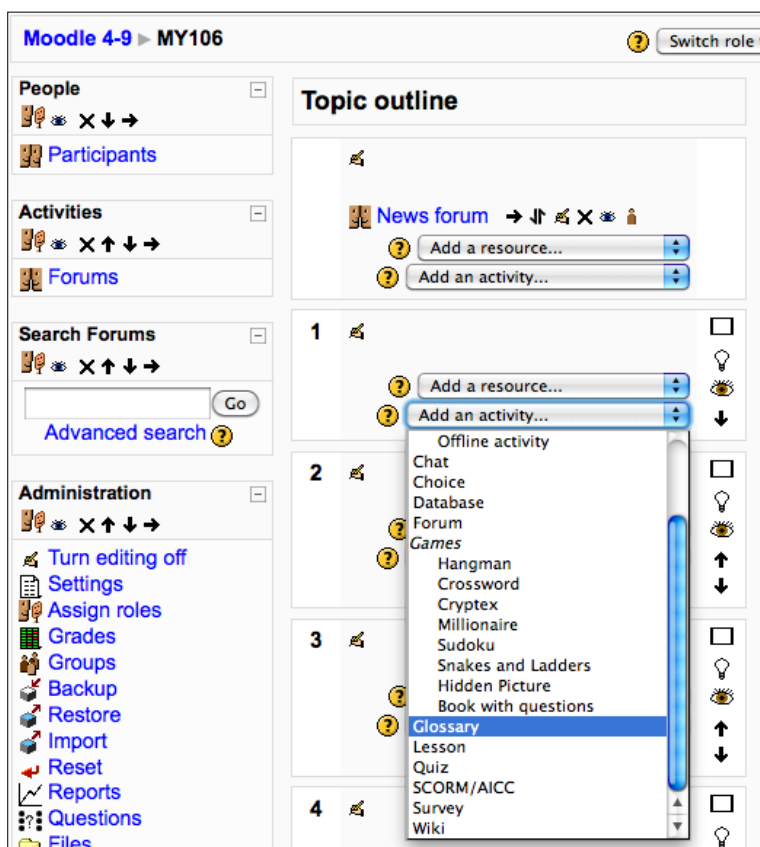
### Time for action – creating a glossary

To get started, follow these steps:

1. Log in to your Moodle site as a course creator. Follow the link from the **Site administration** side block on the home page to **Course | Add/edit courses**.
2. Create a course using the topics course format. Choose a short name like **MY106**, and a long name, for example, **Fun Games**.
3. When you are taken to the course main page, click the button in the top-right of the page to **Turn editing on**. Then choose topic **1** in the column down the middle of the page.



4. From the **Add an activity...** drop-down menu, choose **Glossary**.



To create the glossary itself follow these steps:

5. You will be taken to a page with the heading **Adding a new Glossary to topic 1**. And there will be a form like as shown in the previous screenshot.
6. Fill in the **Name** of the glossary activity, for example, **Alphabet glossary**.
7. Then under the **Description** label add some text. We can write **For use in Game activities**, as shown in the following screenshot.

You can leave the defaults for the options and drop-down menus below the **Description**. However, it is useful to note what the defaults mean.

8. There will be up to 10 **Entries shown per page** in the glossary.

9. This will be a **Secondary glossary**. You can have several secondary glossaries in a course, but only one primary one. And you can import entries from secondary glossaries to the primary one.
10. Leave the default of **No** for **Duplicated entries allowed**.
11. The next option is interesting. By default **Allow comments on entries** is set to **No**. This means that students cannot add comments, though teachers can always add comments. To make a glossary into a collaborative activity, set this to **Yes**.
12. **Allow print view** is uncontroversial—it provides students with an alternative view of a glossary entry, suitable for printing.
13. **Automatically link glossary entries** means that plain text terms from the glossary, elsewhere in the course will be replaced by links to the glossary definition. This option is useful, so it should remain set to **Yes**.
14. As we, the teacher, are providing the glossary entries, **Approved by default** can remain set to **Yes**.
15. We leave the **Display format** set to **Simple, dictionary style**. The other options, **Continuous, without author, Encyclopedia, Entry list, FAQ, Full with author, and Full without author** are less useful in our context.
16. We leave the options **Show 'Special' link, Show alphabet, and Show 'ALL' link** all set to **Yes**.
17. And the option **Edit always** can remain as **No**. This means that students will only be able to edit glossary entries that they create for a limited time.
18. Under the **Grade** section, we will leave the checkbox labeled with **Allow entries to be rated?** un-ticked.

Now, scroll to the bottom of the page and press the button labeled **Save and display**.

http://localhost - MY106: Editing Glossary


**Name\*** Alphabet glossary

**Description\*** ?

Trebuchet 1 (8 pt) Lang B I U S x x x x

For use in Game activities.

Path: body

? 

**Entries shown per page\*** 10

Glossary Type ? Secondary glossary

Duplicated entries allowed ? No

Allow comments on entries ? No

Allow print view ? Yes

Automatically link glossary entries ? Yes  
No  
Yes  
Yes

Approved by default ? Yes

Display format ? Simple, dictionary style

Show 'Special' link ? Yes

Show alphabet ? Yes

Show 'ALL' link ? Yes

Edit always ? No

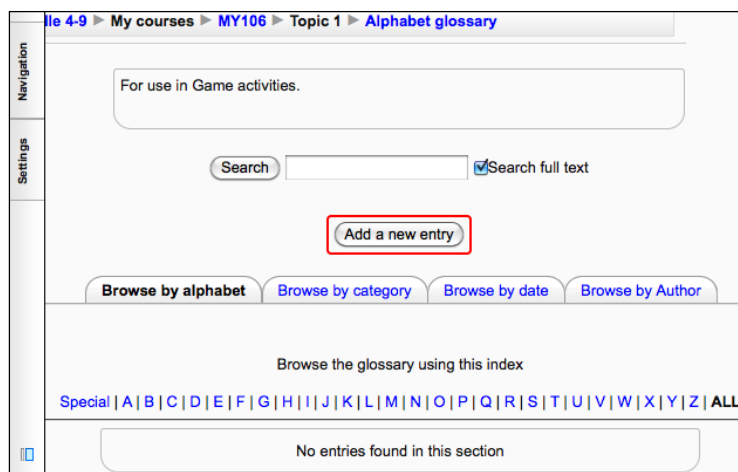
**Grade**

Allow entries to be rated? ☐ Use ratings

Users Only teachers can rate entries

## What just happened?

We created a glossary activity to use as the source of words for some of our game activities. We were able to leave the default options for many of the settings, to produce a simple glossary. As you can see from the next screenshot, there is a **Search** form, a button labeled **Add a new entry**, and a row of tab-links. The first tab, **Browse by alphabet**, is the one selected.



Under the tabs there is a row of links labeled **Special**, **A**, **B**, **C**, and so on. Based on this menu of links the meaning of some of the glossary settings, like **Show 'Special' link**, should be clearer.

Lastly, towards the foot of the page we find the text **No entries found in this section**.

## Adding glossary entries

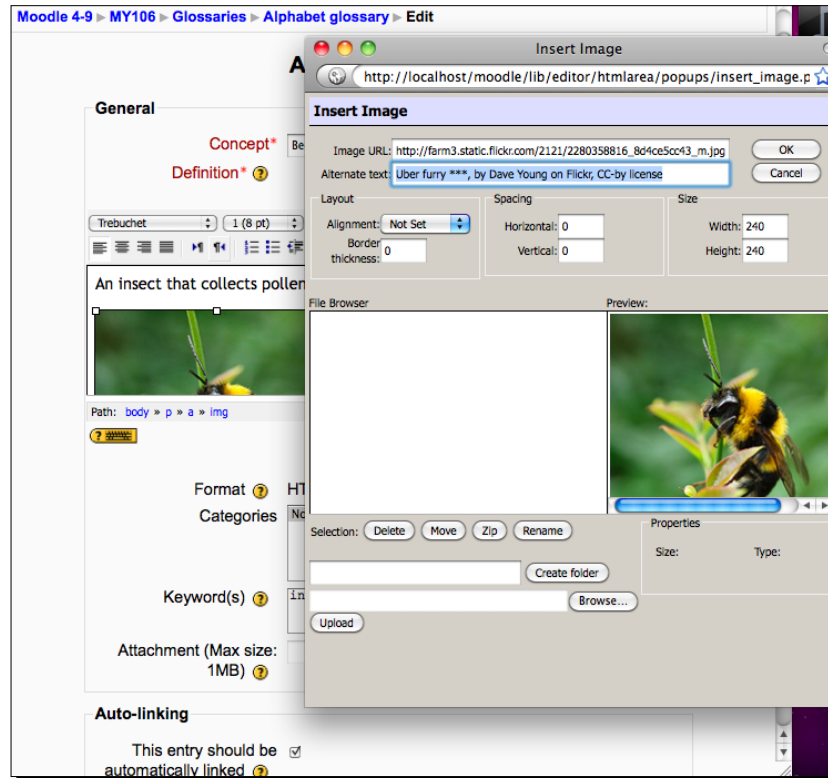
In the last section we made a great start on our glossary. We will continue to the next stage, namely adding entries to our glossary. Remember that we are creating a source of words and clues to use in games such as snakes and ladders. These will give our pupils the opportunity to learn words and so on, while having fun.

### Time for action – adding glossary entries

We are going to find some images to illustrate our glossary definitions. These are the steps required to create the entries:

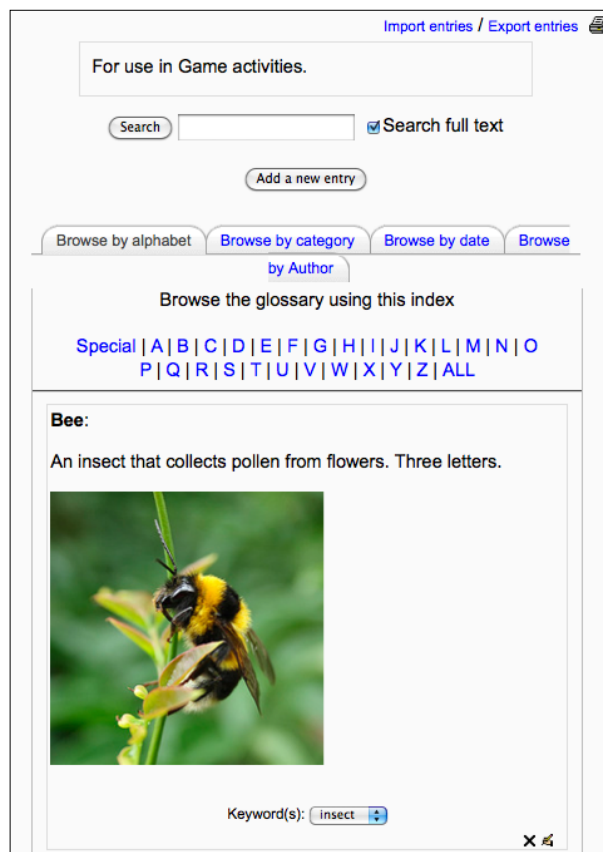
1. We will start off by searching for a **bee** using Google Image Search. I found a Creative Commons Attribution-licensed (CC-by) photograph by **dcysurfer / Dave Young** on Flickr (<http://flickr.com/photos/dcysurfer/2280358816/>). I chose the small, 240 pixel wide image, and copied the address or URL of the image to my clipboard ([http://farm3.static.flickr.com/2121/2280358816\\_8d4ce5cc43\\_m\\_d.jpg](http://farm3.static.flickr.com/2121/2280358816_8d4ce5cc43_m_d.jpg)).
2. As shown in the previous screenshot, in your empty glossary activity click on the button labeled **Add a new entry**. In the form that appears, enter the text **Bee** next to the label **Concept**.
3. In the **Description** editor, type a clue. I entered **An insect that collects pollen from flowers. Three letters**. Then after a newline, I inserted an image inside a link, using the image address copied above. The following screenshot shows what the **Insert Image** dialog window looks like. The link points to the page on **Flickr**.
4. Below the **Description** there are several other options. The **Categories** option will remain as **Not categorized**.
5. If you wish you can add a **Keyword**. I typed **insect**.
6. In the **Auto-linking** section tick the checkbox labeled **This entry should be automatically linked**.

7. The two remaining checkboxes can remain un-ticked. Press the button to **Save changes**.



### ***What just happened?***

We created a glossary entry, with a concept or term and a definition. And we found an image to illustrate the entry, in this case an open content photograph from **Flickr**. The result of our endeavors is in the following screenshot. We will see that for the games to work we should not mention the term that we are defining in the description. That would give the game away!



As we only have one entry so far, it will appear below the menu of alphabet links. I went on to add further entries, like so:

- ◆ **Ant**, Bearded Lady, an image by Jurvetson, <http://flickr.com/photos/jurvetson/18238401/> ([http://commons.wikimedia.org/wiki/File:Ant\\_closeup.jpg](http://commons.wikimedia.org/wiki/File:Ant_closeup.jpg)) (CC-by-2-gen)
- ◆ **Cloud**, Rain Cloud, an image by Barto, <http://flickr.com/photos/barto/59308568/> (CC-by-2-gb)

Each of the photographs listed above is hosted on Flickr, and is available under a Creative Commons Attribution license.

## Glossary auto-linking

One of the key features of a Moodle Glossary is that links to definitions can be automatically inserted where the terms appear elsewhere in the course. Before we continue with our Game activities we will demonstrate auto-linking.

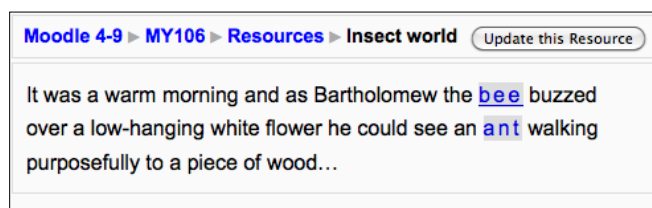
### Time for action – using auto-linking

To try out auto-linking, we will follow these steps:

1. Return to the main page for your **MY106** course, and check that editing is on.
2. In topic 2, choose **Page (Compose a web page** in Moodle 1.9) from the **Add a resource...** drop-down menu.
3. In the **Name** field type a title, for example, **Insect world**. Leave the **Summary** blank and scroll down to the **Compose a web page** section.
4. Under **Full text**, write **It was a warm morning and as Bartholomew the bee buzzed over a low-hanging white flower he could see an ant walking purposefully to a piece of wood....**
5. Scroll to the foot of the page and hit the familiar button **Save and display**.

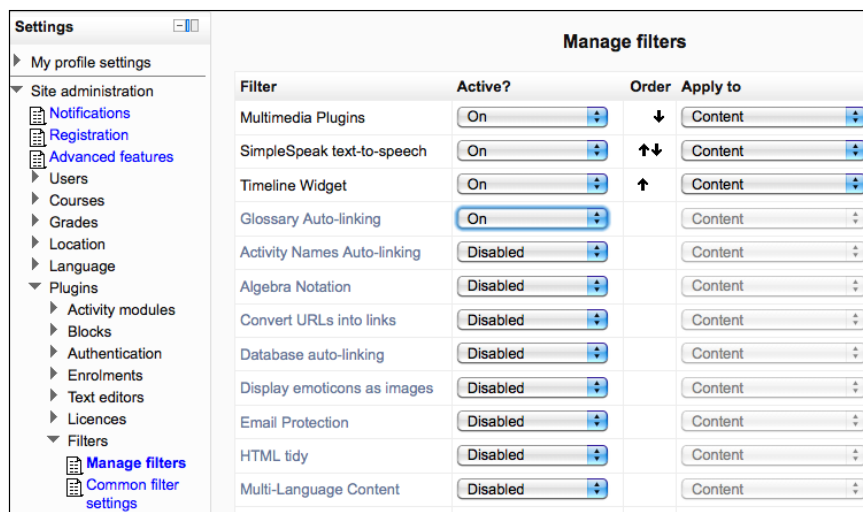
### What just happened?

When you view the resource containing words from your glossary, you will see something like the screenshot below. The words are **bee** and **ant** in this case. They automatically become links to the entries in the glossary.



If you don't see the links, you should probably enlist the help of your friendly IT support person. Ask them to check if the **Glossary Auto-linking** filter is enabled, by visiting the page at **Site administration | Plugins | Filters | Manage filters**.





After this minor detour let us return to the main theme of the chapter, namely games.

## Installing the Games module

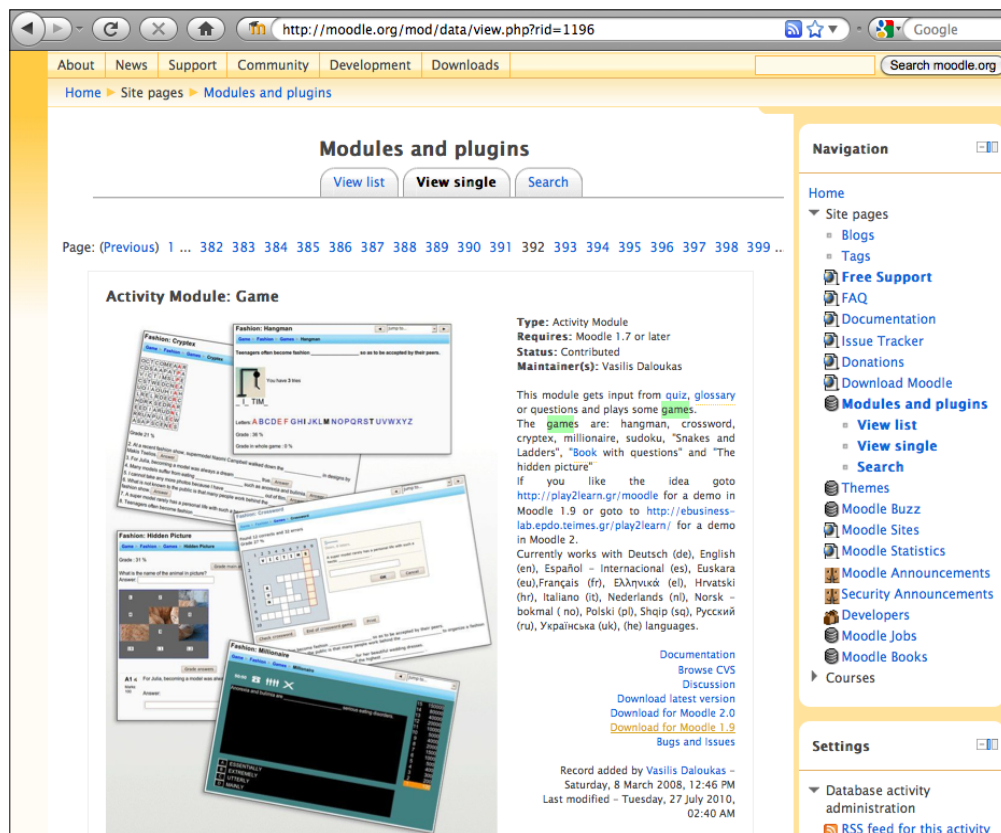
The next task is to download and install the Games module. You will need to hand over to your friendly system administrator to carry out this stage, as you probably won't have the permissions yourself.

### Time for action – installing the module

The system administrator can follow these steps:

1. Visit the fork of the **Game activity** module on Github, [https://github.com/nfreear/moodle-mod\\_game](https://github.com/nfreear/moodle-mod_game).

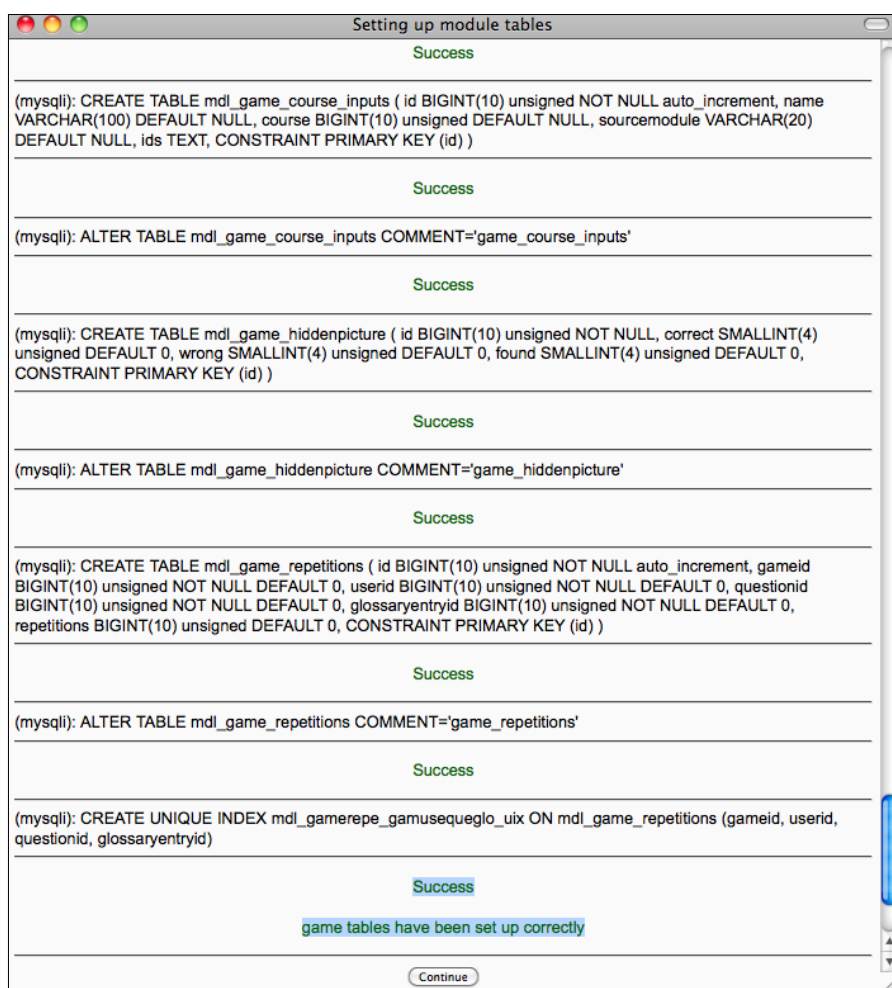
2. View the Readme file on Github, and download the version of the Game module that is appropriate for your version of Moodle. You may be directed back to the original entry in the Moodle **Module and plugins** database (shown below), <http://moodle.org/mod/data/view.php?rid=1196>:



3. Back up the Moodle database, as this module does create a number of new tables in the database.
4. Uncompress the downloaded archive somewhere convenient. Then copy the game directory to your Moodle installation; for example, on Redhat Linux you might copy the contents to `/var/www/moodle/mod/game/`.
5. Log in to your Moodle site as the admin user. Visit the page, **Site Administration | Notifications**. In Moodle 2 you will see a list of all plugins, with game listed as **about to be installed**. Press the **Upgrade** button at the bottom. (In Moodle 1.9, the system will warn you that the module is about to be installed. Press the button.) Database installation will commence.

At the time of writing, the **Game** installation creates 21 tables, with names from `mdl_game` to `mdl_game_sudoku_database` (where `mdl_` is your database table prefix).

When you hit the button there will be a pause. Moodle 2 should display the single word, **Success**, while in Moodle 1.9 you will see a page like the one below, with a log of every SQL `CREATE TABLE` statement, `ALTER TABLE` statement, and so on:



## What just happened?

We downloaded the contributed Game module from **Github** or the **Modules and plugins** database on `Moodle.org`. Our friendly system administrator installed the game module on our behalf, and the system created a number of database tables for the module. We are now ready to use the module to create some fun activities for our class.

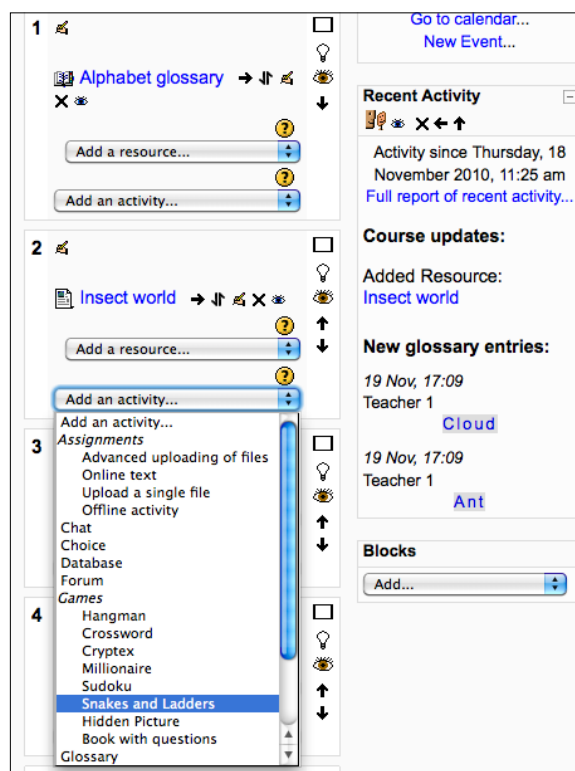
## Creating a game

We're making fine progress. We now have all the pieces in place—a glossary activity to provide some words and clues, and the Game module. It's time to add the first game to our course. This will enable our students to practice their verbal and spelling skills while having fun.

### Time for action – adding snakes and ladders

Adding a **Game** module activity to your course is a straightforward process. Here are the steps that you'll want to follow:

1. Return to the main page of your course, **MY106**, and ensure that editing is on (if the button at the top-right of the page, below the **Logout** link, says **Turn editing on**, then press it).
2. In the center column, under topic **2**, click on the **Add an activity...** drop-down menu. Squeezed between **Forum** and **Glossary**, you will see the available **Games**, eight in total, starting with **Hangman** and finishing with **Hidden Picture (Book with questions in Moodle 1.9)**. Select the second from the bottom, **Snakes and Ladders**, now.





Note that in the previous screenshot you can see each activity and resource as a link with an icon under the topic numbers, 1, 2, 3, and so on. And in Moodle 1.9, in the right-hand column under the heading **Recent Activity**, the **New glossary entries** and other activity will be visible.

3. You will land on a form with the heading **Adding a new Game to topic 2**. Under the **General** section, the **Name** field has been pre-filled with the text **Snakes and Ladders**. Leave this and move on.
4. As we don't have quizzes and questions in this course, the only available **Question source** in the drop-down menu will be **Glossary**. And against the next label, **Select glossary** is our single glossary, which we called **Alphabet glossary**. Simple!
5. Moving on to the **Grades** section, we can leave the **Maximum grade** field blank for this activity to be ungraded.
6. In the section labeled '**Snakes and Ladders**' options, change the default **Background**, from **8x8 - 4 Snakes - 4 Ladders**, to the smaller **6x6 - 3 Snakes - 3 Ladders**.
7. All that remains is to add some instructions in the rich-text editor labeled **Text at the top**. I entered the following:

*Read the clue below and guess the word. Type your answer in the answer box, then press the button. If you're correct you move the number of places shown on the dice.*

*Ladders allow you to climb up. Land on a snake and you will slide back down!*

At this stage your form will look like the one shown in the following screenshot:

**Adding a new Game to topic 2**

**General**

Name\* Snakes and Ladders

Question Source Glossary

Select glossary Alphabet glossary

Select glossary category Alphabet glossary

Select question category

Include subcategories No

Select quiz

**Grades**

Maximum grade

Grading method Highest grade

**'Snakes and Ladders' options**

Background 8x8 - 4 Snakes - 4 Ladders

**Header/Footer options**

Text at the top

Text at the bottom

Read the clue below and guess the word. Type your answer in the answer field, then press the button. If you are correct you will move forward the number of places shown on the dice.

Ladders allow you to climb up. Land on a snake and you will slide back down!

8. You can leave **Text at the bottom** blank. Scroll past the **Common module settings** and click the button to **Save and display** the activity.

You will be taken mostly to an empty page, with the heading **Snakes and Ladders**. Press the button **Attempt game** in Moodle 2.

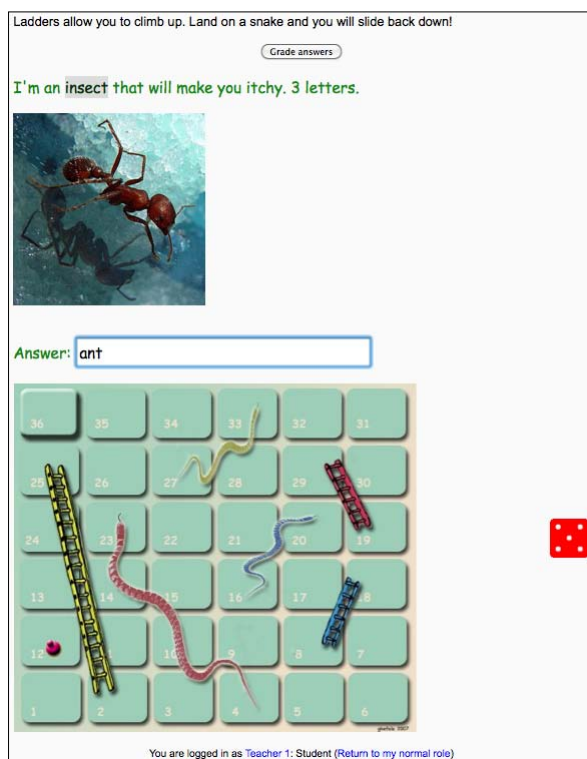
In Moodle 1.9, you will have to impersonate a student to try things out. To do this, go to the main page for the course, **MY106**, and select **Student** from the **Switch role to...** drop-down menu in the top-right. Then return to the **Snakes and Ladders** game. You will see a button labeled **Attempt game now**.

## What just happened?

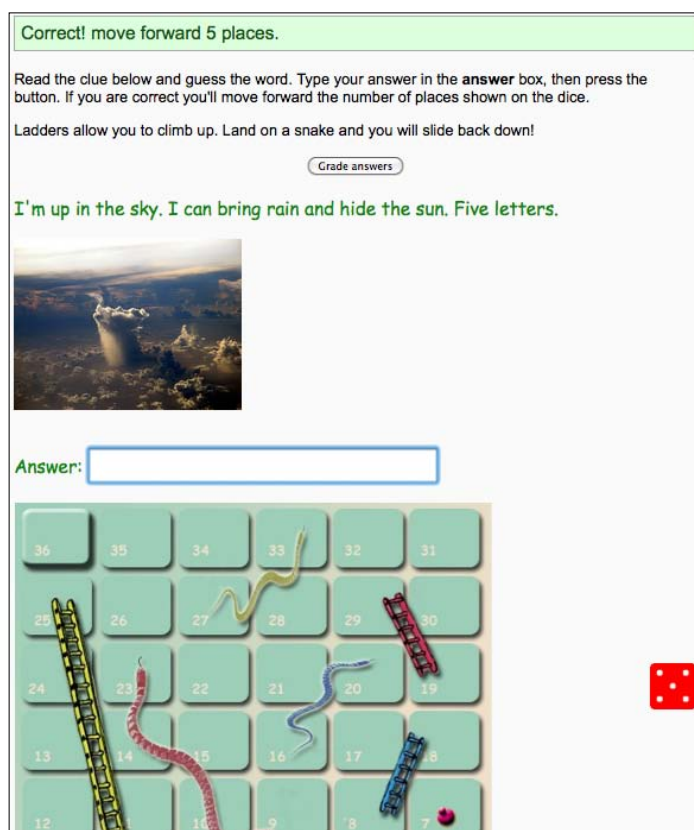
We created a **Snakes and ladders** activity and chose the **Alphabet glossary** that we created earlier as the source of clues and questions. We wrote some instructions to appear above the game.

Clicking on the **Attempt game now** button results in a page as shown in the following screenshot. The page contains the following features:

- ◆ Near the top you will see the instructions that you entered, starting **Read the clue below and....**
- ◆ Next you will see a button labeled **Grade answers**.
- ◆ Then comes a clue including our image, selected in a pseudo-random manner from the glossary word-source.
- ◆ Below the clue is a text field labeled **Answer**.
- ◆ Below and to the right is a red dice with a random number of spots.
- ◆ And to the left is the snakes and ladders board, with 36 numbered squares. The pink player marker is at square 12 in the bottom left, and you can see three snakes and three ladders:



To play, type your answer into the **Answer** field, and press the **Grade answers** button. If you are correct the system will move you forward based on the number of spots on the dice, and taking account of any snakes or ladders you may land on. This is the situation shown in the following screenshot:



Of course, you will need to add more than three words to your glossary. You may need 8-10 clues to make the game playable.

And there we have it! Snakes and ladders is our first game. It will provide a stimulating environment for our students to practice their word and spelling skills.

## Adding a cryptex

Now that we have a glossary and our first game under our belt, we can try our hand at a cryptex or word search puzzle. As we will see, it does not take much extra work to add a second game. And, this will be another fun opportunity for our pupils to practice their vocabulary and word skills.



## Time for action – adding a word search puzzle

Our first task before we create a cryptex puzzle is to add further words to our glossary.

1. Return to the main page for our course (in Moodle 1.9, if you haven't already done so, press the button to **Return to my normal role**). Click on the link in topic 1 for the **Alphabet glossary**.
2. Press the **Add a new entry** button. I added entries for **dinosaur**, **elephant**, and **umbrella**. To make the best use of the glossary, remember each time to tick the box labeled **This entry should be automatically linked**.  
Again return to the main page for the **MY106** course. And follow these steps to add the cryptex:
3. Go to topic 2, and select **Cryptex** from the **Add an activity...** drop-down menu.
4. You may wish to make the **Name** of the activity less cryptic! For example, change the default **Cryptex** to **Hidden words (cryptex)**.
5. Again, leave the selected options for **Question Source** and **Select glossary**.
6. In the section **Cryptex options**, you can set the **Maximum number of columns/rows in cryptex** to 8. This may be sufficiently difficult for your class.
7. You can leave the remaining options blank.
8. For the **Text at the top** type something like, **Can you find the hidden words in the grid below? Press the buttons to give your answers**.

9. As ever, scroll to the bottom of the page and press **Save and display**.

**Name\*** Hidden words (cryptex)

Source of questions: Glossary

Select glossary: Alphabet glossary

Select category of glossary: Alphabet glossary

Select question category: No

Include subcategories: No

Select quiz: No

**Grades**

Maximum grade: 8

Grading method: Highest grade

**Cryptex options**

Maximum number of cols/rows in cryptex: 8

Maximum number of words in cryptex: 8

Allow spaces in words: No

Max tries: 8

**Header/Footer Options**

Text at the top of page: Can you find the hidden words in the grid below? Press the buttons to give your answers.

10. As with the snakes and ladders game, in Moodle 1.9, you will have to return to the course main page and **Switch role to...** a Student. Do this now. Returning to the **Hidden words** puzzle, press the button to **Attempt game now**.

## What just happened?

Click on the button labeled **Attempt game now**, and you will be presented with the page containing the puzzle. There are a number of elements to note:

- ◆ Near the top is the familiar Moodle breadcrumb trail.
- ◆ Beneath this are the instructions you entered in the **Top text** field for the cryptex puzzle.

- ◆ Next comes the cryptex grid. Correctly identified words are highlighted in red.
- ◆ Below the grid is the **Grade** or score, and beneath that is a list of clues, which in our case come from glossary entries.
- ◆ Press the **Answer** button below each clue and a form will appear next to the grid as shown in the figure below. If a word is entered correctly, then it will be highlighted when the page is refreshed.

As you can see in the following screenshot, **CLOUD** has been correctly answered, and the person trying the puzzle is in the process of typing in **DINOSAUR**.


Moodle 4-9 ► MY106 ► Games ► Hidden words (cryptex)

Can you find the hidden words in the grid below? Press the buttons to give your answers.

|   |   |   |   |   |
|---|---|---|---|---|
| O | U | D | I | I |
| D | D | I | N | N |
| C | U | N | O | D |
| C | L | O | U | D |
| U | D | S | D | I |
| U | D | A | N | T |
| D | O | U | I | I |
| D | S | R | D | N |

Grade 36 %

1. I'm an insect that will make you itchy. 3 letters.



Answer

2. I used to roam the earth. Now I'm extinct. 8 letters.

Answer

I used to roam the earth. Now I'm extinct. 8 letters.

DINOSAUR

OK Cancel

You are logged in as Teacher 1: Student ([Return to my normal role](#))

So there we have it. We have created our second game, which will hopefully help our students learn while having fun.

## Hidden picture

The third game we are going to look at from the Game module is the **hidden picture** puzzle. We are going to mix things up a bit, and use the following components:

- ◆ A **quiz** will be the source of clues and questions
- ◆ A **glossary** will hold the hidden picture itself

Let's get started.

### Time for action – adding a hidden picture

The first step is to assemble some images. I found these images on **Geograph** <http://geograph.org.uk>, which is an open online community whose aim in the words of their tag line is to *photograph every grid square* in the British Isles. As you can tell, this section is going to have a geographical flavor!

All photographs contributed to Geograph are available under a **Creative Commons Attribution-Share Alike** license. The images are:

1. Heavy Horse, photograph by **Chris Upson**, sculpture by **Andy Scott** (NS6765, Baillieston), <http://geograph.org.uk/photo/87389>.
2. Phoenix in Easterhouse, photograph by **Chris Upson** (sculpture by unknown) (east of [1], NS6865, Lockwood. Aberdalgie and Easterhouse Roads), <http://geograph.org.uk/photo/128690>.
3. Provan Hall, by **G Laird** (north-west of [1], NS6666, Stepps, North Lanarkshire), <http://geograph.org.uk/photo/1327849>.
4. St. Bridget's Church, Baillieston, by **Emma Mykytyn** (south of [1], NS6764), <http://geograph.org.uk/photo/734368>.

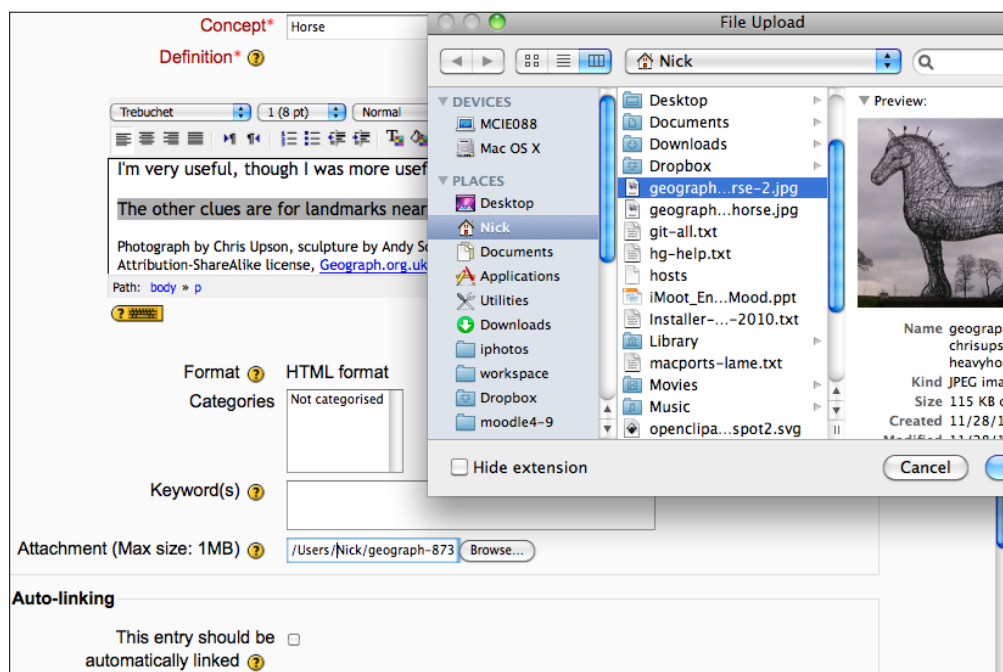
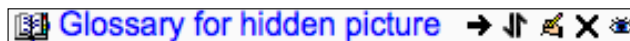


We are going to use the first image in the list above as the hidden picture, so save a copy of the full-size image for this photograph to your computer. In the case of the other photographs, save the smaller images, which are 120 pixels wide.

Then, just follow these steps to add a new glossary:

1. Return to the main page for the **MY106** course. Click the **Turn editing on** button.
2. Under topic **3**, select **Glossary** from the **Add an activity** drop-down menu.
3. In the form, add a **Name**, for example, **Hidden picture**. Type a description, and leave the default values for all the other fields. Scroll to the bottom and press **Save and display**.
4. When the glossary page appears, click on **Add a new entry**.
5. Enter the text for the **Concept**, which is simply **Horse**.

6. Enter a clue, explanation, and the attribution in the **Definition**. I typed, **You can ride me. I'm very useful, though I was more useful in the past. What am I? / The clues below are for landmarks near a sculpture of me / Photograph by Chris... Geograph.org.uk.** (/ indicates a paragraph-break.).
7. Leave the **Keyword(s)** field blank.
8. Move to the **Attachment** field, browse your computer and select the full-size photograph for **Heavy horse** that you saved from the Geograph website.
9. Leave the checkboxes for **Auto-linking** un-ticked. And click on the button to **Save changes**.
10. On the course main page click on the **eye icon** next to the **Glossary** that you have just created. The icon has the tooltip **Hide** and is to the right of the row of icons.



## ***What just happened?***

We sourced photographs relating to a geographical area, from the Geograph website. In our case, the location is Baillieston near Glasgow in Scotland. If there are well-known landmarks near your school you can use those, and make the clues and possibly the answer specific to the location.

Then we created a glossary containing a single entry. We added the large photograph from the landmark, which will be the hidden picture in our game. It was uploaded to Moodle. We added a clue, which is purposefully fairly tricky, and an explanation of the other clues. Note the attribution text does not mention Heavy horse or link directly to the page containing the photograph. This would give the game away!

Also note that we hid this glossary from our pupils, as it is essentially just a container for the main hidden picture image and clue.

There, we've completed the first step towards creating a hidden picture puzzle to keep our pupils pondering.

## **Hidden picture questions**

We are going to create some multiple choice questions. These will form the clues for our hidden picture puzzle.

### **Time for action – adding questions**

To create the questions just follow these steps:

- 1.** Return to the main page for the course, **MY106**.
- 2.** In the **Settings** side block (**Administration** side block in Moodle 1.9), click on the **Course administration | Questions bank** link. You will be taken to the **Question bank** page for the course.
- 3.** The **Choose a question type to add** dialog will appear in Moodle 2. Select the **Multiple choice** radio button and press the **Next** button. (In Moodle 1.9, choose **Multiple Choice** from the **Create a new question** drop-down menu towards the middle of the page.)
- 4.** Fill in the **Question name** field, with **Phoenix in Easterhouse**.
- 5.** Enter a **Question text** and upload the relevant photograph that we saved from **Geograph** earlier.

6. Under **Choice 1**, enter a correct **Answer**, for example, **The phoenix is a legendary firebird found in Persian, Chinese, Greek... mythology**. Set the **Grade** for this response to **100%**, and enter some **Feedback**. For example, **Correct, good choice!** Look at the example in the next screenshot.
7. Under **Choice 2**, enter an incorrect **Answer**, for example, **The phoenix is a bird that lays its eggs on cliffs by the sea**. Leave the **Grade** set to **None**, and type in some **Feedback**. I put, **Woops, that's wrong. Try again.**
8. Scroll to the foot of the page, and click on the button to **Save changes**.
9. Add your other questions based on the remaining photographs. For Provan Hall, you should write a correct statement, for example, **Provan Hall is regarded as the best-preserved medieval fortified country house in Scotland. Parts of it probably date from the 1460s**. And I entered an incorrect statement; **Provan Hall is a well-preserved example of a Victorian school**.

The screenshot shows a quiz editor interface. At the top, there are settings for the question: "One or multiple answers?" (set to "One answer only"), "Shuffle the choices?" (checked), and "Number the choices?" (set to "a., b., c., ..."). Below these are two choice sections.

**Choice 1**

Answer: The phoenix is a legendary firebird that is found in Persion, Chine

Grade: 100 %

Feedback:

Correct, good choice!

Find out more about the [phoenix on Wikipedia](#).

Path: body

**Choice 2**

Answer: The phoenix is a bird that lays its eggs on cliffs by the sea

Grade: None

Feedback:

Woops, that's wrong. Try again.



## ***What just happened?***

We added multiple choice questions to the question bank for our course to act as clues for our hidden picture puzzle. These contain the photographs we gathered from Geograph. Note that we did not need to add a quiz to hold the questions.

## **Creating the puzzle**

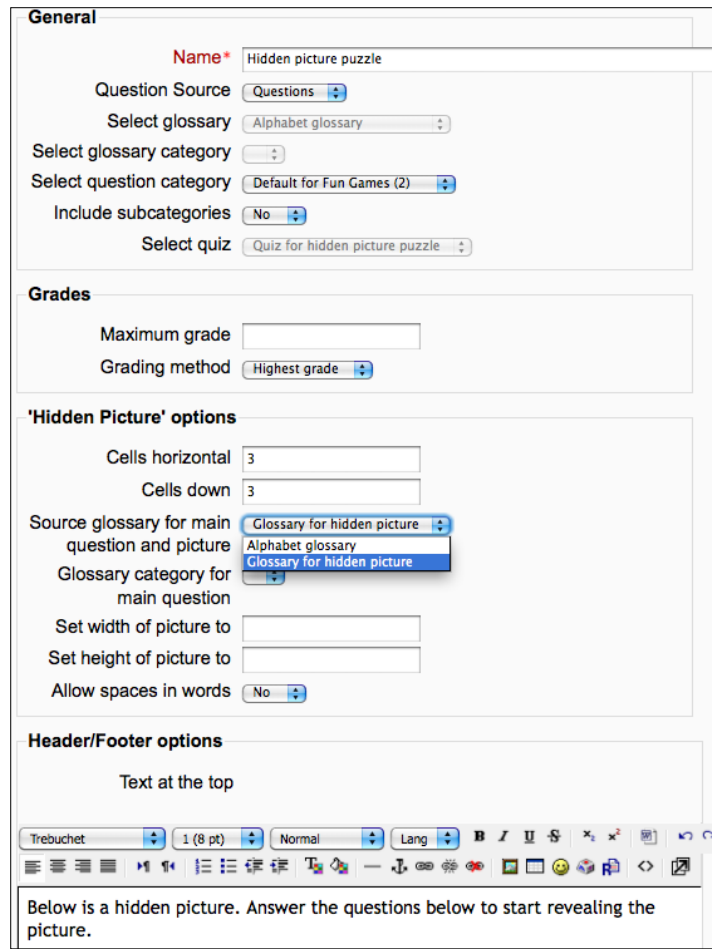
In the previous sections we assembled the components for our puzzle. Now it's time to add the hidden picture game.

### **Time for action – adding a hidden picture game**

The steps to add the hidden picture game are similar to the previous ones, but not identical:

- 1.** Go to topic 3 on the main page for our course, **MY106**. Choose **Hidden Picture** from the **Add an activity** drop-down menu. It is near the bottom of the **Games** group of options.
- 2.** You will be presented with a form headed **Adding a new Game to topic 3**. You may wish to adjust the **Name** of the game slightly. For example, you can put **Hidden picture puzzle**.
- 3.** Change the **Question Source** drop-down menu from Glossary to **Questions**.
- 4.** Against **Select question category**, the only available category, **Default for Fun Games**, will be selected already. You will see the number of questions in the category in brackets.
- 5.** Leave the rest of the **General** options, and jump over the **Grades** section.
- 6.** Under '**Hidden Picture**' options, keep the default of **3** for now. You may wish to adjust the number of **Cells horizontal** and **Cells down** later.
- 7.** Change the **Source glossary for main question and picture** from **Alphabet glossary** to **Glossary for hidden picture**.
- 8.** And add some **Text at the top**. You can enter, **Below is a hidden picture. Answer the questions to start revealing the picture**. You can see the partially completed form in the following screenshot.

9. Finally, scroll to the bottom of the form and press the familiar **Save and display** button.



**General**

Name\* Hidden picture puzzle

Question Source Questions

Select glossary Alphabet glossary

Select glossary category

Select question category Default for Fun Games (2)

Include subcategories No

Select quiz Quiz for hidden picture puzzle

**Grades**

Maximum grade

Grading method Highest grade

**'Hidden Picture' options**

Cells horizontal 3

Cells down 3

Source glossary for main question and picture Glossary for hidden picture

Glossary category for main question Alphabet glossary

Set width of picture to

Set height of picture to

Allow spaces in words No

**Header/Footer options**

Text at the top

Trebuchet 1 (8 pt) Normal Lang

Below is a hidden picture. Answer the questions below to start revealing the picture.

## What just happened?

We added a hidden picture activity to our course, and linked it to the questions and the image attached to the glossary entry that we created earlier.

As with the other Game activities, to try out your hard work you will need to impersonate a pupil. Go to the main page for the course and select **Student** from the **Switch role to...** drop-down menu. When you return to the Hidden picture puzzle and click on the button **Attempt game now**, you will see something like that shown in the following screenshot:

Grade : 33 %

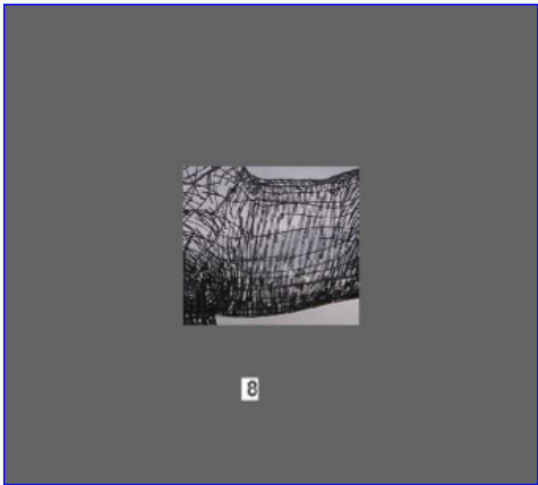
[Grade main answer](#)

You can ride me. I'm very useful, though I was more useful in the past. What am I?

The clues below are for landmarks near a sculpture of me.

Photograph by Chris Upson, sculpture by Andy Scott (NS6765, Baillieston), Creative Commons Attribution-ShareAlike license, [Geograph.org.uk](http://Geograph.org.uk).


Answer:



[Grade answers](#) [End of game](#)

**A8**  
Marks: 100

Look at the photograph below, and choose the statement below that you think is correct.



Choose one answer.

☒ a. The phoenix is a legendary firebird that is found in Persion, Chinese, Greek, Roman and Egyptian mythology

☐ b. The phoenix is a bird that lays its eggs on cliffs by the sea

You will see these components:

1. Near the top of the page, below the breadcrumb trail is your introduction, **Below is a hidden picture....** Next comes a button to **Grade main answer**—that is, to submit our attempt at the hidden picture.
2. Then we have the clue from the glossary entry for the hidden picture. **You can ride me. I'm very useful, though I was more useful....**

3. Next we have a form field labeled **Answer**. If the pupil enters the correct text here and presses **Grade main answer**, the hidden picture is revealed, and they have completed the puzzle.
4. Then we have the hidden picture with the missing bits labeled with a number, which refers to a clue below the picture. For example, there is a label **8** in the figure above. In the above example, a part of the hidden picture has been revealed.
5. Below the hidden picture are buttons labeled **Grade answers** and **End of game**.
6. Then we have the remaining clues. In this case they are multiple choice questions, sourced from the course question bank.

It is useful to note that to use the hidden picture game in the classroom we would need at least as many questions or clues as we have squares in the hidden picture. In this example, there are nine squares, but we had just two clues.

And there we have it. A hidden picture puzzle to keep our students thinking and guessing.

### Pop quiz

Try these quick questions to test your understanding. Careful! There may be more than one correct answer.

1. What are the possible word and clue sources for the Game module?
  - a. Database module
  - b. Glossary activity
  - c. All available question types
  - d. Moodle course
  - e. Some core question types including multiple choice and short answer
2. What is the cryptex activity?
  - a. A hidden picture puzzle
  - b. A hidden word puzzle
  - c. A cryptography game
3. How do you define the picture for a hidden picture puzzle?
  - a. You embed an image in a resource
  - b. You provide the URL or address of the image
  - c. You upload an image as an attachment to a glossary entry

## Have a go hero

In this chapter, we worked with three of the games that comprise the Game module. Look at the table at the front of the chapter again. Choose one of the remaining activities: hangman, crossword, millionaire, Sudoku, or book with questions. Perhaps choose a theme for the game, and use one of the other question types as the source of games. Create the game. Finally, give your game a go! Was it fun?

## Summary

We covered a lot in this chapter about integrating games with Moodle.

These are the specific things we learned:

- ◆ We introduced the third-party **Game** module.
- ◆ We developed an animal-alphabet glossary with photographs to use as the source of words and clues.
- ◆ We learned how to create a snakes and ladders puzzle using the glossary.
- ◆ We created a word search or Cryptex puzzle, using the same alphabet glossary.
- ◆ We developed a hidden picture puzzle. To do this we sourced some images around a geographical theme, created multiple-choice questions, and a glossary to hold the hidden picture.

We also discussed how to automatically link words that are defined in our glossary, when they appear elsewhere in the course text. And we added another interesting source of free images to our armory, namely **Geograph**.

Through using these games, your class will develop their spelling, verbal, and visual skills. And above all, they will learn while having fun.

Now that we've learned about games that are tightly integrated with Moodle, it's appropriate to tackle some puzzle activities that are less integrated. They are more interactive. Flash puzzles are the topic of the next chapter.



# 7

## Interactive Puzzles

*In the last chapter, we used the contributed Game plug-in for Moodle to create games such as snakes and ladders. In this chapter, we are going to continue the game and puzzle theme. The puzzles in this chapter will be a little more interactive. They will be generic activities, not specifically tied to Moodle. They will help your class practice using the mouse, and test their literacy and visual capabilities.*

In this chapter we will:

- ◆ Introduce a Flash-based word search puzzle
- ◆ Plan and create a word search XML file
- ◆ Create the word search puzzle in a Moodle course
- ◆ Introduce a jigsaw puzzle
- ◆ Source images for our jigsaw
- ◆ Create the jigsaw in Moodle
- ◆ Explore how to use these puzzles in ICT teaching

Throughout this chapter we will be using the **Adobe® Flash® Player** plug-in that is installed in most Web browsers. Visit this address, <http://adobe.com/software/flash/about/>, on the Adobe site now to test your Adobe Flash plug-in.

There is our plan. Let's get the show on the road.

## An alternative word search

The word search puzzle that forms part of the Game module is an interesting option, particularly if you wish to collect grades or scores while your class plays. However, if you wish to have your class actually interacting with the grid using a mouse, there are other options.

One such option is **Duncan Keith's Word Search** puzzle (<http://subtangent.com/flash/>). This uses the Adobe Flash plug-in, installed in most web browsers, and is available under both the **Creative Commons Attribution – NonCommercial – ShareAlike 1.0 License**, and the **GNU General Public License**. The first screenshot shows the sort of puzzle that we will be creating:



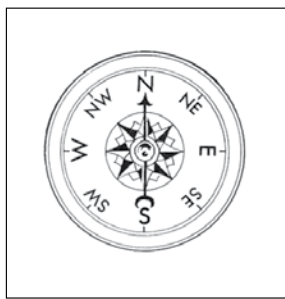
## Time for action – planning the word search

We are going to create a word search puzzle containing names of shapes. The first step is to plan our puzzle, using a spreadsheet software application. I used **Microsoft Excel**, but if you don't have access to this you can use Calc from **OpenOffice**, which can be downloaded free for Windows and Mac OS X from <http://openoffice.org/>.

1. Start with a blank worksheet, and add row and column labels from **0** to **7**, as shown in the next screenshot.
2. Start filling in words on the horizontals, verticals, and diagonals. Let's use seven words—**square**, **star**, **circle**, **triangle**, **diamond**, **ring**, and **kite**. Ensure that any intersections between words make sense. For example, circle and triangle can intersect at their **Es** or their **Rs**.

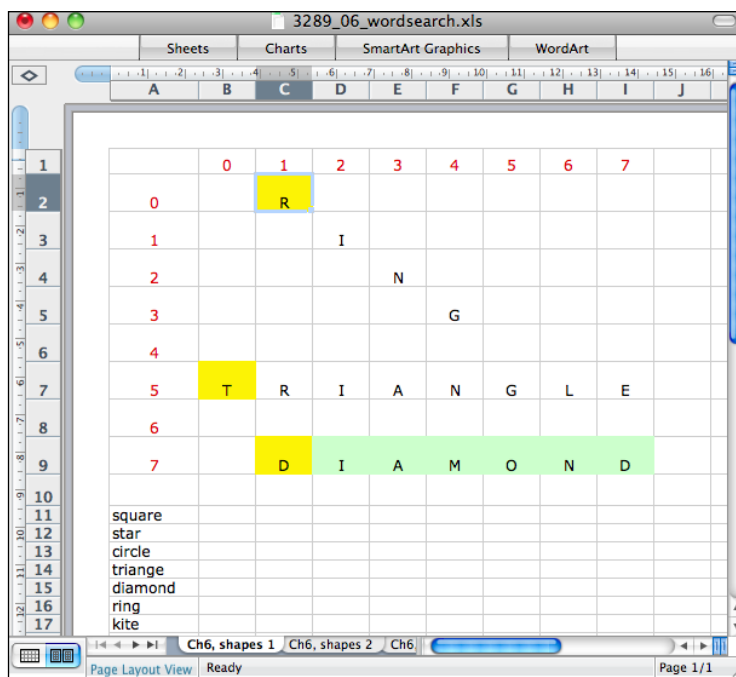


3. Note, by using compass orientations, we are putting words in easterly (the word **diamond**), southerly, and south-easterly (**ring**) directions. We won't put words in the more difficult directions such as north, west, and south-west. Refer to the following compass to help you:



The previous compass rose is Public Domain and sourced from Wikimedia Commons ([http://commons.wikimedia.org/wiki/File:East\\_1\\_%28PSF%29.png](http://commons.wikimedia.org/wiki/File:East_1_%28PSF%29.png)).

4. When you have put all your seven words in, fill in the blanks on the grid in our spreadsheet:



## ***What just happened?***

We created an 8 by 8 grid in a spreadsheet software package. We will find the numbering from 0 to 7 (as opposed to 1 to 8) to be useful in the next section. Then we chose seven fairly simple words for shapes, with the longest being eight letters. We put them in the grid, in a variety of directions and taking note of any intersections.

This is the first stage towards creating a fun mouse-driven puzzle for our students. Now that we have planned the activity, let's create the XML file for the online puzzle.

## **Creating the word search XML**

The next stage in generating our word search puzzle is to create an XML file. We will use the free **Notepad2** text editor we introduced in the previous chapter. If you need to install it, refer back to the instructions in the *Installing a text editor* section in *Chapter 5, Setting Homework*.

### **Time for action – creating the XML**

To create the XML for our word search puzzle, follow these steps:

- 1.** Enter the XML prologue, and the opening and closing tags for the root element `<?xml version="1.0" encoding="utf-8"?> <wordsearch> </wordsearch>`. Remember that you can add new-lines and extra space—these are not significant in XML.
- 2.** Within the `<wordsearch>` element create the three elements `<title>` containing text (for example, **2-D Shapes**), `<grid>` and `<words>`. Add `width` and `height` attributes to the opening `<grid>` element, both with values of 8.
- 3.** Add `<gridline>` opening and closing tags to the `<grid>` element. Copy the first line of eight letters from the spreadsheet, as shown in the next example file. I entered **BRSFPHRC**. Repeat for the next seven rows.
- 4.** Add `<word>` opening and closing tags to the `<words>` element. Enter one of the solution words between the opening tags. For example, put **square** below. Then add the `x`, `y`, and `dir` attributes. `dir` is short for direction, and refers to the compass direction we used earlier. So, use abbreviations such as **SE** for south-east, **E** for east, and so on. `x` and `y` refer to the horizontal and vertical coordinates respectively, starting from 0,0 in the top-left corner of the grid.

5. Repeat the <word> elements for the other five solution words:

```
<?xml version="1.0" encoding="utf-8"?>
<wordsearch>
  <title>2-D Shapes - 1</title>
  <grid width="8" height="8">
    <gridline>BRSFPHRC</gridline>
    ...
  </grid>
  <words>
    <word x="2" y="0" dir="SE">square</word>
    ...
  </words>
</wordsearch>
```

## What just happened?

We used the spreadsheet that we completed in the previous section to create the word search XML file. The format of this file is hopefully fairly straightforward, with two sections specifying the word search grid and the list of solution words. The only tricky part is perhaps the direction and X-Y coordinates for the solution words.

As you edit the XML file you will want to check the validity of the XML file in a browser such as Firefox or Internet Explorer. This will help you spot errors before we move on to the next stage. If you run into XML errors refer back to the suggested fixes in the *Troubleshooting* section in *Chapter 5, Setting Homework*.

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <wordsearch>
3   <title>2-D Shapes - 1</title>
4   <!-- The title is mandatory -->
5   <grid width="8" height="8">
6     <!-- The width and height attributes are mandatory -->
7     <gridline>BRSFPHRC</gridline>
8     <gridline>XSIQOSII</gridline>
9     <gridline>MSQUTNR</gridline>
10    <gridline>JTRZGASC</gridline>
11    <gridline>XAKITERL</gridline>
12    <gridline>TRIANGLE</gridline>
13    <!-- ... -->
14  </grid>
15  <words>
16    <!-- Coordinates are measured from top-left (0,0) -->
17    <!-- All attributes are mandatory. Use compass dir.-->
18    <word x="2" y="0" dir="SE">square</word>
19    <word x="1" y="2" dir="S">star</word>
20    <word x="7" y="0" dir="S">circle</word>
21    <word x="0" y="5" dir="E">triangle</word>
22    <word x="1" y="0" dir="SE">ring</word>
23    <word x="2" y="4" dir="E">kite</word>
24    <!-- ... -->
25  </words>
26 </wordsearch>
```

The complete word search XML looks like this:

```
<?xml version="1.0" encoding="utf-8"?>
<wordsearch>
  <title>2-D Shapes - 1</title>
  <grid width="8" height="8">
    <!-- The width and height attributes are mandatory -->
    <gridline>BRSFPHRC</gridline>
    <gridline>XSIQOSII</gridline>
    <gridline>MSQNUTNR</gridline>
    <gridline>JTRZGASC</gridline>
    <gridline>XAKITERL</gridline>
    <gridline>TRIANGLE</gridline>
    <gridline>EDAROGOL</gridline>
    <gridline>ADIAMOND</gridline>
  </grid>
  <words>
    <!-- Coordinates are measured from top-left (0,0) -->
    <!-- All attributes are mandatory. Use compass dir.-->
    <word x="2" y="0" dir="SE">square</word>
    <word x="1" y="2" dir="S">star</word>
    <word x="7" y="0" dir="S">circle</word>
    <word x="0" y="5" dir="E">triangle</word>
    <word x="1" y="7" dir="E">diamond</word>
    <word x="1" y="0" dir="SE">ring</word>
    <word x="2" y="4" dir="E">kite</word>
  </words>
</wordsearch>
```

We are nearly at the point where we can befuddle our class with a word search puzzle. Let's upload our XML.

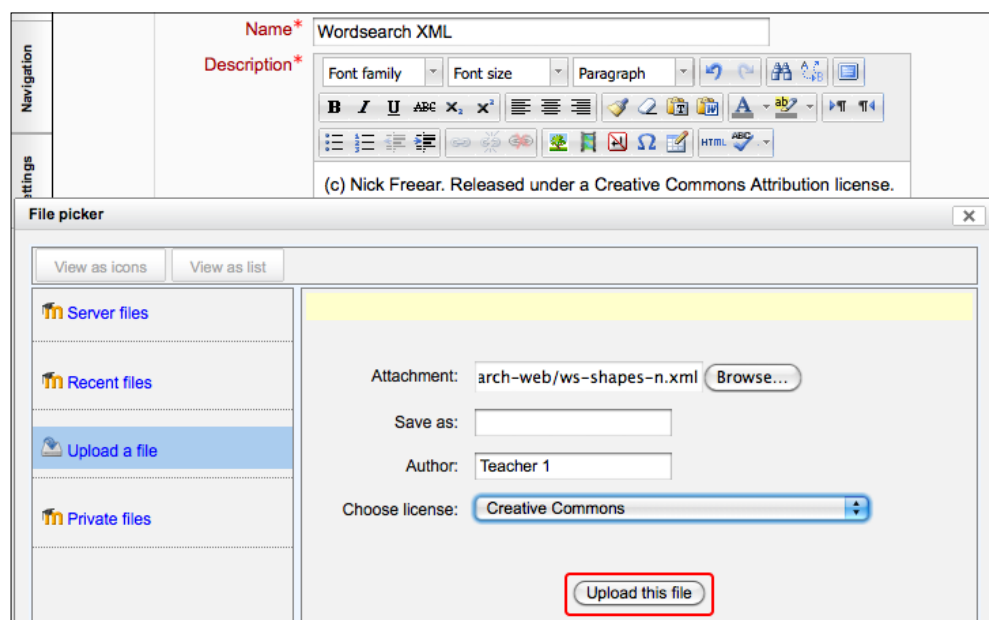
## Uploading the puzzle XML

In the previous section, we created the XML file for the puzzle. The next stage in creating a puzzle for our students is to upload the file to Moodle.

## Time for action – uploading the XML file

These are the steps that you will want to follow to upload the word search XML file to Moodle 2:

1. Log in to your Moodle system as a teacher, and create a course. In a fit of originality, we'll call the course **Interactive puzzles** and give it a **Course short name** of **MY107**. Let's use the **Topics format**.
2. In topic **1** of your newly created course, choose **File** from the **Add a resource** drop-down menu. Give the **File** resource a **Name**, for example **Wordsearch XML**, and a **Description**.
3. Under **Content** press the **Add** button. In the **Upload file** dialog that appears, choose the link labeled **Upload a file**, press the **Browse** button, find the file `ws-shapes-n.xml` on your computer, and hit the button to **Upload this file**.



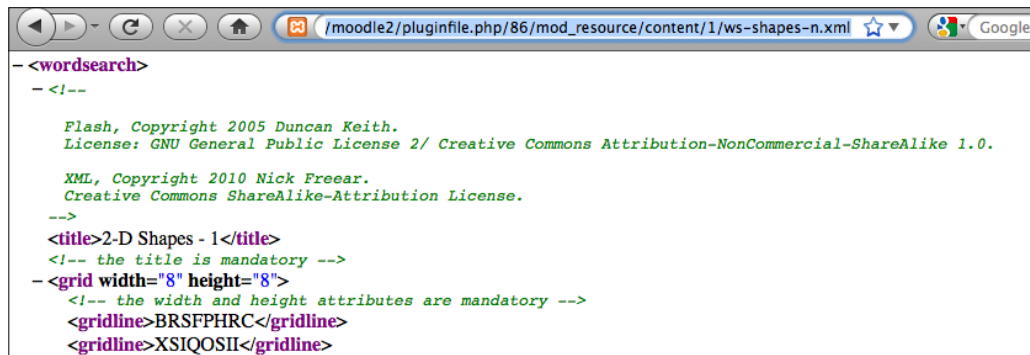
4. Scroll to the foot of the page and press the button to **Save and display**. You will land on a page displaying the name and description that you gave the resource. Under the description click on the link labeled **ws-shapes-n.xml**.
5. You will be taken to the XML file, which should look something like the next screenshot. Copy the URL from the address bar of your browser, and paste it somewhere (a text or Word document for example) for the next stage.

In Moodle 1.9 these are the steps to follow:

1. Log in to your Moodle site as a teacher, and create a course.
2. In your new course, find the **Administration** side block that is often on the bottom left of your page. Click on the **Files** link towards the bottom. In the **Files** manager page that appears, press the **Upload a file** button on the bottom right.
3. Then press **Browse**. Using the **File upload** dialog that appears, find the **word search XML** file on your computer and press the **Open** button (OK button on Windows) to close the dialog window. Press **Upload this file**, and the filename will appear in the list of files.
4. In the browser, right-click with your mouse on the file named **ws-shapes.xml**. Choose **Copy Link Location** from the right-click context menu. Paste the URL into a text file for later use.
5. Return to the main course page by pressing the **MY107** link in the breadcrumb trail at the top of the page.

## What just happened?

We created a course to hold our interactive puzzles. Then we uploaded our XML as a **File** resource (in Moodle 2). We viewed the raw XML file in our browser one last time to ensure there were no errors. And this allowed us to copy the URL for the file, for example, `/moodle2/pluginfile.php/86/mod_resource/content/1/ws-shapes.xml`. We will need it soon.



```
<?xml version="1.0"?>
<wordsearch>
  <!--
    Flash, Copyright 2005 Duncan Keith.
    License: GNU General Public License 2/ Creative Commons Attribution-NonCommercial-ShareAlike 1.0.

    XML, Copyright 2010 Nick Freear.
    Creative Commons ShareAlike-Attribution License.
  -->
  <title>2-D Shapes - 1</title>
  <!-- the title is mandatory -->
  <grid width="8" height="8">
    <!-- the width and height attributes are mandatory -->
    <gridline>BRSFPHRC</gridline>
    <gridline>XSIQOSII</gridline>
```

We have one component for our puzzle in place. It's time to create the puzzle activity for our class.

## Creating a word search puzzle

Let's bring together the word search XML file and the Flash that will run the word search puzzle. Of course, we will do this in the context of our Moodle course.

### Time for action – embedding Flash

Follow these steps to create the puzzle in Moodle 2:

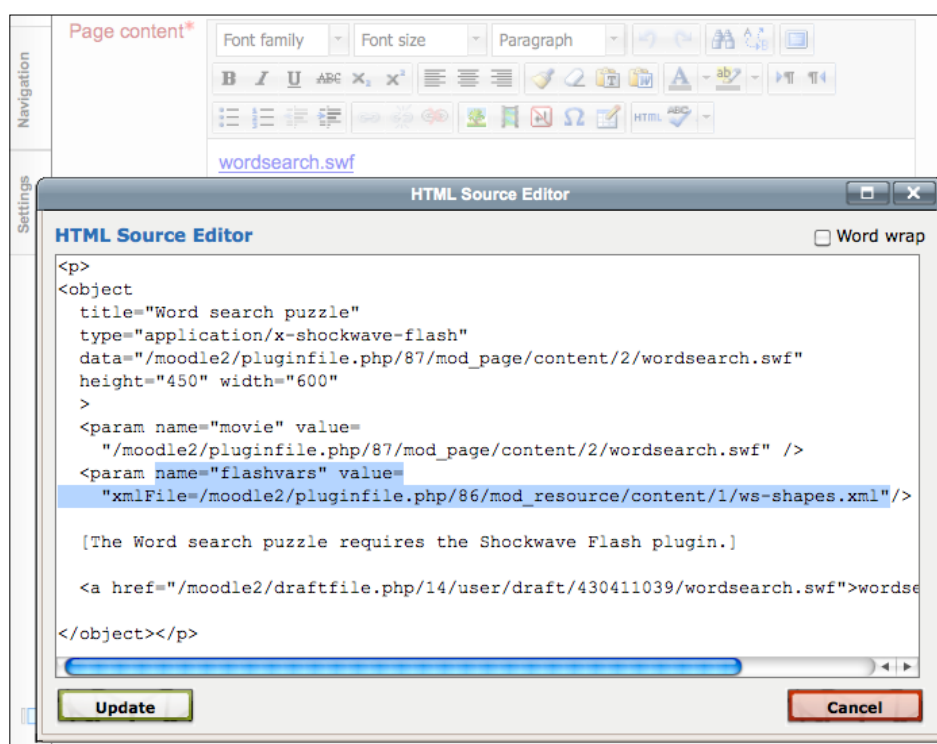
1. In your browser, visit the following address for the word search Shockwave Flash file on Duncan Keith's website: <http://subtangent.com/flash/wordsearch.swf>.
2. Save the file to your own computer. In the Firefox or Internet Explorer menu bar choose **File | Save page as** and ensure that you preserve the `.swf` file extension.
3. Return to the **MY107** course. In topic **1**, choose **Page** from the **Add a resource** drop-down menu. Give the **Page** resource a **Name**, for example **Wordsearch puzzle**, and a **Description**.
4. In the rich editor next to the **Page content** label, find and click the **Moodle Media** button, next to the **Insert/edit image**, on the bottom row (see in the following screenshot). In the dialog that appears, press the link to **Find or upload a sound, video or applet**.



5. In the second dialog, choose the link labeled **Upload a file**, press **Browse**, find the file `wordsearch.swf`, and hit **Upload this file**. In the remaining dialog press the button labeled **Insert**. Returning to the rich editor, you will see a link labeled **wordsearch.swf**.
6. Press the **HTML source** button. Start editing the raw HTML, by first adding an opening `<object>` tag before the link (`<a href=...></a>`), and a closing `</object>` tag after.
 

```
<object attributes...>
  <param name="" value="" />
</object>
```
7. Add a **type** attribute to `<object>` with the value `application/x-shockwave-flash`, and a **data** attribute with a value such as `/moodle2/pluginfile.php/87/mod_page/content/2/wordsearch.swf` (adjust to your URL). Also add a **title**, and a **height** and **width** of 450 and 600, respectively.

8. Now, inside the `<object>` element add two `<param>` elements. The first has a name of `movie` and a value attribute equal to the data attribute above. The second has a name of `flashvars` and a value attribute of `xmlFile=/moodle2/pluginfile.php/86/mod_resource/content/1/ws-shapes.xml` (again, adjust to match the URL you noted earlier. Note the `xmlFile=` prefix).
9. Beneath the second `<param>` element add some fallback text, for instance, The Word search puzzle requires the Flash plugin. Note, this step is particularly important now that there are more devices without Flash, for example iPads and many Android devices.



10. Press the **Update** button to return to the regular rich editor. Then scroll to the end of the form and press the **Save and display** button.

In Moodle 1.9, these are the alternative steps to upload the Flash and create the puzzle:

1. On the course main page, again choose the **Files** link from the **Administration** side block. Follow the same steps as we used previously to upload the `wordsearch.swf` Flash file to the course. Again, copy the link location for the Flash file to a text file.



2. Returning to the course main page, choose **Compose a web page** from the **Add a resource** drop-down menu. Enter a **Name** for the page, for example **Wordsearch puzzle**.
3. Under the section, **Compose a web page**, press the **Toggle HTML source** button on the bottom row of the rich-editor.
4. Go to point 6 of the instructions for Moodle 2, previously. Follow the instructions to create the `<object>` and `<param>` tags, adjusting the paths to the Flash and XML files appropriately.

The complete `<object>` element looks like the following. Note that the paths or URLs below take the form that you should expect for Moodle 2:

```
<object
  title="Word search puzzle"
  type="application/x-shockwave-flash"
  data=
    "/moodle2/pluginfile.php/87/mod_page/content/2/wordsearch.swf"
  height="450" width="600">
  <param name="movie" value=
    "/moodle2/pluginfile.php/87/mod_page/content/2/wordsearch.swf" />
  <param name="flashvars" value=
    "xmlFile=/moodle2/pluginfile.php/86/mod_resource/content/1
    /ws-shapes.xml" />
  [The Word Search puzzle requires the Shockwave Flash plugin.]
</object>
```

## What just happened?

We created a Page resource and uploaded the Flash file to it. Then we hand-crafted an `<object>` tag to display the word search puzzle by combining the XML with the Flash. The syntax we have used for the `<object>` and `<param>` elements are important for backwards compatibility with a range of browsers including Internet Explorer, Firefox, and Safari. In particular, note the `data` attribute and the apparent duplication of the URL to the Flash file in the line starting `<param name=movie...>`. The syntax will work into the future with **HTML5**.

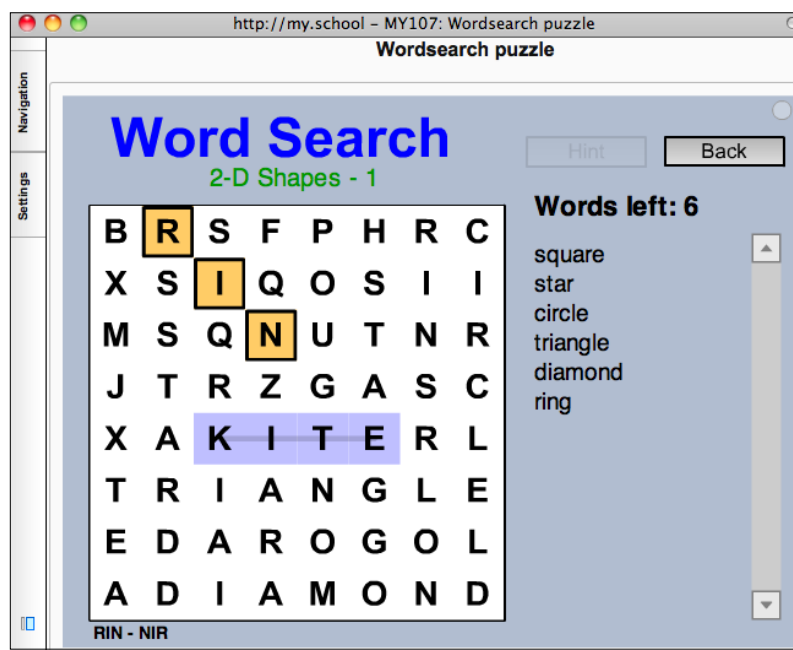


Also worthy of note is the `type` attribute with the value of `application/x-shockwave-flash`. This is a **MIME** type (Multi-purpose Internet Mail Extensions), which is an internationally agreed identifier. It helps e-mail software, web browsers, and other tools to correctly interpret and display documents, web pages, and resources like images. You may have come across other MIME types, including `image/png` (PNG images), `text/javascript` (JavaScript), and `audio/mp3` (MP3 encoded audio).

When you press the **Save and display** button in the final step, you are taken to a page with a large gray box. At the top will be the phrase **Word Search / Can you find the missing words?** Below you should see a button labeled **Start**. If you see the phrase **Failed to load ws-shapes.xml**, you will need to go back to check the URLs you used within the `<object>` element. For example, ensure that they all start with a slash `/` (or possibly `http://`).

Click the button labeled **Start** and you will see something like the next screenshot. On the left there is a white grid containing the letters that we entered in the XML file, starting with **BRSF...** in the top-left. To the right on a gray background is the list of words to find within the grid, starting with **square**. In the screenshot, you can see that I have already found **KITE** in the grid, so the count of **Words left** is **6**.

There is a **Hint** button to help if you get stuck. Press it and the first letter of one of the words remaining on the grid will flash. And we have a **Back** button to reset the puzzle to the start.



So, there we have our first puzzle. Now we'll take a step back and look at an alternative means of creating the word search XML.

## Online word search generator

In the previous sections, we discovered how to use a spreadsheet package such as Excel and a text editor to handcraft the XML file for the Flash word search puzzle. This is undoubtedly the route to take for the maximum control over the puzzle.

However, sometimes you may want to take a quicker route. Let's look at an automated, online tool.

### Time for action – using an online generator

**MathsNet** provides an online generator for the word search puzzle, based on software by **Duncan Keith** of `Subtangent.com`. It can be found at the following address:  
<http://mathsnet.net/puzzles/wordsearch/makewordsearch.html>.

These are the steps to use the online tool:

1. Visit the address in your browser. You will find a form such as the one shown in the next screenshot.
2. Enter a **Title**, for example **2-D Shapes**.
3. Type in a list of words separated by line-breaks where it says **Type your words here**. I entered **square**, **star**, **circle**, and **ring**.
4. Set the **Difficulty** of the puzzle to **Easier**. Note that you cannot choose down, right, and south-easterly as we did earlier when hand-crafting the puzzle.
5. Next, **Choose the output format**. Let's use one of the **Plain text** formats for testing, then **XML** to produce the final puzzle file.

6. Press the button to **Make it!** When you choose the **XML** output format, you will be prompted by your browser to save the file. Save it to your computer, and then continue from the previous section titled **Uploading the puzzle XML**.

The screenshot shows a web browser window with the title 'MathsNet: Wordsearch generator'. The page has a blue header with the MathsNet logo and navigation links: 'home geometry ASA2 curriculum puzzles articles books download about us try a short tour'. Below the header, the main heading is 'Wordsearch Generator'. There is a link 'Back to the Wordsearch' in red. The form includes a 'Title' field with the text '2-D Shapes'. Below it is a text area labeled 'Type your words here' containing the words 'square', 'star', 'circle', and 'ring'. A 'Clear' button is below the text area. The 'Difficulty' section has three radio buttons: 'Harder - up, down, left, right and diagonal words.', 'Easier - up, down, left and right.' (which is selected), and 'Easy - down and right only.'. The 'Choose the output format' section has five radio buttons: 'XML. Use this if you want to make a file for use with the wordsearch.swf movie.' (selected), 'HTML. Make a web page you can print out or edit in a word processor.', 'Plain text (no spaces between letters.)', 'Plain text (spaces between grid letters.)', and 'Plain text (tabs between grid letters.)'. A dropdown menu is open over the 'XML' option, showing a list of numbers from 1 to 9, with '6' selected. At the bottom, there is a 'Number of columns for word list:' label and a dropdown menu showing '6'.

### ***What just happened?***

We used an online word search generator to create the XML file. We noted that though the tool is fairly easy to use, we do not have much control over the difficulty of the resulting puzzle. For example, we can choose **Easier**, which gives us up, down, left, and right. Or we can use **Harder**, which additionally gives us diagonal words. However, when we hand-crafted the XML we were able to use down, left, and the diagonal from north-west to south-east.

Now that we have our first Flash puzzle under our belt, let's look at a jigsaw puzzle.

## A jigsaw puzzle

Previously in this chapter, we have worked through the creation of a Flash-based word search puzzle. We are going to continue with Flash interactive puzzles, and go completely visual. Let's create a jigsaw puzzle for our class.

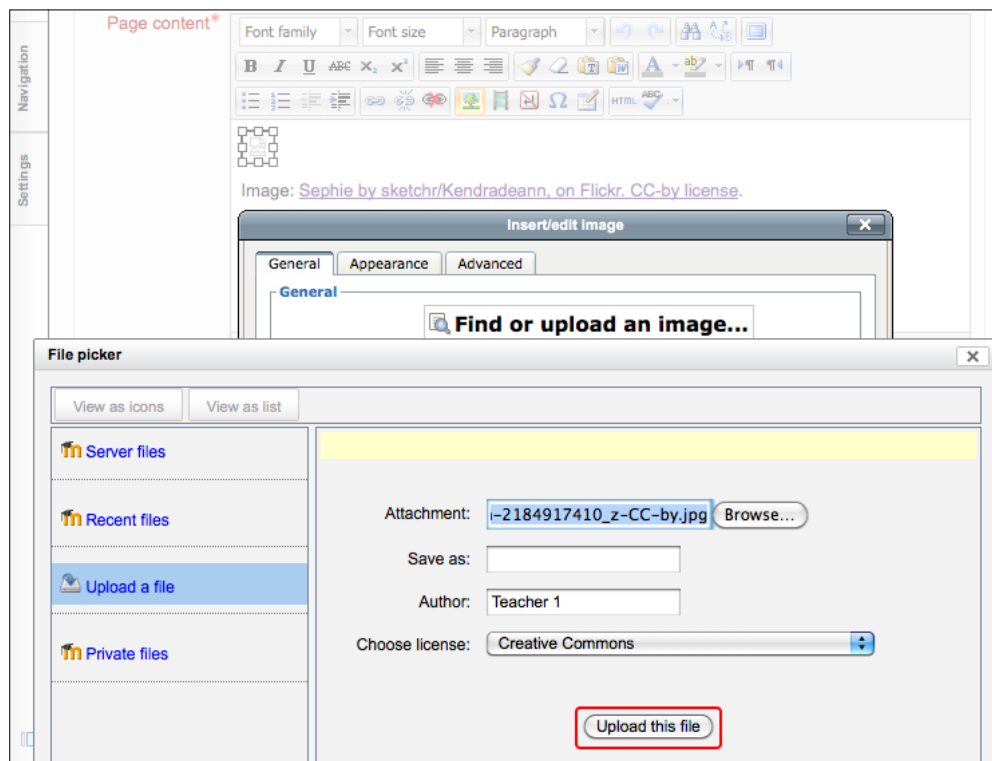
The first step is to source and upload a picture.

### Time for action – uploading a jigsaw picture

Here are some steps to follow to find an image for our jigsaw on Flickr, and upload it to our Moodle 2 system:

1. Visit the page for an image called Sephie Finished by **sketchr/ Kendradeann** on Flickr: <http://flickr.com/photos/kendradeann/2184917410/>. This attractive picture of a dog is available under a Creative Commons Attribution license.
2. Follow the link to **Actions | View all sizes**, and save a copy of the **Medium**-sized image (500 x 362 pixels) to your computer.
3. Log in to Moodle as a teacher and create a **Page** resource in topic **2** of your **MY107** course. Give the **Page** a **Title**, for example, **Jigsaw puzzle**, and a **Description**.

4. Scroll to **Page content** field. Click on the rich-editor button to **Insert/edit image**, and in the dialog that appears press **Find or upload an image...** Upload the image that you saved from **Flickr**, as shown in the next screenshot. You may also like to add an attribution link beneath the image, pointing back to the Flickr page.



5. Go to the foot of the page and press **Save and display**. Ensure that the image displays correctly.

### ***What just happened?***

We sourced an attractive example image from Flickr. You can if you wish find another image online, though I would recommend that you search for Creative Commons licensed content. Then we created a page resource in Moodle, and uploaded the image to it. And we checked that we could see the image as expected.

Now that we have the image, it's time to add the Flash.

## Creating the online jigsaw

We will use the **Duncan Keith's Jigsaw Flash**, again dual-licensed under Creative Commons Attribution-NonCommercial-ShareAlike and GNU GPL licenses. You can see a demonstration at <http://subtangent.com/maths/jigsaw-test.php>.

### Time for action – creating the jigsaw

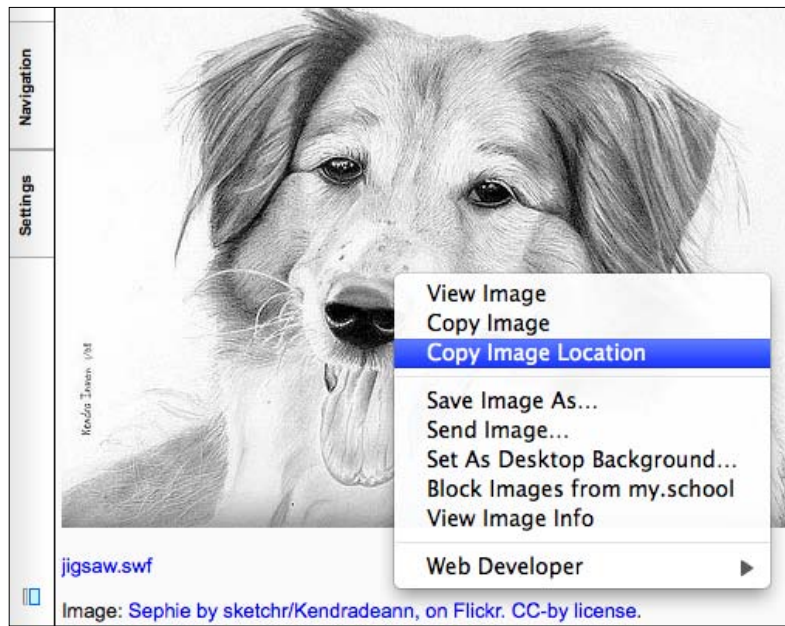
These are the steps that you should follow to prepare the jigsaw in Moodle 2:

1. Visit the address, <http://subtangent.com/maths/flash/jigsaw.swf> in your browser. In the browser choose **File** from the menu, then **Save page as....** Save the file as `jigsaw.swf` on your computer.
2. Log back in to Moodle as a teacher. Select the **Update** link next to the **Jigsaw puzzle** page in your **MY107** course. Scroll down to the **Page content** rich-editor.
3. Insert a new line after the image, and put your mouse cursor on it. Press the **Moodle Media** button (bottom row, next to the **Insert image** button). As previously, in the dialog, click the link labeled **Find or upload a sound, video or applet....**
4. A second dialog window will appear. Choose to **Upload a file**, and press the **Browse...** button to find the file `jigsaw.swf` on your computer. Press **Upload this file**, then press **Insert**. You will see a blue link beneath the image, labeled **jigsaw.swf**.
5. Go to the end of the form and press **Save and display**.

6. Right-click on the image in the destination page with your mouse. Press **Copy Image Location** in your browser's right-context menu. Paste the URL in a text file on your computer. Right-click on the **jigsaw.swf** link and press **Copy Link Location** in the right-context menu. Save this URL in the text file too. In each case you can remove the domain name, for example `http://my.school`. You may be left with something like the following:

```
/moodle2/pluginfile.php/89/mod_page/content/6/flickr-  
kendradeann-218-m.jpg
```

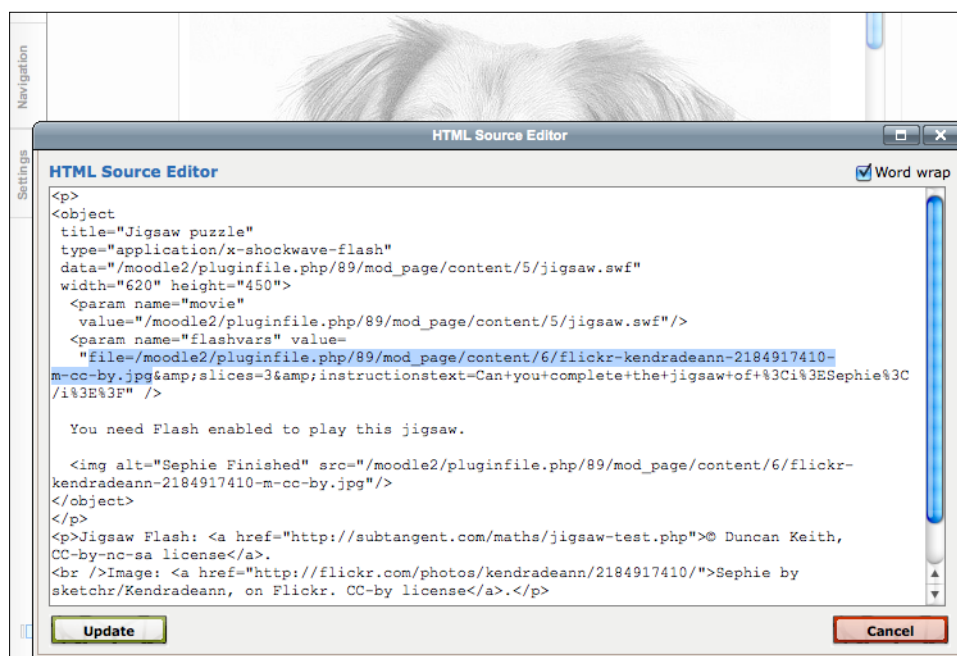
```
/moodle2/pluginfile.php/89/mod_page/content/5/jigsaw.swf
```



Now that we have the URLs or paths, we can create the puzzle. Follow this procedure in Moodle 1.9 and 2:

1. Press your browser's back button. In the **Page content** field, click the **Edit HTML source** button in the editor. This will open the **HTML Source Editor** window.
2. As we did previously, start by adding an opening `<object>` tag before the image markup, and a closing `</object>` tag after the image. Add `title`, `type`, `data`, `width`, and `height` attributes to `<object>`, as shown in the following screenshot:





3. Add a `<param>` element within the `<object>` element. Give it a name attribute equal to `movie` and a value attribute equal to the SWF path we noted earlier.
4. Add a second `<param>` element. The name attribute should be `flashvars`. The value attribute starts `file=/moodle2 ...jpg` – adjust this to the path for your JPEG image. What about the rest of the value attribute?
5. Visit this useful page from **Stephen Ostermiller** (GPL), <http://ostermiller.org/calc/encode.html>. Enter an instruction text, for example, Can you complete the jigsaw of `<i>Sephie</i>` the dog? Press the **Encode** button next to **URL**, and you should end up with this gibberish – `Can+you+complete+the+jigsaw+of+%3Ci%3ESephie%3C/i%3E+the+dog%3F`. Neat!



6. Add the URL encoded sentence from the previous step to the `flashvars` variable. That is, `value="file=/...jpg&slices=3&instructiontext=Can+you+complete+..."`. Note, in the middle you have `slices=3`, which specifies how many pieces the jigsaw will be divided into both horizontally and vertically.
7. Press the **Update** button to close the **HTML Source Editor** window. Scroll to the bottom of the form for our **Page** resource, and press **Save and display**.

The complete HTML source looks like the following:

```
<object
  title="Jigsaw puzzle"
  type="application/x-shockwave-flash"
  data="/moodle2/pluginfile.php/89/mod_page/content/5/jigsaw.swf"
  width="620" height="450">
  <param name="movie"
    value="/moodle2/pluginfile.php/89/mod_page/content
      /5/jigsaw.swf"/>
  <param name="flashvars" value=
    "file=/moodle2/pluginfile.php/89/mod_page/content/6/flickr-
      kendradeann-218-m.jpg&slices=3 &instructiontext=Can+you+
      complete+the+jigsaw+of+%3Ci%3ESephie%3C/i%3E+the+dog%3F" />

  You need Flash enabled to play this jigsaw.

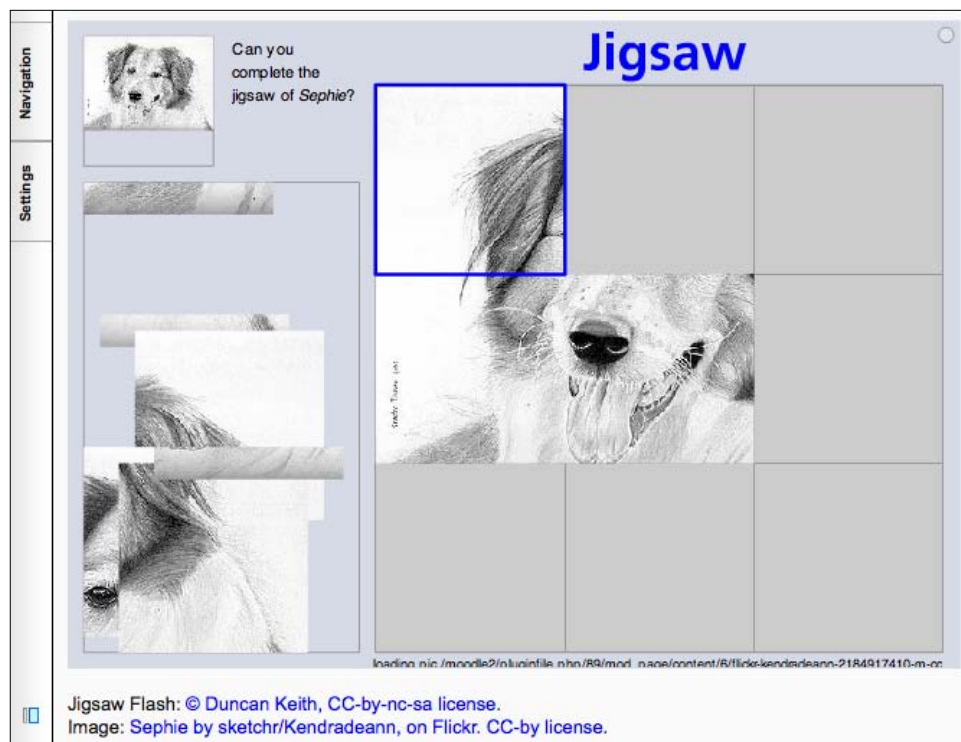
  <img alt="Sephie Finished" src=
    "/moodle2/pluginfile.php/89/mod_page/content/6/flickr-
      kendradeann-218-m.jpg"/>
</object>
```

## ***What just happened?***

We procured and uploaded the jigsaw Flash object. Having made a note of the URLs to the Flash and image files, we then went back and crafted the embed code. As noted the previous steps are the same for Moodle 1.9 and 2, though the form labels vary slightly.

We discovered how to add a parameter named `instructiontext` that is specific to the jigsaw Flash movie to the `flashvars` (Flash variables) variable. We found that we can use some simple HTML-like syntax for the instruction text, for example, `<i>` to mark up italics. Finally, we found that characters like space, `<` and `>` needed to be **URL encoded** in the `flashvars` variable.

In the following screenshot, we can see the jigsaw puzzle embedded in the Moodle activity:



We have now worked through two Flash-based interactive games. We can see some patterns in the HTML syntax, for example the `<object>` opening tag should always have `type`, `data`, `width`, and `height` attributes, with the correct values. There should be a `<param>` with a `name` attribute set to `movie`, and a `value` attribute set to the same URL as used in the object's `data` attribute. And there will usually be at one other `<param>` for the `flashvars` (Flash variables parameter).

We have used a free and open source collection of Flash plugins. You can of course use other sources of Flash plugins, but check the license.

#### Flash accessibility



It is worth noting that Flash-based activities are not always accessible to those with disabilities. A Flash video player, such as the one used on YouTube, can be made moderately accessible in Windows, for example to those with visual impairments who use screen reader software, and those who only use a keyboard. However, the Flash puzzles in this chapter require the use of a mouse and are visual in nature. If someone in your class requires an alternative you should look at the games and activities presented in *Chapter 6, Fun Games* and *Chapter 9, Embedding the Web*. See the *Appendix A, Accessibility for Online Teaching* for more information.

## Pop quiz

Why not try this quick quiz to check what you've learned? Beware! There may be more than one correct answer.

1. Which technology is the main focus of this chapter?
  - a. Java
  - b. Adobe/ Shockwave Flash
  - c. JavaScript
2. What is the format of the configuration file for the word search puzzle?
  - a. Microsoft Word document
  - b. Word search Text
  - c. XML
3. Which HTML tags are used to embed the puzzles?
  - a. `<param>`
  - b. `<a>`
  - c. `<object>`
  - d. `<iframe>`

## Have a go hero – jigsaw class exercise

One of the activities in this chapter was integrating a Flash-based jigsaw into your Moodle course. However, you don't always need a virtual learning environment.

Based on your experience, using the `<object>` tag, employ **Notepad2** or another text editor to create a simple HTML page which embeds a jigsaw. Explore how you could run a "create your own jigsaw" activity with your class.



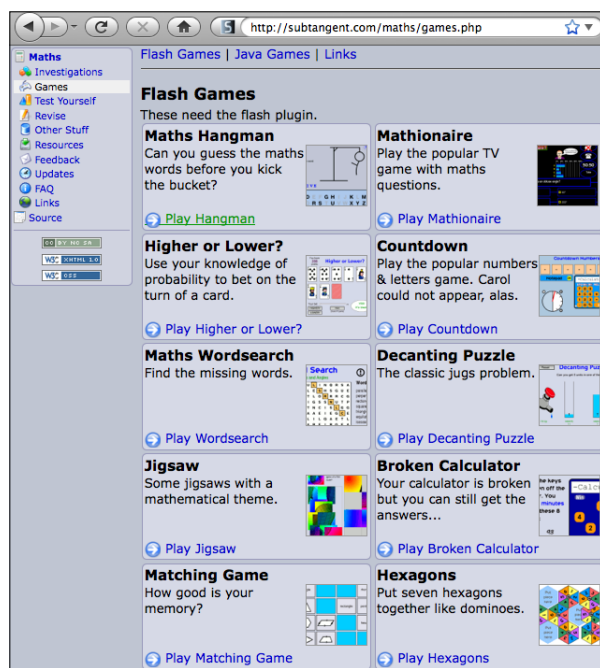
You can place your HTML file in the same directory as the `jigsaw.swf` and an image. The HTML page should start `<!DOCTYPE html><html>` and end with `</html>`.

Solution: use the same syntax for the `<object>` and `<param>` tags as previously, and adjust the paths to the SWF and image files as appropriate. You will be able to view the puzzle in your browser. Open Windows Explorer and your web browser, and use the mouse to drag the HTML file from Windows Explorer to the browser. The browser's address bar should contain the path, starting `file://`. As this task demonstrates, you and your class can create and view simple HTML pages on your own computer, extend your knowledge of ICT, and have fun!

```
<!DOCTYPE html><html><title>My Jigsaw</title>
<object
  type="application/x-shockwave-flash"
  data="jigsaw.swf"
  ... >
  <param ... />
</object>
</html>
```

## Have a go hero – more Flash games

Return to Duncan Keith's **Subtangent** site (<http://subtangent.com/maths/games.php>). Investigate the other Flash games, as shown in the following screenshot, and select one that is appropriate for your class. Work out how to create one of the games in Moodle. You can also look at other collections of Flash, HTML5, or Java games.



## **Summary**

In this chapter, we dealt thoroughly with Flash-based interactive puzzles. These will challenge the dexterity, visual, and verbal skills of your class.

Specifically, we covered:

- ◆ Creating a word search XML file using a desktop editor
- ◆ Uploading arbitrary files such as XML and Flash movies to Moodle
- ◆ Creating the embed code for the word search puzzle in Moodle
- ◆ Creating word search XML files using an online tool
- ◆ Creating the embed code for a Flash-based jigsaw puzzle

We also re-used images, and discussed Flash accessibility, MIME types, and URL encoding.

Now that we have looked at games, it is time to explore tools for story-telling. This is the topic of the next chapter.

# 8

## Stories Revisited

*Storytelling, reading, and narrative are important skills for your class to acquire. In Chapter 3, Telling Stories, we created a storytelling exercise centered on pictures in a Moodle Database activity. In this chapter, we are going to create an online book of a well-known story sourced from free content. This will give our class the opportunity to hone their reading skills, either individually or as part of face-to-face exercises.*

*This chapter uses the third-party Book module, written by **Petr Škoda**, who is a core Moodle developer. There are versions of the module for Moodle 1.8, 1.9, and 2 at the time of writing. Note that the Moodle 2 version requires at least version 2.0.2 of core Moodle. There are example Books at the American University of Beirut, <http://moodle.aub.edu.lb/mod/book/?id=1>.*

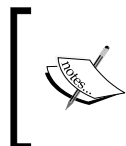
*The chapter also uses the third-party pop up Dictionary plug-in, written by **Patrick Thibaudeau**. This is currently available for Moodle 1.8 and 1.9.*

In this chapter we will:

- ◆ Find an open-content story suitable for our class, taken from Project Gutenberg
- ◆ Install the contributed Book module
- ◆ Create a Book resource and add the parts of the online book
- ◆ Add formatting and illustrations to our online book
- ◆ Install a third-party Dictionary filter and block plugin for Moodle
- ◆ Integrate an external dictionary site using the plugin
- ◆ Explore audio book options

Student learning outcomes:

- ◆ Learning together, collaboration
- ◆ Using a dictionary
- ◆ Literacy, reading, speaking, pronunciation, and narrative



In the remaining chapters we are going to concentrate on creating activities in Moodle 2. It is generally possible to create the same activities in Moodle 1.9 and you can refer back to earlier chapters for examples of how to accomplish this.

So let's get on with it...

## Finding a book

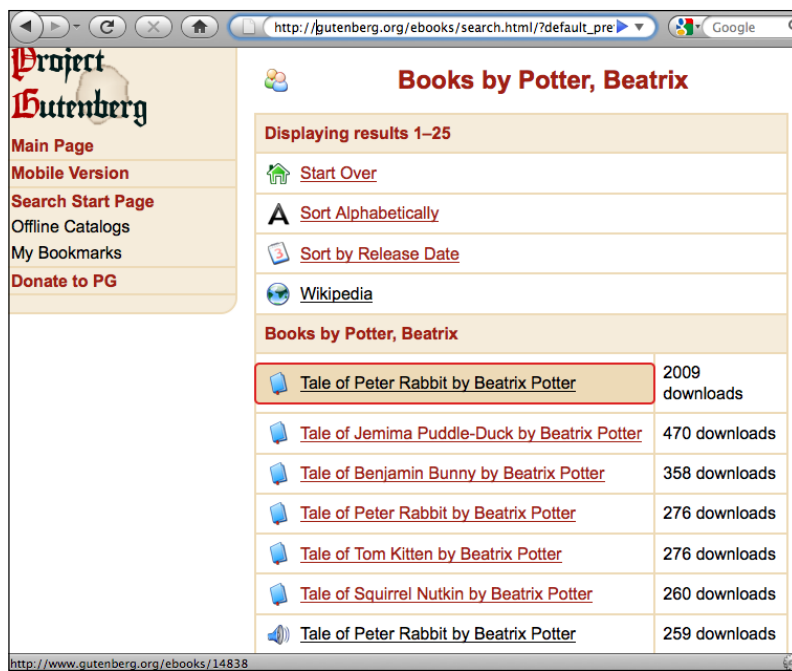
**Project Gutenberg** was set up in 1971 by William S. Hart to make books that are outside copyright in the United States available in digital format. Initially, books were available on floppy disks by mail, and it has since evolved into a voluntary collaborative effort to scan, format, and digitize ex-copyright, public domain books including classics from authors such as Conan Doyle, Charlotte Bronte, Jack London, and Mark Twain, and recent copyright books submitted by their authors. As of June 2011, 36,000 e-books are available in a range of formats (source: Wikipedia), under a liberal open content license and through the main website, <http://gutenberg.org/>. There are many classic children's books on Project Gutenberg, complete with pictures, so this will be the source of our book.

### Time for action – finding a book

I've chosen **The Tale of Peter Rabbit by Beatrix Potter** for this example. These are the steps to follow to acquire the digital book with images:

1. Visit the Project Gutenberg site at <http://gutenberg.org>, and enter **potter** in the **Author** search field.
2. On the search results page, click on the link labeled **Potter, Beatrix**. She currently appears at the top with the most downloads.
3. On the next page you will see a list of titles of books by our author, as shown in the following screenshot. The **Tale of Peter Rabbit** appears first with the most downloads at the time of writing. Click on the link for this book now, and you will be taken to the page for the EBook with the ID of **14838**: <http://gutenberg.org/ebooks/14838>.





4. Under the heading **Download This eBook** you will see a list of formats including **HTML** and **EPUB (with images)** – we know we're in the right place! It is most efficient to download a compressed ZIP archive containing our e-book complete with images. This can be found under **More Files**. The link is labeled **14838-h.zip** and the download size is 1.2 MB (megabytes).
5. Download the ZIP file now and save it to your computer. The page under **More Files** is very basic, so you will need to press your back button to get back to the rest of the site.
6. Unzip the ZIP archive somewhere convenient on your computer. We'll be using it shortly.

### ***What just happened?***

We chose **The Tale of Peter Rabbit** as our example book. It is suitable for a range of novice readers and it is quite short at 26 pages. There are Potter's attractive illustrations which look great online, and there is an audio version, which we will explore later.

Another interesting children's classic would be **Alice in Wonderland** by Lewis Carroll (EBook #19033 with images, not #11), which has 48 pages. Or if you and your class want a challenge there is **Treasure Island** by Robert Louis Stevenson (EBook #27780 with images, not #120), at about 260 pages. Again, these options have illustrations and there is an audio version.

There are also books in other languages, though there is not as much choice.

We have our e-book, so now let's install a module to display it.

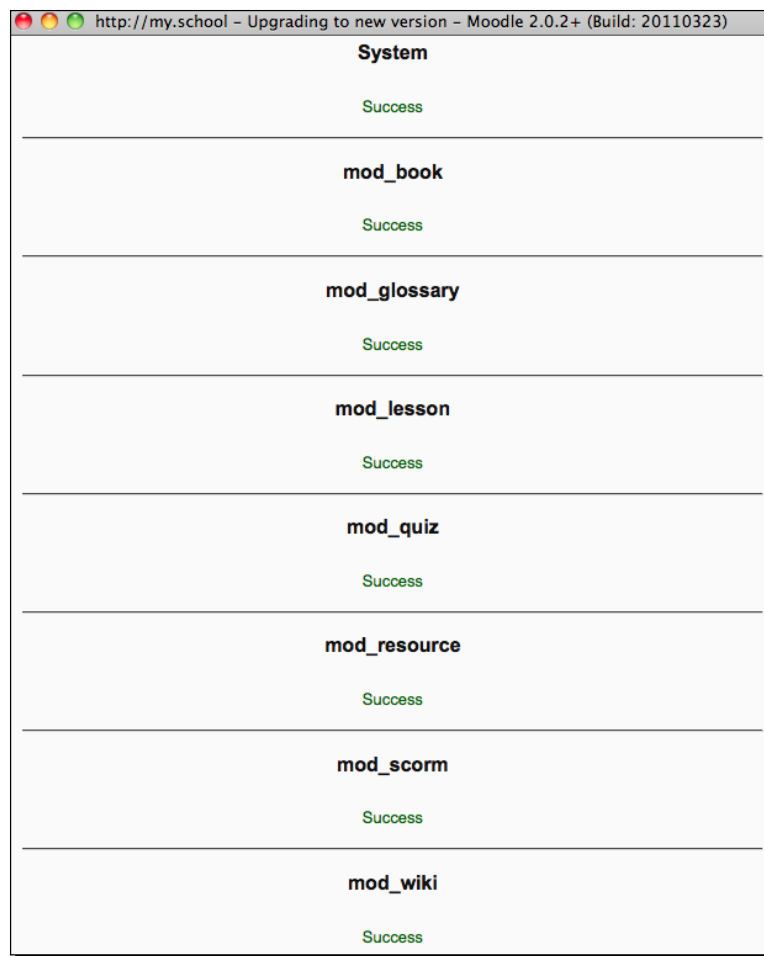
## Installing the Book module

In order to create the reading exercise for our class we need the third-party Book module, written by **Petr Škoda**. This will need to be installed by your system administrator or support staff, as they have the requisite permissions. Please hand over to them at this point.

### Time for action – installing the activity module

Here are the steps to install the third-party Book module, in Moodle 2:

1. Back up your Moodle database.
2. Download the appropriate version of the code from the Modules and plugins database on the Moodle website. The link is <http://moodle.org/mod/data/view.php?rid=319>.
3. Copy the module code to a directory called `book` on the server. In Red Hat Linux the destination might be `/var/www/moodle/mod/book`.
4. Log in to Moodle as an administrator.
5. In the **Settings** side block, expand the **Site administration** section. Follow the **Notifications** link at the top.
6. You will be taken to the **Plugins check** page. In the left-hand **Activity module** column the book module will be listed as **Non-standard (about to be installed)**.
7. Scroll to the bottom of the page, and press the button labeled **Upgrade** (below the **Reload** link).

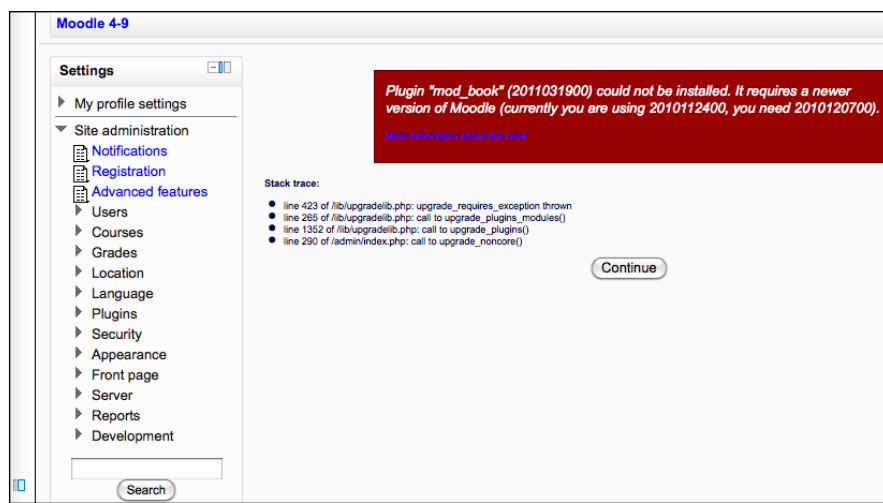


8. Press the **Continue** button to complete the installation of the plugin in the database. You will see output like that in the previous screenshot.

## What just happened?

We downloaded and installed the third-party Book module for Moodle. As noted earlier, if you are using Moodle 2, you will need at least version 2.0.2 of core Moodle. Otherwise, you will see a red error message similar to the one shown as follows:

Plugin "mod\_book" (2011031900) could not be installed. It requires a newer version of Moodle (currently you are using 2010112400, you need 2010120700).



If you come across this error, the solution is to download a newer version of core Moodle 2 – Moodle 2.0.2. Ensure that you have backed up your database and the source code, including third-party plug-ins. After you have upgraded, follow the steps to install the Book module again.

Look at your database after the installation and you will see two new tables named `book` and `book_chapters`. You may now hand back to the teacher with our thanks.

We have installed the Book activity module, so let's carry on and use it.

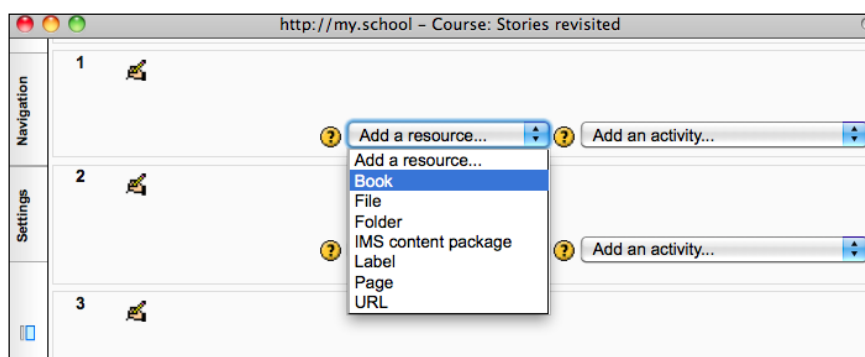
## Creating a book

We are going to create a course and add a Book resource. This can form a key part of your class' online resources to practice reading.

## Time for action – creating a book

We are going to follow these familiar steps to create our first book in Moodle 2:

1. Log into your Moodle site as a teacher and create a course. I gave mine a **Course full name** of **Stories revisited**, and a **Course short name** of **MY108**. You should ensure that the **Maximum upload size** is set to a reasonable number – **5MB** (megabytes).
2. When you are taken to the course main page, press **Turn editing on** and then scroll down to topic **1**. Choose **Book** from the **Add a resource** drop-down menu:



3. You will be taken to a form with the heading **Adding a new Book to Topic 1**. Enter **The Tale of Peter Rabbit** for the **Name** of the book. In the rich-editor labeled **Summary**, I typed **By** followed by a newline, then **Beatrix Potter**. You can complete the summary with instructions to your class and links back to the e-book page on Project Gutenberg.
4. Center-align all the text in the **Summary**, and format the author's name using **Heading 3** from the **Paragraph-Format** drop-down menu.

5. Choose **None** from the drop-down menu labeled **Chapter numbering**. We will roll our own numbering.

**Adding a new Book to Topic 1 ?**

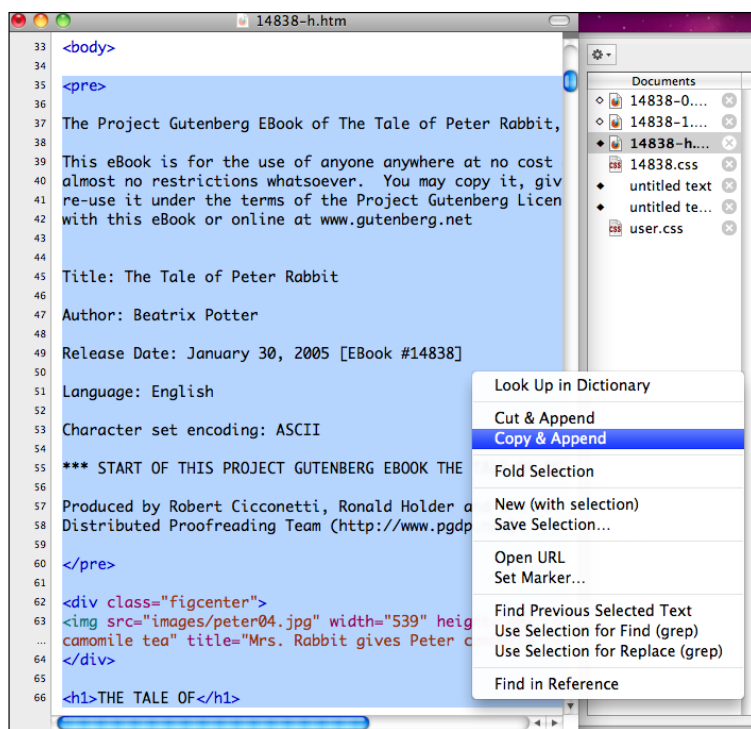
**General**

**Name\*** The Tale of Peter Rabbit

**Summary\*** By  
**Beatrix Potter**  
First published by Frederick Warne & Co., 1902  
[Book #14838 on Project Gutenberg | License](#)

Chapter numbering ?  
Custom titles ?

6. At the foot of the page press the button to **Save and display**.
7. The system takes you to another form where you can add the first chapter. Against **Chapter title** enter **Front matter**.
8. Find the unzipped e-book archive you downloaded from Project Gutenberg. Open the file **14838-h.htm** in a text editor such as **Notepad2** (on Windows) or **TextWrangler** (on Mac OS X).
9. As shown in the next screenshot, select all the text from the `<pre>` (pre-formatted) element after the opening `<body>` element to the `<hr ... />` (horizontal rule) element after the text that starts **Printed and bound in Great Britain**. Copy this to your clipboard.



**10.** Return to the **Add chapter form** in Moodle. Press the **Edit HTML Source** button in the rich-editor next to **Content**. Paste the first part of the book into the **HTML Source Editor** dialog. Scroll to the foot of the dialog and press **Update**.

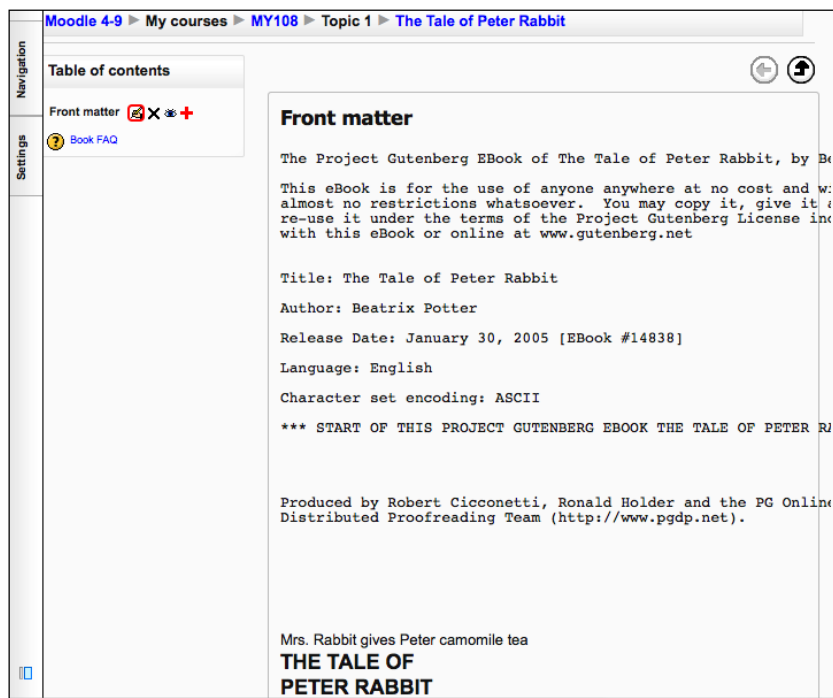
**11.** Press the **Save changes** button at the bottom of the form.

### ***What just happened?***

We've covered a lot of ground in this section.

As usual we created a fresh course for the chapter. We added a Book resource to it with a title and summary. Then we copied a part of the text from our downloaded e-book into the **Add chapter form**.

When you view the result initially, you will see something like the following screenshot. If you compare this with the **HTML** view of the book on the **Project Gutenberg** website, you will notice that there are some illustrations missing, and the formatting of the headings are not as expected. They should be centered.



Overall, we can identify some easy gains and challenges:

1. It is relatively simple to add text content to our Book resource.
2. It is a bit fiddly to divide an e-book from Project Gutenberg into chapters or other sensible components, prior to adding it to the Book resource.
3. We need to add images to our e-book.
4. We'd like to improve the formatting or styling of the e-book.

We will deal with these points next. We are well on the way to creating an online book to entertain our class and to help with reading practice.



## Improving our book

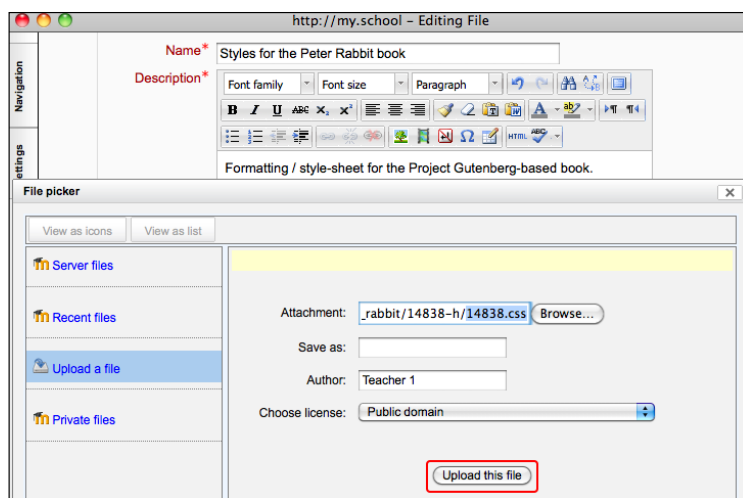
As we discovered in the previous section, it is fairly straightforward to add content to our online book. However, we need to invest more time in the formatting and images for the book. And it would be helpful if there were a pre-prepared version of the Project Gutenberg book in HTML format. We'll tackle these next.

### Time for action – adding custom styles

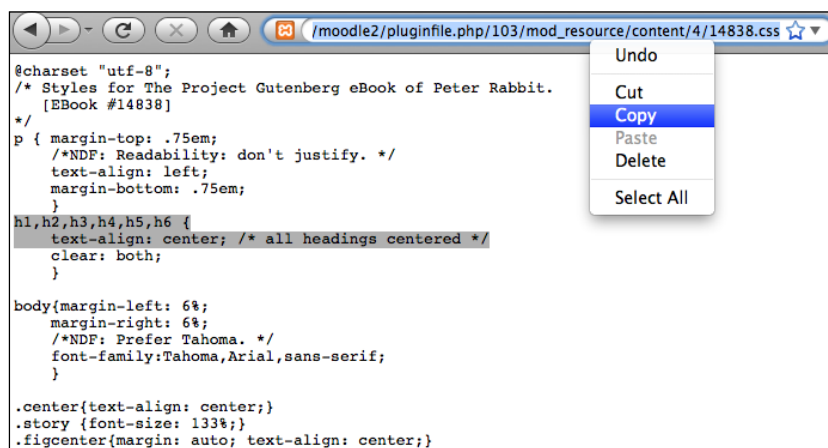
These are the steps that we will follow to download the alternative, pre-prepared book, and add a stylesheet for our book to Moodle 2:

1. Visit <http://freear.org.uk/moodle>, follow the link for **Downloads**, and download the ZIP archive of the **Project Gutenberg e-book of Peter Rabbit**. Save it to your computer and unzip it to a convenient folder.
2. You will see a directory named `14838-h` containing a number of files, for example:
  - A HTML file for the book's front matter, `14838-0.html`
  - A style-sheet named `14838.css`, which controls the formatting of our online book
  - A sub-directory named `images`, which contains a number of GIF and JPEG images
3. Return to the course main page in Moodle and ensure that editing is turned on. In topic 1, choose **File** from the **Add a resource** drop-down menu.
4. You will be taken to a page with the heading **Adding a new File to Topic 1**. Enter a **Name** such as **Styles for the Peter Rabbit book** for the resource. Enter a **Description**.
5. Under the section of the form headed **Content**, press the **Add** button.

6. A File picker dialog will appear. Choose the **Upload a file** link if it is not already chosen. Press the **Browse...** button and find the 14838.css stylesheet on your computer. Returning to the dialog, press the **Upload this file** button.

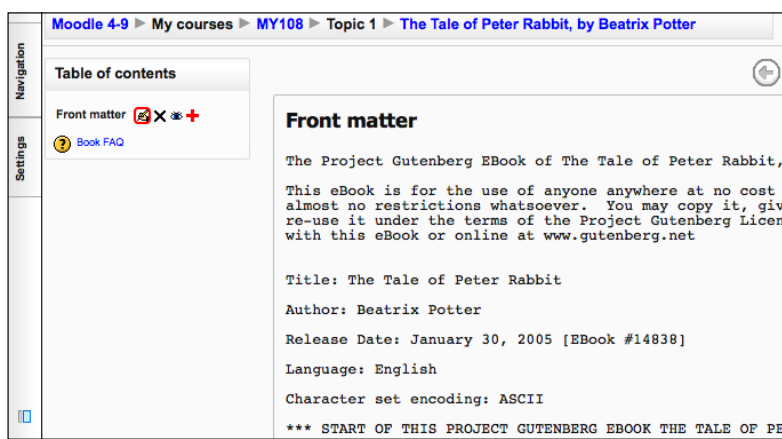


7. Leave the defaults for the **Options** and **Common module settings** sections. Press the button to **Save and display**. On the resulting page, click the link where it says **Click 14838.css link**. You will see something like the next screenshot:



8. Select the URL of the file in the address bar of your browser, and copy it to your clipboard. Paste the link in a text editor for use in the next few sections, for example, `http://my.school/moodle2/pluginfile.php/103/mod_resource/content/4/14838.css`.

9. Hit your browser's **Back** button and return to the course main page using the **MY108** link in the site breadcrumb trail. Click on **The Tale of Peter Rabbit** book link under topic 1.
10. Press the **Edit** icon-link next to the **Front matter** link in the **Table of contents** block, as shown in the next screenshot:

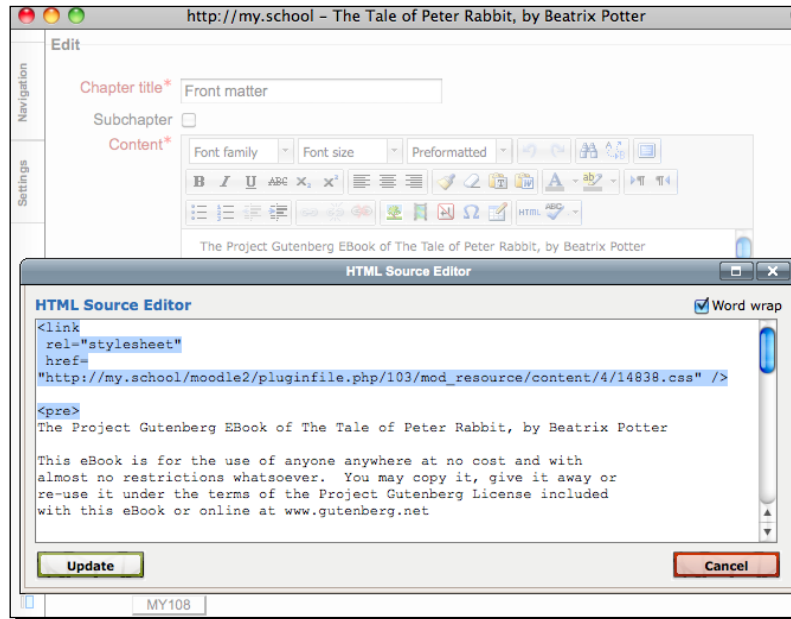


11. Press the **Edit HTML Source** button on the bottom row in the **Content** rich-editor. And, in the **HTML Source Editor** dialog, before the opening `<pre>` (pre-formatted) tag, enter the following code, substituting the URL for the stylesheet you copied previously:

```
<link
  rel="stylesheet"
  href=
"http://my.school/moodle2/pluginfile.php/103/mod_resource/
content/4/14838.css" />
```

```
<pre>
  The Project Gutenberg EBook of The Tale of Peter Rabbit, by
  Beatrix Potter
  ...
```

You can see the `<link>` element being added in the following screenshot:



12. Press the **Update** button on the dialog, then press **Save changes** below the form.

### ***What just happened?***

In this section, we downloaded a second, pre-prepared ZIP archive of our Project Gutenberg e-book. We found that the single large HTML file `14838-h.htm` had been divided into the more usable parts, `14838-0.html` for the front matter, `14838-1.html` for the first part of the book, and so on. Note that our story, **The Tale of Peter Rabbit**, does not have chapters. We also found that there was a stylesheet (CSS) file `14838.css`, to control formatting.

We uploaded the CSS file to our Moodle site as a **File** resource. Then we used the `<link>` HTML element to incorporate the stylesheet in our online book. When you view the book now, you should see that the title of the book is now centered on the page. This is achieved with the following stylesheet rule declaration:

```
h1, h2, h3, h4, h5, h6 {
    text-align: center;
}
```

We can see the declaration consists of two parts. There is a selector, which in this case is a comma-separated list of elements `h1` (heading level 1), `h2`, and so on. The second part, highlighted, declares that the selected elements should be center-aligned (note that we require the American spelling of center in the code).



Here, we have only dipped our toe in the water of **Cascading Style Sheets** (CSS). As a teacher, you do not have to do much with themes or CSS. Suffice to say that Moodle has a powerful theming system, and individual plugins and modules can apply their specific stylesheets. Generally, you should not try to use arbitrary CSS files directly in your Moodle resources!

In this section, we improved the formatting of our e-book. Let us continue by adding pictures to make the book enjoyable for our class.

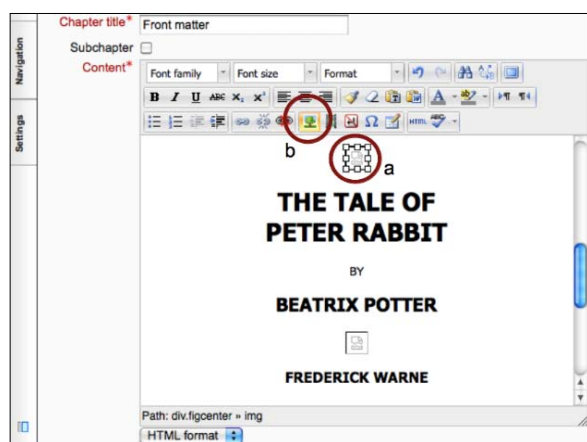
## Adding pictures

We are going to continue to improve our online Peter Rabbit book, by adding the illustrations that we downloaded from Project Gutenberg.

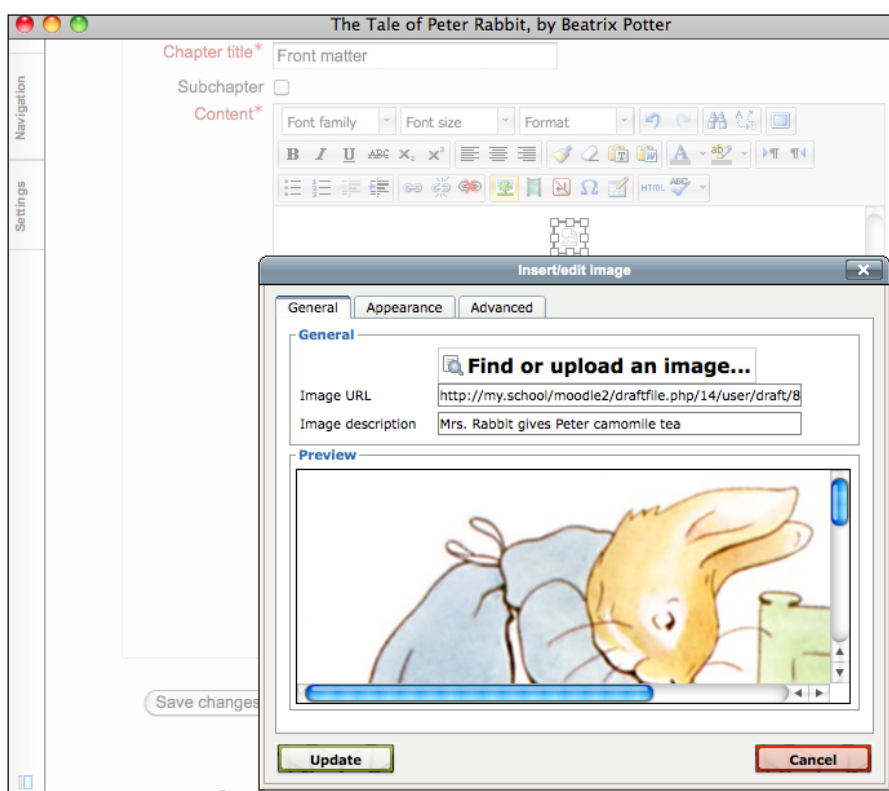
### Time for action – adding pictures to our book

These are the steps that we can take to add pictures to our online book, in Moodle 2:

1. Return to the book resource page in Moodle. Again click on the **Edit** link next to **Front matter** in the **Table of contents** block.
2. In the **Content** rich-editor, scroll down until you find the first broken image icon in the content. You can see there are two of them in the following screenshot, with the first selected by my mouse (marked **a**):



3. Press the **Insert/edit image** button in the rich-editor, marked **b** in the previous screenshot. In the **Insert/edit image** dialog, you will see an entry for the **Image URL** that looks something like: `http://my.school/moodle2/images/peter04.jpg` (clearly too short for Moodle!)
4. Click on the **Find or upload an image** link. Choose **Upload a file** on the left of the **File picker** dialog. Browse to the directory `14838-h/images/` on your computer and select, in this case, `peter04.jpg`. Back in the File picker press **Upload this image**.
5. Returning to the **Insert/edit image** dialog, press the **Update** button, as shown in the following screenshot:



6. Repeat for the other missing image in this section of the book. Then press the **Save changes** button beneath the editor.

## What just happened?

We found the broken or missing images in the first part of our book. Then we uploaded the illustration from our Project Gutenberg archive in the appropriate place. When you view the end result in your browser, you may find that the original illustrations from **Peter Rabbit** are too big and over-spill the space left for the Book resource in your site's theme.

If that is the case try this procedure:

1. Return to the book's chapter/section editor and select the image.
2. In the **Insert/edit image** dialog, choose the **Appearance** tab. Ensure that the **Constrain proportions** checkbox is ticked.
3. Edit the first of the **Dimensions**, which is the width. For the `peter04.jpg` image, try reducing the width from 539 to 450 pixels – the height will be re-calculated automatically to maintain the aspect ratio.

Our online book is well on the way now, and the original illustrations will add interest and enjoyment for our class.

Let's press on and finish off the book.

## Completing our online book

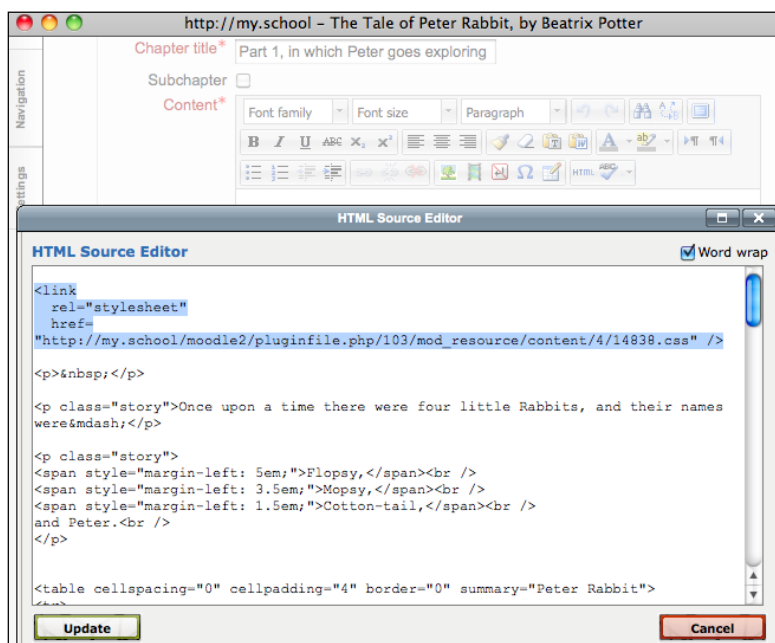
In the previous sections, we have lavished attention on the first part of the Peter Rabbit book. Now is the time to add more parts.

### Time for action – adding the remaining parts to the book

You will want to follow these steps to add the remaining parts of the book to Moodle 2:

1. Return to the **Peter Rabbit** book in our course. With editing switched on, you'll see a distinctive red-cross icon next to the **Front matter** item in the **Table of contents** block. Click it to **Add a new chapter**.
2. In the **Editing chapter** form, add a **Chapter title**. I put **Part 1, in which Peter goes exploring**. Press the **Edit HTML Source** button in the rich-editor labeled **Content**.
3. Open the file `14838-1.html` in your text editor (for example, Notepad2) from the pre-prepared archive you downloaded. Select the HTML markup in the editor from the `<link>` element to the final `</table>` tag inclusive. Copy to your clipboard.
4. Paste the markup into the HTML Source Editor dialog in Moodle.

5. Adjust the href attribute in the stylesheet <link> element to match what you saved in your notepad in the **Improving our book** section. The URL should be something like, `http://my.school/moodle2/pluginfile.php/103/mod_resource/content/4/14838.css` and the result should look like this:



6. Press the **Update** button in the dialog. Then scroll to the bottom and press **Save changes** for this part of the book.
7. Go back and edit the part to add the missing images. Follow the instructions in the *Adding pictures* section.
8. Repeat steps 1 to 7 in this procedure for the remaining four parts of the Peter Rabbit story (14838-2.html, 3, and so on), linking to the stylesheet and uploading the illustrations as you go.

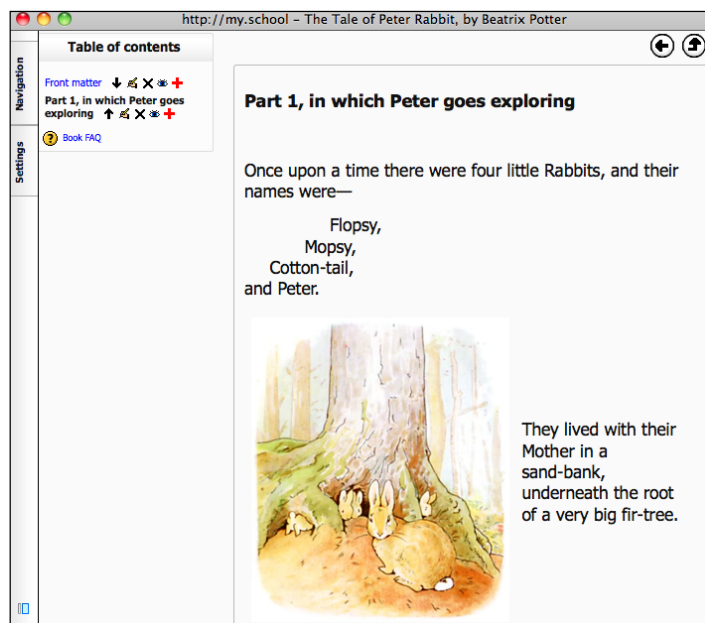
## What just happened?

We added the remaining parts to our book. These are termed chapters in a **Book** resource. The parts were sourced from the pre-prepared archive we downloaded previously, with filenames such as 14838-1.html. We linked to the stylesheet that we uploaded earlier. And we uploaded and embedded the illustrations from the downloaded archive.



The dimensions of many of the illustrations in the Peter Rabbit story will need to be reduced. This is because in the Project Gutenberg e-book they are positioned next to the text using a HTML <table> element.

You can see the results in the next illustration, which contains the famous introduction: **Once upon a time there were four little Rabbits, and their names were—Flopsy, Mopsy, Cotton-tail, and Peter.**



Congratulations! You have completed an online book, which I hope will captivate your class.

But wait, we can help our class in their reading practice by incorporating an online dictionary.

## Have a go hero

Here is an activity to help extend your understanding of the preceding topics.

Up to this point in the chapter, we have constructed a text and picture-based online book. Project Gutenberg and Librivox (<http://librivox.org>) also collected audio renditions of public domain books, including The Tale of Peter Rabbit and other children's stories.

Find and download the audio files for your chosen story. Explore how to upload and play them from within the Book resource.



You will probably want to download the MP3 audio, though you may also download OGG files, for browsers including Mozilla Firefox.

>> Solution: if the MP3 audio multimedia filter is enabled in Moodle, then any uploaded MP3 file is automatically replaced by an audio player.

## Installing a dictionary

We have created an e-book in our course. In order for our class to benefit from the book it would be great if we could link to a dictionary. **Patrick Thibaudeau** has contributed the flexible **Popup Dictionary** filter and block plug-in (<http://moodle.org/mod/data/view.php?rid=1751>).

At the time of writing the plugin is only available for Moodle up to version 1.9.

You will need to hand over to your Moodle support person who has the permissions to install the plugin.

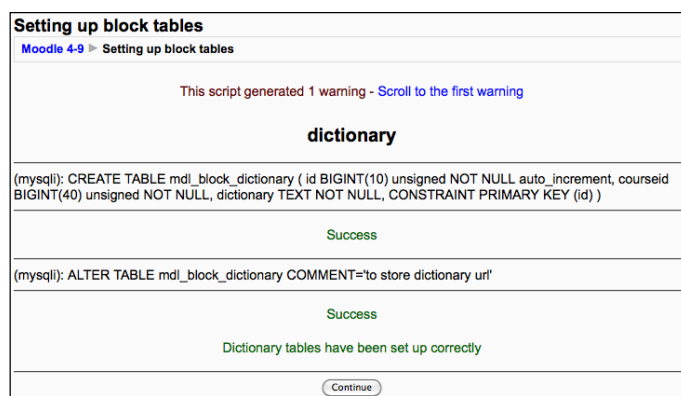
### Time for action – installing a pop-up dictionary

Follow these instructions to install the plug-in in Moodle 1.9:

1. Back up your Moodle database.
2. Visit the page on Github for the Popup Dictionary plugin:  
[https://github.com/nfreear/moodle-plugin\\_dictionary](https://github.com/nfreear/moodle-plugin_dictionary).
3. Check the installation instructions and links in the **README** file, press the **Downloads** link on the right, and download the appropriate ZIP archive (currently, [https://github.com/nfreear/moodle-plugin\\_dictionary/zipball/master](https://github.com/nfreear/moodle-plugin_dictionary/zipball/master)).
4. Uncompress the ZIP archive to your computer. Copy the `blocks/dictionary` directory to the Moodle blocks directory on your server. On Red Hat Linux for example, the destination may be `/var/www/moodle/blocks/dictionary`.
5. Copy the `filter/dictionary` directory to the filter directory on your server, for example, to `/var/www/moodle/filter/dictionary`.
6. Unless the README says otherwise, add the following line to the `header.html` file in your Moodle 1.9 theme, modifying the `src` attribute as appropriate for your installation:

```
<script type="text/javascript" src=
  "http://my.moodle/filter/dictionary/dictionary.php"></script>
```

7. Log in to Moodle as an administrator. Follow the **Notifications** link from the **Site Administration** side-block. Press **Continue**, and you will see output like the following:



Don't worry about the phrase **This script generated 1 warning** in the previous screenshot. So long as you see a phrase like **Dictionary tables have been set up correctly**, then everything has gone smoothly.

You will be taken to a settings page with pairs of fields labeled, for example **filter\_dictionary\_name** and **filter\_dictionary\_uri**. The first few pairs have been pre-filled. If they seem appropriate then keep them and add the following beneath:

- ◆ Google translate (en|en)  
http://translate.google.com/#en|en|
- ◆ Freedictionary.org  
http://freedictionary.org/?Query=
- ◆ Yahoo! Kids  
http://kids.yahoo.com/reference/dictionary/english/  
search?query=
- ◆ Kids Open Dictionary  
http://dictionary.k12opened.com/?p=modify&tab=view&w=

Scroll to the foot of the settings page and press **Save Changes**.

## ***What just happened?***

We downloaded and installed the **Popup Dictionary** plugin. Then we configured a number of additional online dictionary and pronunciation services, to give our teachers plenty of options. Note how most of the URLs end with an = equals sign, as the query will be appended to the end. Also note the capital Q for the parameter Query in the URL for h.

That completes the installation and configuration, so please hand back to the teacher. Now we can make use of the dictionary in the example course.

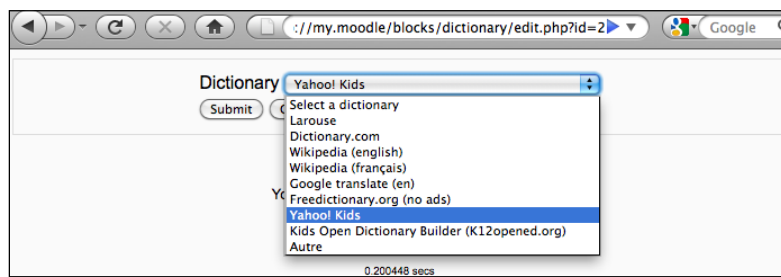
## **Integrating a dictionary service**

We will now integrate a dictionary in The Tale of Peter Rabbit. This will enable our class to explore the meanings and pronunciation of words.

### **Time for action – using the dictionary**

These are the steps to follow in order to incorporate a dictionary in your Moodle 1.9 course:

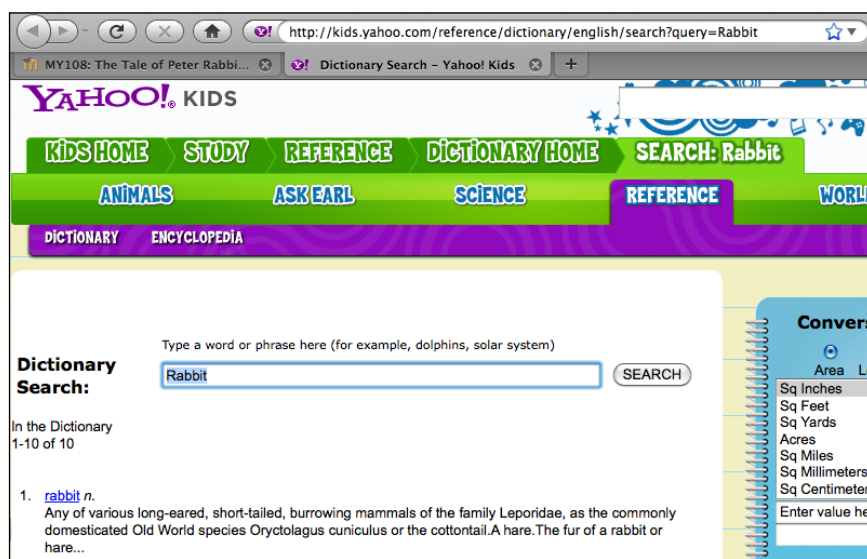
- 1.** Log in to your Moodle site as a teacher. Choose the **MY108** course.
- 2.** Press the **Turn editing on** button and scroll to the bottom-right of the course main page. Choose **Dictionary** from the **Add** drop-down menu in the **Blocks** side-block.
- 3.** A **Dictionary** block will appear. Click on the link to **Add a dictionary**.
- 4.** Choose **Yahoo! Kids** from the drop-down menu on the **Add a dictionary** page. Press **Submit** to save your selection and return to the course main page.



5. Now follow the link to the Peter Rabbit online book. Choose **Part 1**, and you should find that an icon (🔍) has appeared in the top-left of the main content area. Hover the mouse over a word in the story, for example **Rabbits**. You will see a tooltip, which says **New window: double click on a word to view it in the dictionary**. Double-click and see what happens.

### ***What just happened?***

We added our first block to a course, namely the Dictionary block. Then we configured the dictionary block. And, we went to our online book, to try out the dictionary functionality. By double-clicking on a word in the story, we opened the external online dictionary in a new window, as shown in the following screenshot:



You will want to try out the various dictionary services, as they all have pros and cons. **Google Translate** is not a dictionary, but offers an audio pronunciation feature; **Freedictionary.org** offers images to illustrate the definitions, while **Yahoo! Kids** has a more appealing design and contains several interesting resources for children.

So there we have it. An online book to help your pupils hone their reading skills, and on dictionary to help your class explore the rich world of words. And to cap it all, we've just introduced our first Moodle side-block.

## Have a go hero

Here is an activity for you to extend your understanding of the topics in this chapter.

In the last section, we integrated an online dictionary with our e-book. One of the available dictionaries was the Kids Open Dictionary Builder (<http://dictionary.k12opened.com>). This has a number of interesting properties:

1. The dictionary is being built collaboratively by volunteers, and is dedicated to the public domain.
2. Dictionaries can be exported in various formats including Moodle Glossary import XML.

Your mission is to add one or more entries to the Kids Open Dictionary, and to export a dictionary relevant to an e-book (Peter Rabbit or another that you've chosen) and import it into a Moodle Glossary.

>> Solution: the Glossary activity should be in the same course as the book. And the exported glossary will already have the auto-linking feature enabled for each word.

## Pop quiz

Here are some quick questions to check your understanding of the chapter. Beware, as there may be more than one correct response.

1. What is/are the criteria for the inclusion of a book in the Project Gutenberg collection?
  - a. The rights to the book are bought by Project Gutenberg.
  - b. The copyright on the book has expired in the USA.
  - c. Volunteers are prepared to scan in and format the book.
  - d. The book is outside copyright in Western Europe.
  - e. The book is submitted by the author.
2. What are the particular features of the Book plug-in?
  - a. A table of contents is displayed.
  - b. It incorporates a Moodle filter.
  - c. An index is generated.

## Summary

We covered a lot of new ground in this chapter.

Specifically, we:

- ◆ Explored the Project Gutenberg web site and found a suitable book, namely The Tale of Peter Rabbit, by Beatrix Potter
- ◆ Created a Book resource and added formatting and illustrations
- ◆ Learned how to integrate an online dictionary into our book

We also introduced CSS stylesheets, audio books, and Moodle side-blocks.

Now that we've learned about a new resource type and creating online books, it's time to tackle embedding external, visual activities, games, and simulations. This is the topic of the next chapter.





# 9

## Embedding the Web

*In the previous chapters we have explored embedding content from the Web into learning activities for our class. We have downloaded open content from various sources, including images from Flickr and elsewhere, and Flash animations from [Subtangent.com](http://Subtangent.com).*

*Often as a teacher you will wish to go down the path of sourcing content and uploading to Moodle, so that your course material is not dependent on third-party sites, or for your children's safety for example.*

*However, sometimes it is better to break out of your controlled, **walled-garden**, to embed a little bit of the Web directly into your courses. This can be quicker, and you can use it to encourage sharing and collaboration. In this chapter we will explore some quick hacks and tricks to achieve this.*

During this chapter there will be one activity that requires the use of the latest Web browsers, for example, Firefox version 4 at the time of writing. Other activities will require Java and Flash to be installed on your computer.

In this chapter we will:

- ◆ Introduce **Scratch**—a programming language and environment tailored to children
- ◆ Explore different ways of embedding Scratch projects in our course activities
- ◆ Find out how to incorporate news feeds into our course using a side-block
- ◆ Try our hand at **PhET** science simulations and link to one from within Moodle
- ◆ Embed a HTML5 jigsaw puzzle

So let's get on with it...

## What is Scratch?

**Scratch** is a programming language developed especially for teaching and learning. It was created by the Lifelong Kindergarten Group, at the **Massachusetts Institute of Technology (MIT)**, specifically for children from 6 to 16 years (it has been used by younger children and many adults). It features a visual environment where you can drag building blocks for a software program from a menu. Its purpose is not just to encourage interest in ICT and programming, but to nurture design, problem solving, and creative skills in a stimulating and accessible way.

You are free to visit the Scratch website, and download the integrated desktop development and viewing computer program (<http://scratch.mit.edu>). There are software downloads available for Windows, Mac OS X, and Ubuntu. You may also want to explore resources for educators (<http://info.scratch.mit.edu/Educators>).

There is an active and international Scratch community, centered on the pages at <http://scratch.mit.edu/galleries/>. It has always been the aim of Scratch's creators to foster the sharing and reuse of **Scratch projects**, as the software programs are termed.



**Michael Badger** has written the *Scratch 1.4: Beginner's Guide Book*, published by Packt. This book takes you step-by-step through the installation and use of the Scratch environment and language. It helps you learn to develop software using this fun, visual environment.

Find out more at <http://packtpub.com/scratch-1-4-beginners-guide/book>.

## Embedding Scratch projects

We are going to explore two different routes to embedding Scratch projects. You will be able to set your class an activity where they work in small groups, and find a Scratch project for another group to try. Each group will be able to embed their chosen project on your course website and comment on their classmates' submissions.

### Time for action – embedding Scratch applets

Let's follow these steps to start exploring how to embed scratch projects:

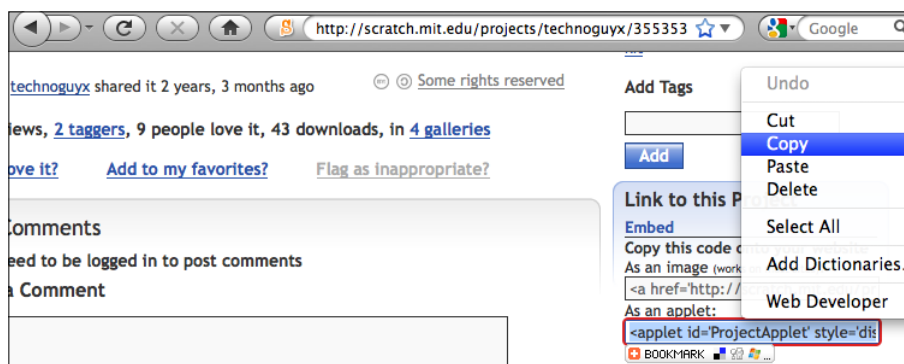
1. Visit the Scratch galleries at <http://scratch.mit.edu/galleries>. Take a look around, and consider registering on the site.

2. Find the **Drum Kit 2.0** project by **technoguyx** at, <http://scratch.mit.edu/projects/technoguyx/355353>. You should see something like the screenshot below. If you are having problems, visit <http://java.com/en/download/testjava.jsp> to check your Java installation.

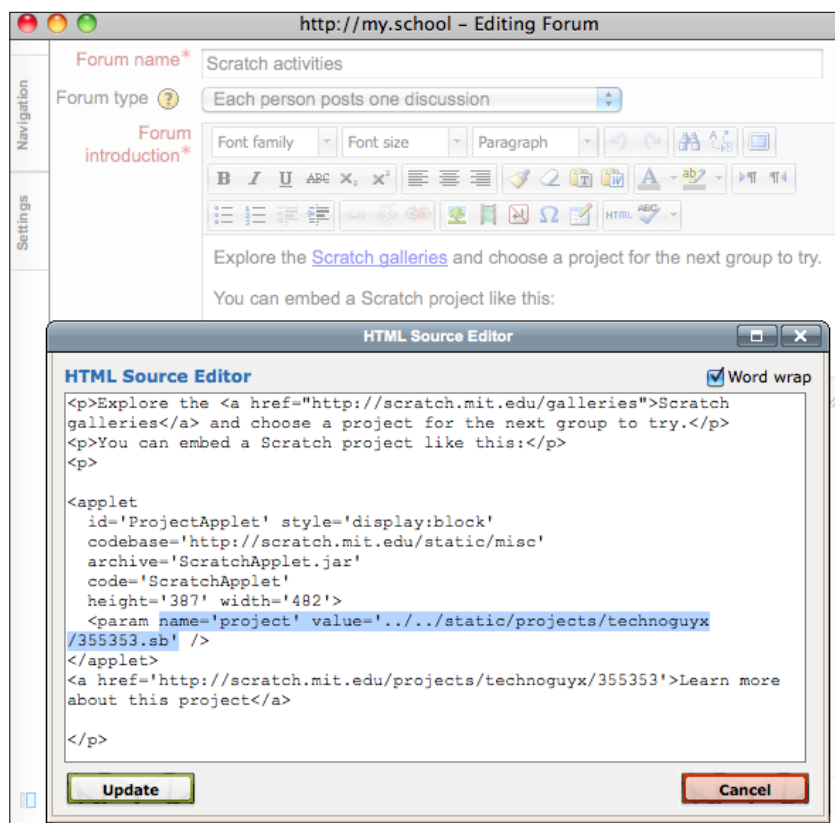


3. Select the drum kit with your mouse and try playing it using your keyboard. *Space* is the bass or kick drum, *D* is the Hi-hat cymbal, *V* is the snare drum, and so on. Fun, fun!

4. Scroll to the right of the page, and click on the **Embed** link beneath **Add tags** and **Link to this Project**. Select all the text starting `<applet id=...` in the second edit box. Copy to the clipboard as shown in the next screenshot, and paste into a new file in your text editor:



5. Log in to your Moodle site as a teacher, and create a new course. I gave mine a **Course full name** of **Embedding the Web** and a **Course short name** of **MY109**. I used the **Topics** course format.
6. Go to topic 1 in your new course. Choose **Forum** from the **Add an activity...** drop-down menu. In the activity form enter a **Forum name**—I put **Scratch activities**. I chose **Each person posts one discussion** as the **Forum type**, and I set **Forced subscription** as the **Subscription mode**. You may wish to change **Read tracking for this forum** to **On**.
7. In **Forum introduction**, press the **Edit HTML source** button. Paste the embed code that you copied previously from the Scratch site into the **Edit HTML Source** dialog, after your class instructions. You will end up with something like the next screenshot. Note that line breaks have been inserted for clarity only.



8. Press the **Update** button to close the **HTML Source Editor** dialog. Scroll to the end of the form and press the familiar **Save and display** button.

## What just happened?

In the previous section we found a project from the Scratch community, and embedded it using the given embed snippet in a forum.

We chose a multi-threaded forum (**Each person posts one discussion**). While straightforward for your class to grasp if they are new to forums, the discussion is spread over multiple pages. This is an advantage as some Scratch projects can make the computer work hard. It is generally best to limit you and your class to one embedded project per page on your Moodle site. You can get some idea of the weight or size of the project by looking in the top right of the project page on the Scratch site. Under the heading **Download this project!** you will see the phrase **Download the 10 scripts of "Drum Kit 2.0"....** You will find some projects with just one script, and some with more than 10.

Note that all projects uploaded by the community to the Scratch website are available under a **Creative Commons Attribution Share-Alike License** ([http://info.scratch.mit.edu/License\\_to\\_play](http://info.scratch.mit.edu/License_to_play)). So you and your class are free to share them with each other and with friends and relations. When they are embedded, the Scratch projects run in a Java applet within your browser. You can also download the .sb (Scratch binary) project files to your computer and run them offline in the Scratch development environment.



The interactive games in some of the previous activities were not accessible to all groups, notably those with visual impairments. For example, the jigsaw and word search puzzles required visual and mouse interaction respectively. Some Scratch projects, including **Drum Kit 2.0**, have keyboard shortcuts and produce auditory feedback, so they are more accessible. Note that you should list keyboard shortcuts in the HTML page, either before the embedded project, or accessible via a link placed before the embedded code.

Find out more about **accessibility** in the *Appendix A, Accessibility for Online Teaching*.

In your role as a teacher you have the permissions to embed Java applets. However, your class does not have permission in Moodle for security. How are we to allow our class to embed a Scratch project?

## The Scratch embed filter

One way to allow your students to embed a Scratch project for themselves is to install the **Scratch embed** filter, available from the Moodle modules and plugins database (<http://moodle.org/modules>). This is a contributed plugin by the author, which is available for Moodle 1.9 and 2.0. It is released under the GNU General Public License.

Note that you need to be a site administrator to have permission to install this filter.

### Time for action – installing and using the filter

These are the steps to follow to install and use the filter, in Moodle 2.0 (the steps for Moodle 1.9 are very similar):

1. Visit the page for the **Scratch embed** filter in the Moodle modules and plugins database, [http://moodle.org/plugins/view.php?plugin=filter\\_scratchembed](http://moodle.org/plugins/view.php?plugin=filter_scratchembed). Follow the link on the bottom-right of the page to download the appropriate Zip archive for your version of Moodle.

2. Un-compress the Zip archive and copy the directory, which will be called something like `nfreear-moodle-filter_scratchembed-SOMETHING`, to your server and rename it `scratchembed`. For example, on Redhat Linux you might copy it to `/var/www/moodle/filter/scratchembed`.
3. Log in to Moodle as an administrator and go to **Site Administration | Plugins | Filters | Manage filters**. Choose **On** or **Off but available** from the drop-down menu that is next to the label **Scratch embed**.

The site administrator may now hand back to the teacher. Thank you!

To use the **Scratch embed** filter, follow these steps:

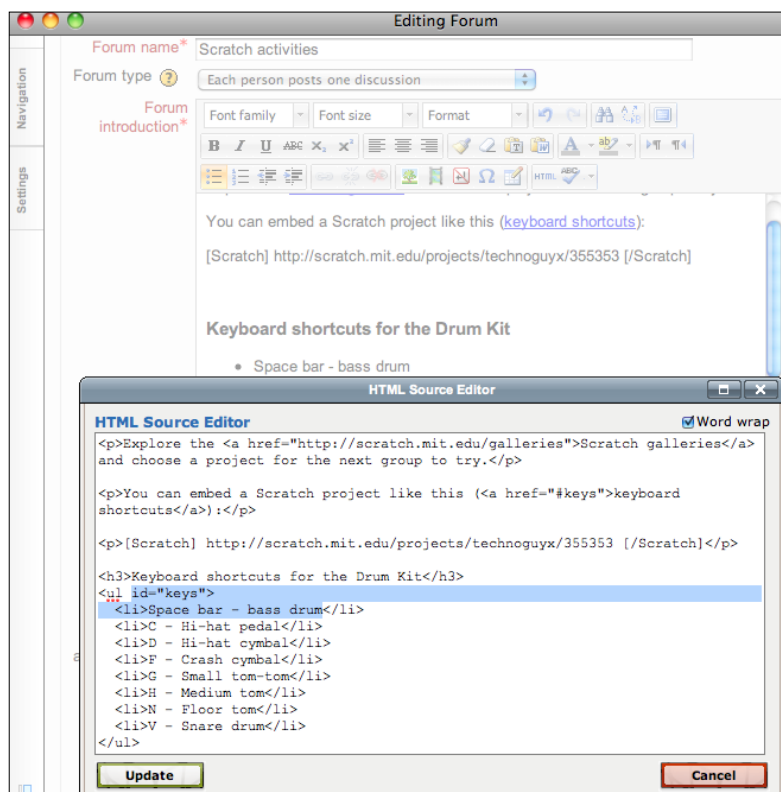
4. Return to our **Scratch activities** forum in the latest course. From the **Settings** side-block, choose **Forum administration** and **Edit settings**.
5. In the **Forum introduction**, delete the previous embed code. Referring to the next figure, this is everything beneath the phrase, **You can embed a Scratch project like this**.
6. Replace it with the link to the project page on the Scratch site, enclosed in `[Scratch]` and `[/Scratch]` square-bracket tags:
 

```
[Scratch]http://scratch.mit.edu/projects/technoguyx/355353[/Scratch]
```
7. As suggested in the previous accessibility tip, list the keyboard shortcuts for the **Drum Kit 2.0** project below the embed code. Press the **Edit HTML Source** button in the editor.
8. Start by adding an ID attribute to the list, using HTML mark-up like the following:
 

```
<ul id="keys"><li>Space...</li></ul>
```
9. Above the embed code put a link to the list with the mark-up:
 

```
<a href="#keys">Skip to keyboard shortcuts</a>
```

The result is shown in the next screenshot:



10. Press the **Update** button to close the **HTML Source Editor** dialog. Then, scroll to the foot of the page and press the button to **Save and display**.

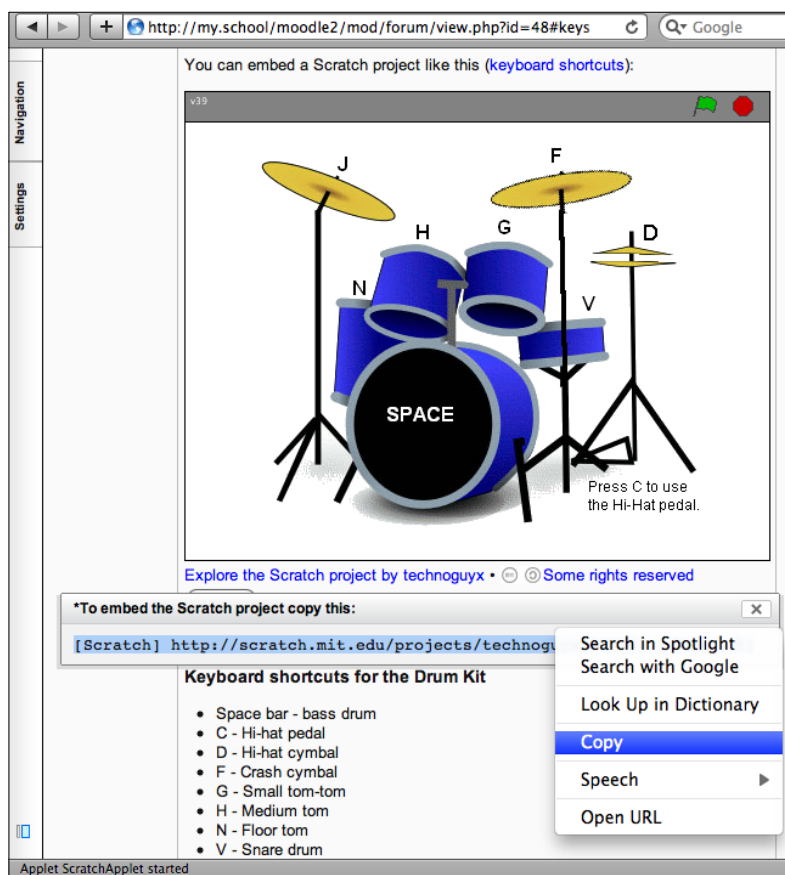
## What just happened?

We installed the Scratch embed filter plugin for Moodle. Then we went back and edited our forum activity to use the filter.

We explored how to present the keyboard shortcuts using the unordered list `<ul>` and list item `<li>` elements. We found that we can link or refer to a so-called fragment identifier (`#keys`) from within a hyper-link `<a href="#keys">shorcuts</a>`.

As we discussed before, the filter offers several benefits. It allows your class to safely and easily embed a Scratch project on the site, while avoiding the security restrictions. And, as the next screenshot indicates, when you press the button labeled **Embed** beneath the project, a panel appears containing the snippet of code required to embed that project. This can be copied and pasted elsewhere, to facilitate embedding by example.





So there we have it. You have embedded your example Scratch project in a forum. You can go on to use the Scratch forum in a class activity, where each group of pupils finds and shares their favorite Scratch games.

Now that we have started to incorporate the Scratch community into our class, we may want to keep up to date, with news about what they are doing. How can we achieve this? Next up, we look at the RSS feeds.


## RSS feeds and blocks

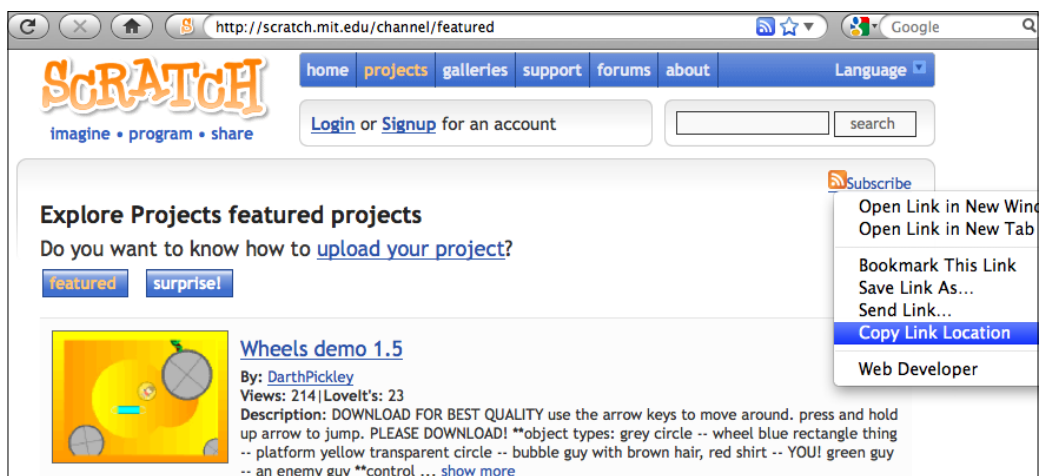
During the previous chapter we added our first side-block to Moodle, for the **Dictionary** plugin (*Integrating a dictionary service* section in *Chapter 8, Stories revisited*). We are going to work with another block now.

An **RSS feed** (or just feed) normally contains dynamic content such as news from another site. The contents of a feed can be displayed in a side-block in your Moodle, perhaps to show updates related to a course activity on a third-party site. We are going to experiment with RSS feeds including an example from Scratch.

## Time for action – adding an RSS feed

This is how we go about adding an RSS feed to our forum activity, in Moodle 1.9 and 2:

1. Visit the featured projects page on the Scratch site, <http://scratch.mit.edu/channel/featured>. On the right, find the link labeled **Subscribe** with the icon . Right-click on it with your mouse and choose **Copy Link Location**. Paste the link (<http://scratch.mit.edu/feeds/getFeaturedProjects>) into your text editor for later.



2. Ensure that **Editing is on** in your course. Go to the **Scratch activities** forum page. In the bottom right you will see a side-block labeled **Add a block**. Choose **Remote RSS feeds** from the **Add...** drop-down menu.
3. Initially you will see the message **RSS feeds are disabled**. Don't worry! Click on the **Configuration** icon for the block.
4. On the next page is a form headed **Configuring a Remote news feed block**. Click on the link labeled **Add/edit feeds** about halfway down the page.
5. Press the **Add a new feed** button on the next page—heading **Nothing to display**. Paste the URL for the feed in the text box labeled **Feed URL**. And click on the **Add a new feed** button.

6. You will be taken to a page listing the new feed. Look out for text like **Last 15 featured projects**. Click on the link labeled **Featured Projects on Scratch[r]** to view the feed.
7. Choose the **MY109** link in the breadcrumb trail then select the link **Scratch activities** from the course main page. Again, choose the **Configuration** icon in the remote news feed block.
8. Our new feed **Featured Projects on Scratch** will appear in the selected box labeled **Choose the feeds....**. Highlight it, and choose **Yes** for **Display each link's description** at the top of the page. Choose **Yes** for **Should a link to the original site... be displayed**. Scroll to the foot of the configuration form and press the **Save changes** button.

### ***What just happened?***

We visited the Scratch website and found the **Featured projects** feed. Then we added a **Remote RSS feed** block in our forum. There followed a slightly convoluted process to add this feed to our **Scratch activities** forum in Moodle.

When you view the feed you will notice that there is an image and some descriptive text for each item. In the case of the Scratch feed the description for each item can be very long. Indeed the feed side-block probably stretches way down your page, despite us only displaying the first five items (the default). Can we improve matters?

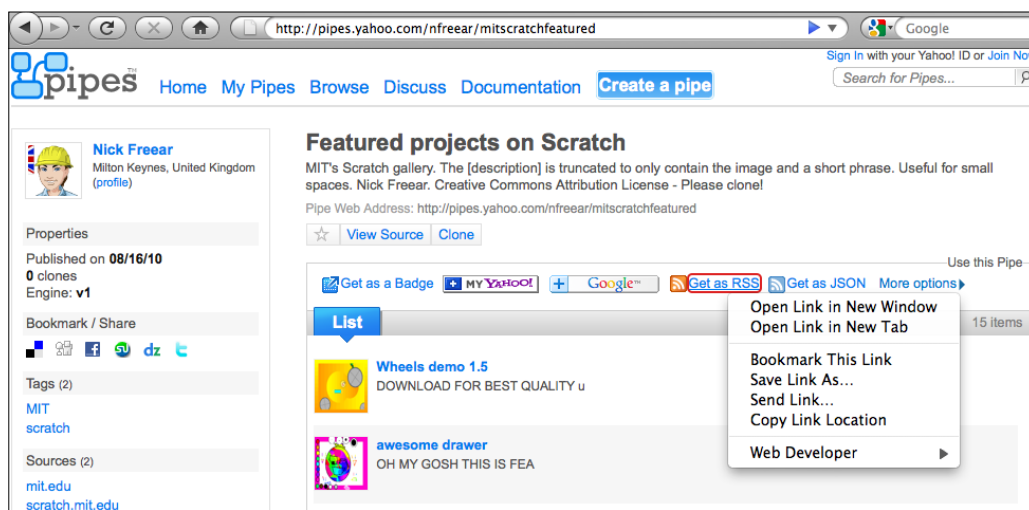
## Improving our RSS feed

Yahoo! has released a clever online tool called **Yahoo! Pipes** (<http://pipes.yahoo.com>). Anyone can log in and create a **Pipe** to consume some data like a feed from the Web, do something with it, and regurgitate it as a new feed. Yahoo! Pipes has a visual interface in which you can drag the building blocks to create your Pipe or script to produce the desired output. We are going to use a pre-defined Pipe to improve the feed displayed in our side block.

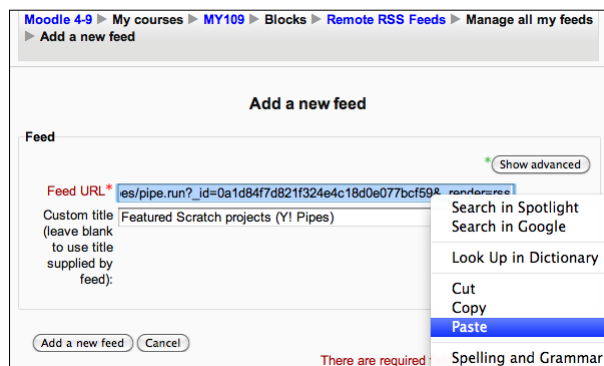
### Time for action – using Yahoo! Pipes

We can follow these steps to replace the Scratch feed with a pre-defined Yahoo! Pipes feed:

1. Visit the link [http://pipes.yahoo.com/nfreear/mit\\_scratch\\_featured](http://pipes.yahoo.com/nfreear/mit_scratch_featured), and you will see something like the next screenshot:



2. Find the link **Get as RSS**, right-click your mouse and choose **Copy Link Location**. Paste the URL to a new document in your text editor ([http://pipes.yahoo.com/pipes/pipe.run?\\_id=0a1d84f7d821f324e4c18d0e077bcf59&render=rss](http://pipes.yahoo.com/pipes/pipe.run?_id=0a1d84f7d821f324e4c18d0e077bcf59&render=rss)).
3. Return to the configuration page for the feed block and follow the same steps we took in the section *Time for action - adding an RSS feed* to add a new feed. When you are prompted to enter the **Feed URL**, paste the link you copied in the last step. It is useful to also set the **Custom title**. I put **Featured Scratch projects (Y! Pipes)**.

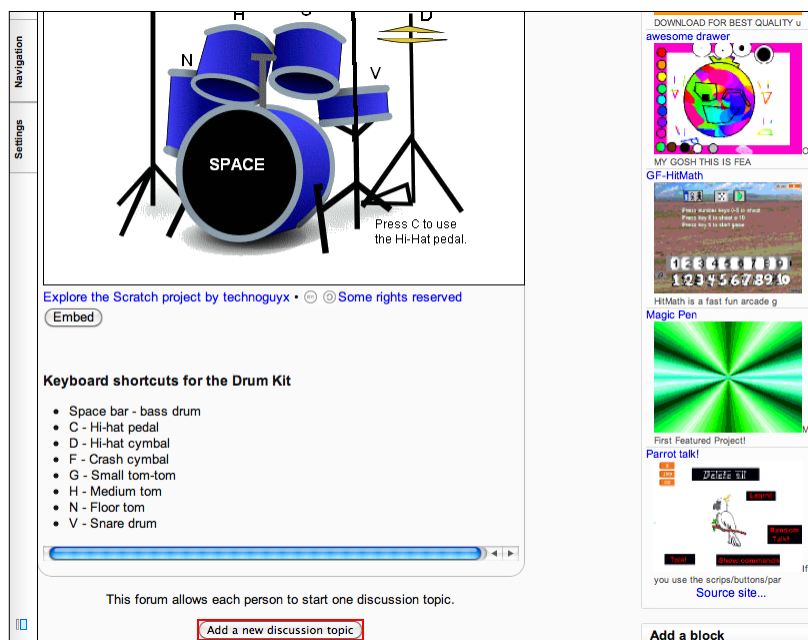


4. In the **Choose the feeds...** select box on the feed configuration page, select the **Pipes** feed in place of the original one. Scroll down and press the **Save changes** button.

### What just happened?

We visited the page for our pre-defined **Yahoo! Pipe** ([http://pipes.yahoo.com/nfreear/mit\\_scratch\\_featured](http://pipes.yahoo.com/nfreear/mit_scratch_featured)), and copied the feed URL. Then we re-configured the feed side-block to use the replacement RSS feed.

When you view the forum activity now, you will see something like the following screenshot. Images and links from the feed are displayed in a column on the right, while the Scratch project displays in the central area of the page:



You can see that the long description from the original featured-projects feed has been truncated, so what you see is hopefully more manageable.

Part of the power of **Yahoo! Pipes** is the ease with which you can **Clone** or copy someone else's Pipe. Feel free to do this with our pre-defined Pipe and start experimenting with some changes.

Note that in these two sections we added a remote feed side-block to our forum activity. You can of course add a feed block to the course main page, other activity modules and the home page of your site.

## Feeds

We have used a feed within our Moodle course website. I should point out that RSS feeds were originally developed so that individuals could keep up with news from multiple websites via feed-reader (news aggregator) desktop or online software. For more, see Wikipedia and the **Open Directory Project**, [http://dmoz.org/Computers/Software/Internet/Clients/Web/Feed\\_Readers](http://dmoz.org/Computers/Software/Internet/Clients/Web/Feed_Readers).

We have explored a feed provided by the Scratch website, but many sites around the web provide feeds, which you may find a useful companion to your teaching:

- ◆ Scratch featured projects, by MIT, <http://scratch.mit.edu/channel/featured/>:
  - RSS feed, [http://pipes.yahoo.com/nfreear/mit\\_scratch\\_featured](http://pipes.yahoo.com/nfreear/mit_scratch_featured)
- ◆ Games made by Scratch gods, by blizzari, <http://scratch.mit.edu/galleries/view/105904>:
  - RSS feed, [http://pipes.yahoo.com/nfreear/scratch\\_gods](http://pipes.yahoo.com/nfreear/scratch_gods)
- ◆ Scratch MIT on Flickr: <http://flickr.com/groups/scratchmit/pool/>.
  - Feed, [http://api.flickr.com/services/feeds/groups\\_pool.gne?id=832913@N23&lang=en-us&format=atom](http://api.flickr.com/services/feeds/groups_pool.gne?id=832913@N23&lang=en-us&format=atom)
- ◆ CBBC Newsround, by the BBC, <http://news.bbc.co.uk/cbbcnews/>:
  - RSS feed, [http://newsrss.bbc.co.uk/rss/cbbc\\_news/homepage/rss.xml](http://newsrss.bbc.co.uk/rss/cbbc_news/homepage/rss.xml)
- ◆ CBBC Newsround Pictures, by the BBC, <http://news.bbc.co.uk/cbbcnews/>:
  - RSS feed, [http://newsrss.bbc.co.uk/rss/cbbc\\_news/pictures/rss.xml](http://newsrss.bbc.co.uk/rss/cbbc_news/pictures/rss.xml)

- ◆ World News for Schools (podcasts), BBC, <http://bbc.co.uk/podcasts/series/wnc>:
  - RSS feed, <http://downloads.bbc.co.uk/podcasts/cbeebies/wnc/rss.xml>
- ◆ iSpot, your place to share nature. An OPAL project by The Open University\*, <http://ispot.org.uk/>:
  - RSS feed, [http://www.ispot.org.uk/feeds/latest\\_observations.rss](http://www.ispot.org.uk/feeds/latest_observations.rss)



\* I should declare an interest here. I work for The Open University, and some of my colleagues built the excellent iSpot website.

We've tried our hand at embedding Scratch projects created by children of all ages. Let's turn our attention to some serious science. But first, try this exercise.

## Have a go hero

At the start of this chapter we concentrated on embedding **Scratch** projects created by the community. Scratch is an educational programming language and development environment that allows you to create interactive and animated projects using pictures, and sounds. Why not download the Scratch development software (<http://scratch.mit.edu>)? Find an existing project in the community pages (<http://scratch.mit.edu/galleries>) and download it (this requires registration). Open the project and try modifying it in some simple way. Then embed your customized project in a course.

You may wish to look at online video tutorials ([http://info.scratch.mit.edu/Video\\_Tutorials](http://info.scratch.mit.edu/Video_Tutorials)), the site that accompanies **Michael Badger's** book, *Scratch 1.4* (<http://www.scratchguide.com>), the forums (<http://scratch.mit.edu/forums>), and other resources.

Hint: for simplicity it is advisable to select a project that does not contain many scripts or sprites!

## Science simulations

**PhET** is a project at the **University of Colorado** whose mission is to create "Fun, interactive, research-based simulations of physical phenomena..." (<http://phet.colorado.edu/>). It is funded by educational and charitable foundations from around the world, and the simulations are available free online in a range of languages. We're going to incorporate some into our course.

The PhET website and simulations are copyright of the University of Colorado, and simulations are dual-licensed under a **Creative Commons Attribution License** (executables) and the **GNU General Public License** (program source code and executables). For more details on licensing see, <http://phet.colorado.edu/en/about/licensing>.

## Time for action – incorporating PhET simulations

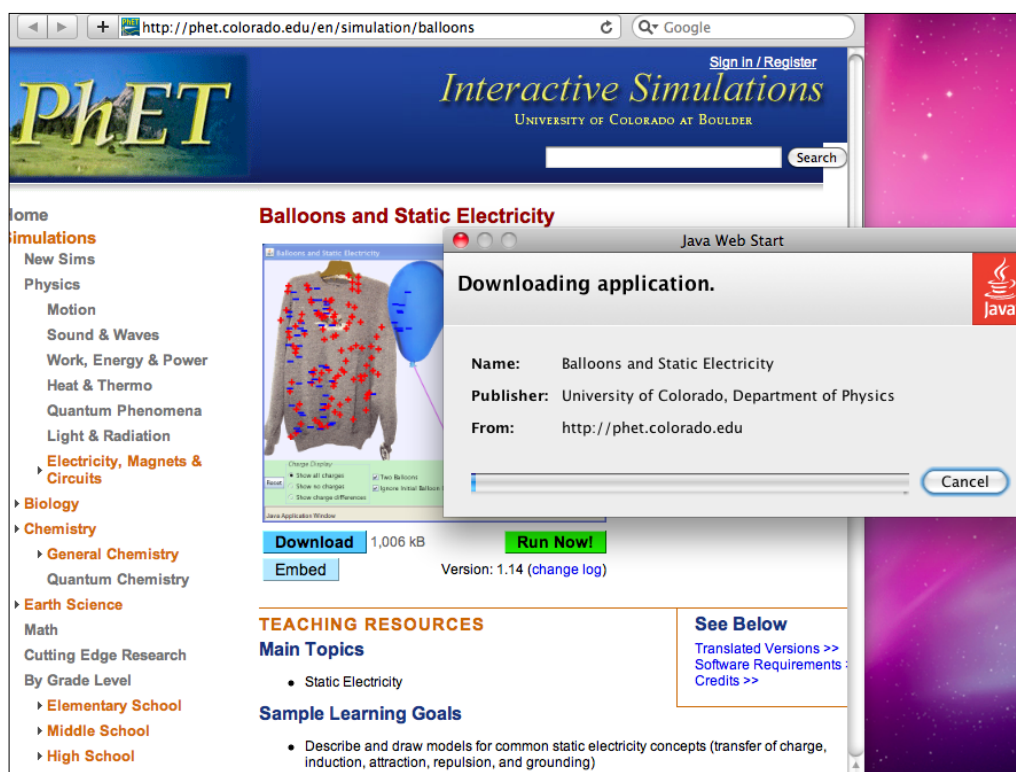
Imagine that you are planning a lesson about electricity. How would you enhance your pupils' learning? Try these steps to incorporate a relevant simulation:

1. Explore the PhET website, particularly looking at the simulations categorized for elementary school as shown in the next screenshot, <http://phet.colorado.edu/en/simulations/category/by-level/elementary-school>:

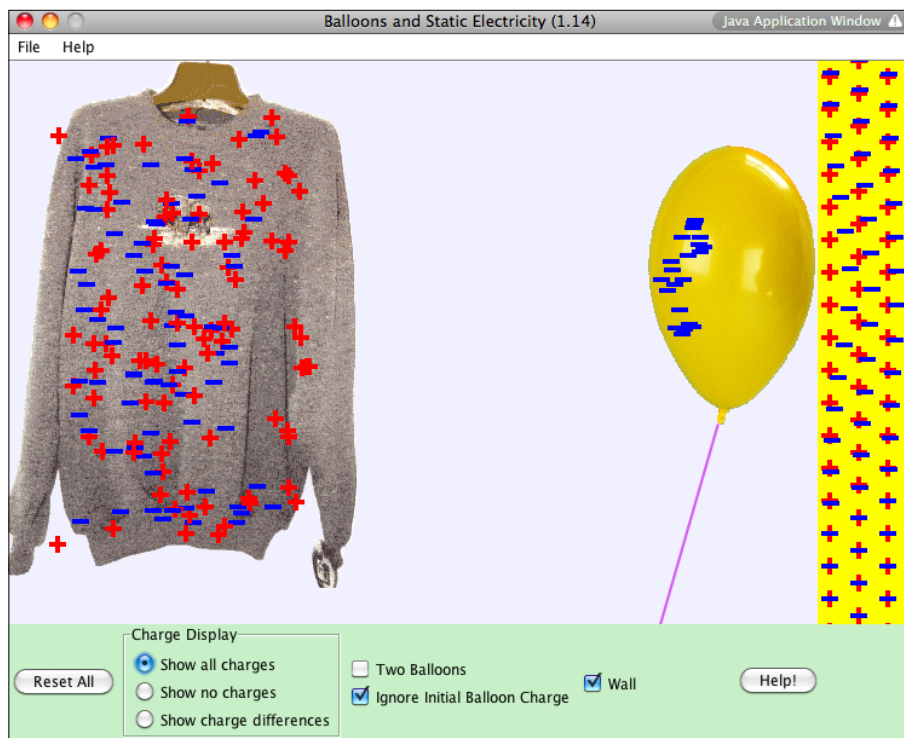




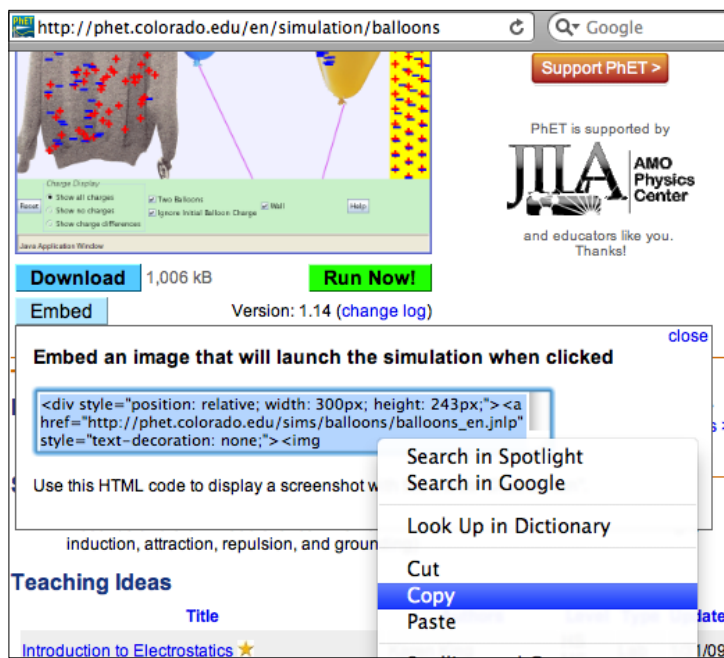
- We will use the **Balloons and Static Electricity** simulation, <http://phet.colorado.edu/en/simulation/balloons> (by Sam Reid, developer and Wendy Adams, interviewer). Visit this page now in your browser, and click the **Run Now!** button below the illustrative screenshot.
- Your browser will prompt you with a dialog window, including a question like **What should Firefox do with this file?** Choose to run or **Open with Java Web Start** and press **OK**. As shown in the next screenshot, a progress window will appear with the title **Java Web Start** and the heading **Downloading application:**



4. The software will launch after Java completes the download and verification steps. You will be able to try out the simulation by rubbing the balloon on the jumper. Watch what happens.

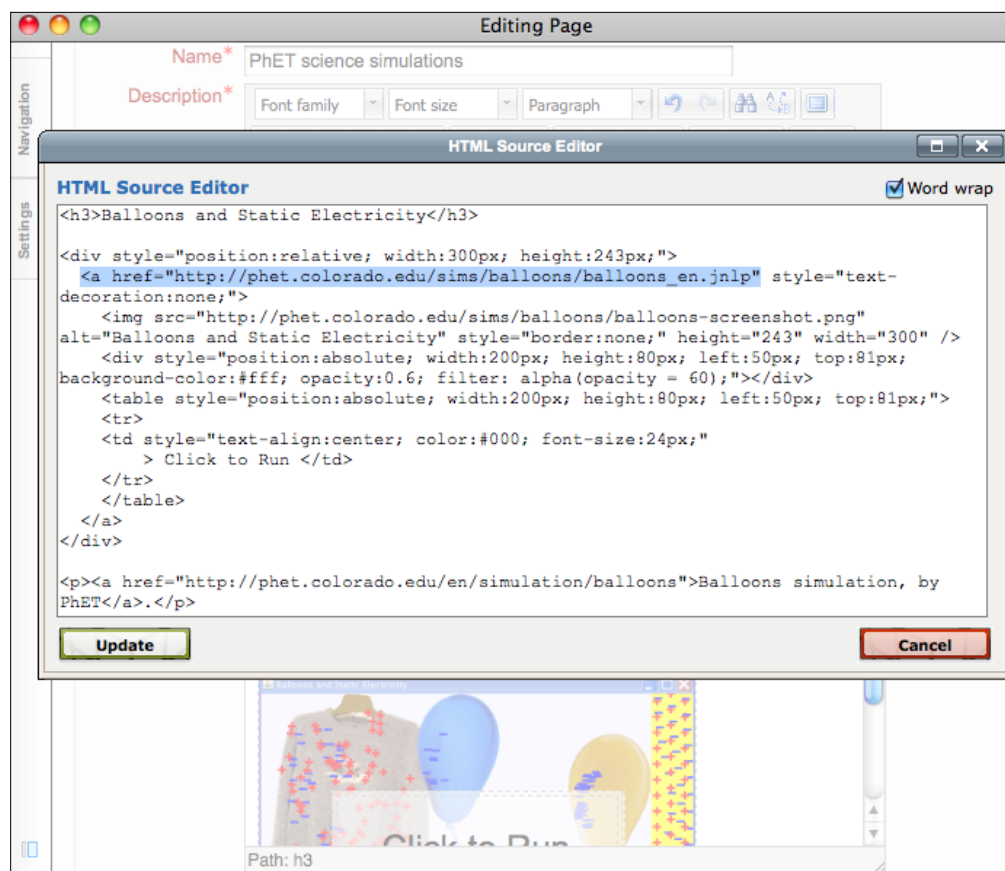


5. To incorporate this simulation in your course, return to the simulation page on the **PhET** website. Press the **Embed** button. Copy the HTML snippet that appears in the popup dialog, shown as follows and paste it to your text editor for a later step:



6. Log in to the **MY109** course page as a teacher. Turn editing on and choose **Page** from the **Add a resource...** drop-down menu in topic 2.
7. Give your page a name, for example **Science simulations from PhET**.

8. In the rich-editor for **Page content** press the **Edit HTML source** button. Paste in the embed code you copied previously from **PhET**. Scroll to the foot of the **HTML Source Editor** dialog and press the **Update** button.



9. Go to the bottom of the **Page** resource form and press the button to **Save and display**.

Try it out. Press the **Click to Run** link in the middle of the preview image, and enjoy.

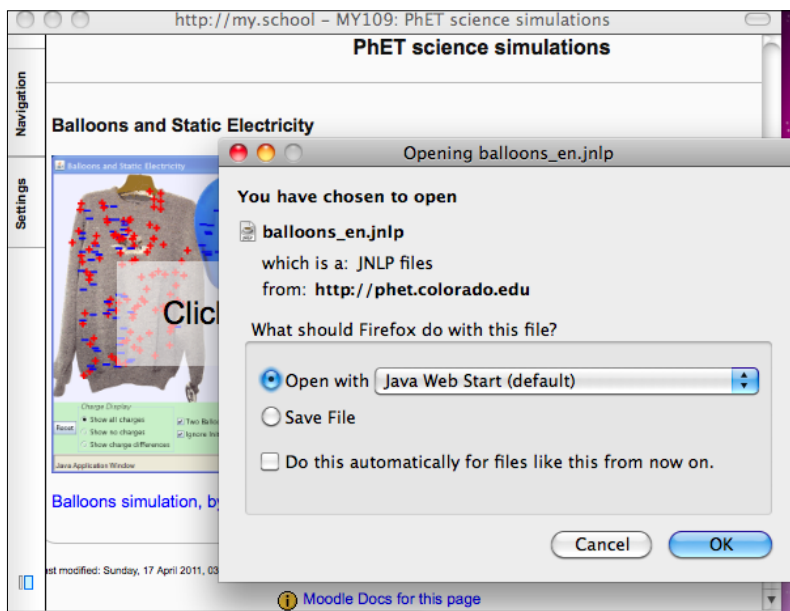
## What just happened?

We explored the PhET website and tried a simulation of balloons and static electricity. Then we copied the HTML snippet from the website, and pasted it into a page resource in our course. Along the way, we learned what warning dialogs to expect from our Web browser.

An interesting part of the HTML snippet that we copied is the link to a JNLP file:

```
<a href="http://phet.colorado.edu/sims/balloons/balloons_en.jnlp"...
```

This prompts your system to launch the Java simulation program, which runs independently of your Web browser. Below you can see a screenshot of the HTML snippet in a Moodle resource, and the prompt that you see when you click on the link to the JNLP file:



Note that since most of the PhET simulations are standalone programs, you can embed as many of these snippets as you wish on a single web page. There is not the same potential to overload your computer that we found with the Scratch projects, which run in a Java applet within your browser.

Unlike the Scratch applets there is no direct embedding of a program with most PhET simulations. Instead you can copy an HTML snippet that links out to the standalone simulation. So, with fewer security implications, you and your class are free to copy these raw link-snippets into comments and resources in your course website. There they can be useful and fun illustrations of key principles in science and maths.

## Have a go hero

We looked at the excellent science and maths simulations from the PhET project. Explore the site further, and consider incorporating more into your maths and science teaching.

Possible solutions: I found a maze and lunar landing simulation for example.

## HTML5 jigsaw

In *Chapter 7, Interactive Puzzles* we looked at Flash-based games and jigsaw puzzles. There are more recent technologies than Flash, which are well suited to learning games and puzzles. Specifically, **HTML5** is at the time of writing the next generation web standard. It contains many new features, including the one that we will take advantage of in this section—the `<canvas>`.

To try this activity you should install a recent version of one of these Web browsers. Note that all but Internet Explorer are available for Windows and Mac OS X, and all are free to download and use. Here are the minimum supported browser versions, with download links:

- ◆ Mozilla **Firefox** 3.0 onwards: <http://getfirefox.com/>
- ◆ **Opera** 10 onwards: <http://opera.com/download/>
- ◆ Google **Chrome** 3.0 onwards: <http://google.com/chrome>
- ◆ **Safari** 3.0 onwards: <http://apple.com/safari/download/>
- ◆ Microsoft **Internet Explorer 9** onwards: <http://microsoft.com/windows/internet-explorer/>
- ◆ Google **Chrome Frame** for Internet Explorer 6, 7, and 8: <http://google.com/chromeiframe>

Note that Google Chrome Frame may be useful if you do not have permissions to install a new browser on the computers in your classroom.

The information above was taken from **Mark Pilgrim's** book titled *HTML5: Up & Running* and the website of the book is, <http://diveintohtml5.org/canvas.html>.

### Time for action – exploring the jigsaw

These are the steps to follow to try out and configure the jigsaw in a compatible browser:

1. Visit the jigsaw page at <http://raymondhill.net/puzzle-rhill/>. You will see a page with the title **Jigsaw Puzzle by Raymond Hill...**, and six tabs across the top of the page. The **Solve** tab will be selected, and a default jigsaw is pre-loaded. Have a play.
2. Now choose the **Create** tab. You are presented with a form as shown below. You can start to create a simple demonstration jigsaw for your class. I chose **Straight** from the drop-down menu labeled **Cut**, and a **Screen size** of **800x600** (pixels).

3. Then choose:

- ❑ 6 for the **Suggested number of pieces**
- ❑ **Regular** for the **Complexity of puzzle pieces**
- ❑ **No rotation** for **Whether pieces can be rotated**

http://raymondhill.net/puzzle-rhill/jigsawpuzzle-rhill.php?pi

Solve Presets **Create** Share OpenSocial About

Cut **Straight**

Screen size **800x600** (Screen size determine work area size, which determine puzzle size, which influence the size of individual pieces)

Suggested number of pieces **6** (This number is used as a guide only, restrictions may apply. More pieces = smaller pieces)

Complexity of puzzle pieces **Regular**

Whether pieces can be rotated **No rotation**

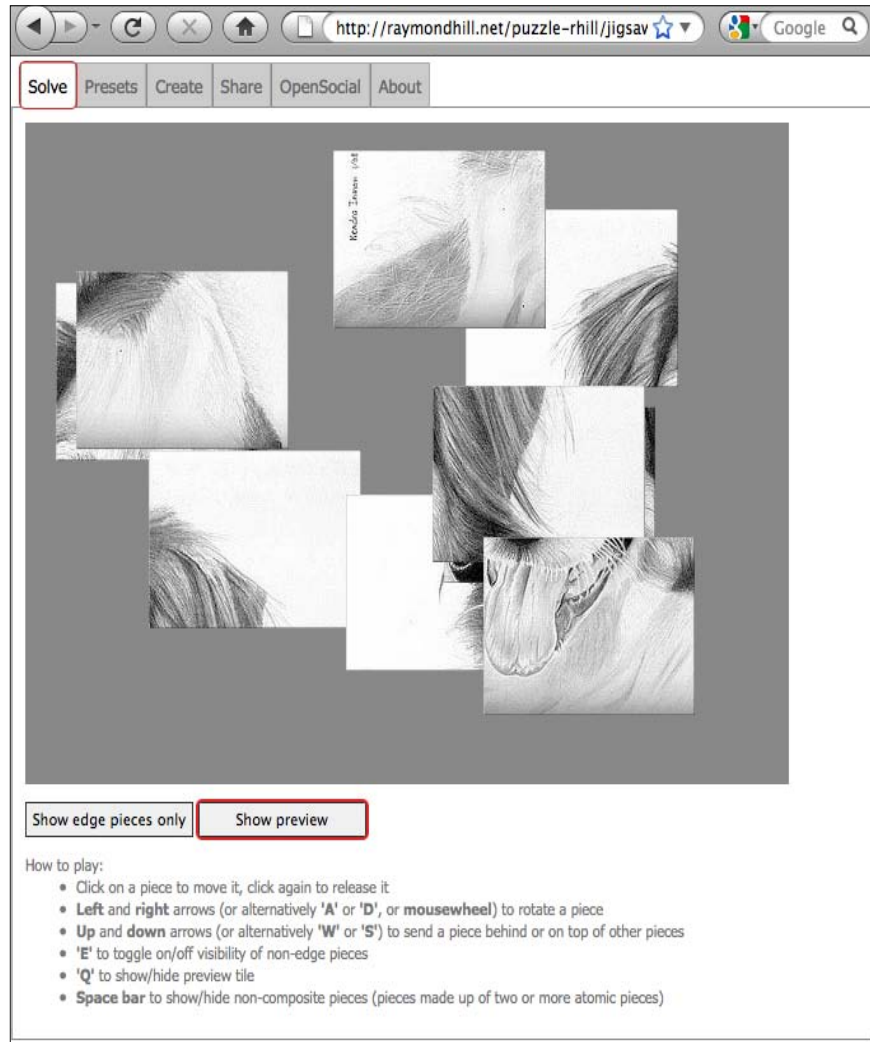
URL of picture to use  (Of interest: [general collections of public domain images](#))

Beware: Clicking on the "Create" button will cause the current jigsaw puzzle in the "Solve" tab to be replaced. Also, note that the more pieces there are, the longer it will take to generate the jigsaw puzzle, so you might experience a delay.

**Create**

4. We are going to reuse the Creative Commons-Attribution Licensed image that we used in the *Chapter 7* jigsaw. It is **Sephie\_Finished**, by sketchr/**Kendra Goering** (<http://flickr.com/photos/kendradeann/2184917410/>). The URL for the image is [http://farm3.static.flickr.com/2380/2184917410\\_a8ba85081d\\_d.jpg](http://farm3.static.flickr.com/2380/2184917410_a8ba85081d_d.jpg).

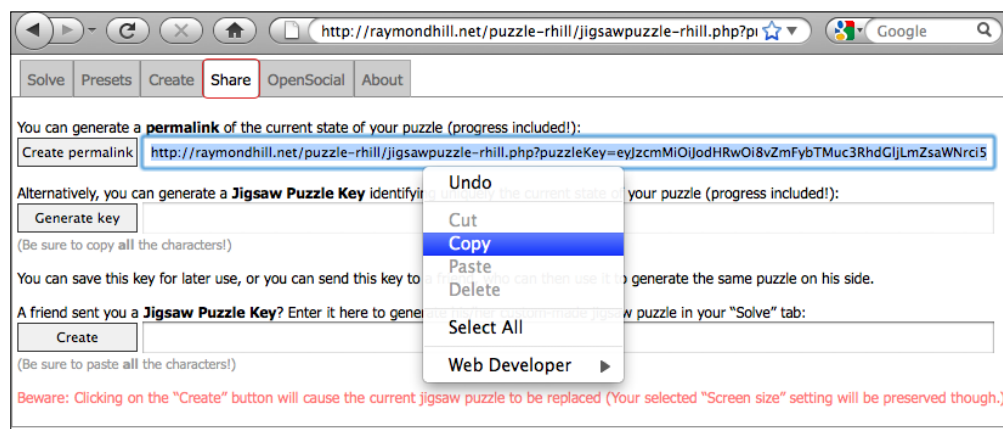
5. Paste the URL with the .jpg extension into the box labeled **URL of picture to use**, and press the **Create** button.



6. You will see the **Solve** tab as shown above. At this point you may be tempted to have a go, but just hold on! First, go to the **Share** tab and press the **Create permalink** button at the top.
7. A link will appear in the neighboring text box, as shown in the following screenshot. You should copy it to the clipboard and paste it to a text editor for the next stage. It will take the following form:



`http://raymondhill.net/puzzle-rhill/jigsawpuzzle-rhill.  
php?puzzleKey=eyJzcmMiOi...(a very long key)...dGVwcyI6MX0 .`



And now you can give your jigsaw a try!

## What just happened?

In the preceding section we visited the page for **Raymond Hill's** Jigsaw puzzle, which uses the HTML5 `<canvas>`. The puzzle is licensed under a **Creative Commons Attribution Non-Commercial Share-Alike License**. After a little experimentation we reused an image from Flickr to create our own simple jigsaw. Then we generated a **permalink** for our puzzle. We copied this long URL, which contains a unique `puzzleKey` parameter, to a text editor for use in the next section. Note that this key saves the **state** of the puzzle; hence we generated it before attempting to solve the puzzle.

## Creating your own jigsaw activity

We are going to use the permalink that we copied in the previous section to embed the jigsaw in our course. We can then build an activity where the class will create their own jigsaw and get their classmates to try them out.

This will develop their search, IT and visual pattern recognition skills, and their collaborative abilities.

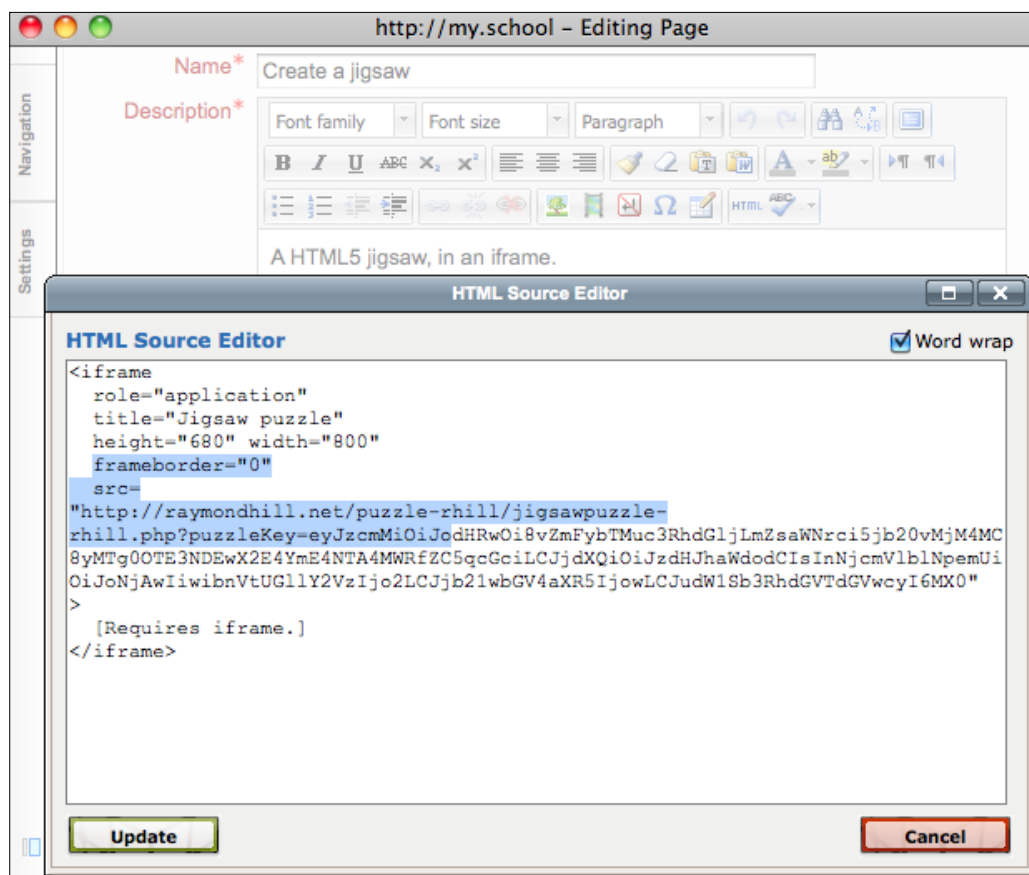
## Time for action – using an iframe

These are the steps to follow to embed the HTML5 jigsaw in Moodle 2. We will use a Page resource:

1. Log in to your course site as a teacher and go to the **MY109** course main page.
2. Scroll to topic 3, and choose **Page** from the **Add a resource...** drop-down menu. Add a **Title** and **Description**.
3. In the editor for **Page Content** press the **Edit HTML source** button. Enter the following HTML snippet in the **Edit HTML Source** dialog, substituting your saved permalink in the `src` attribute as appropriate:

```
<iframe
  role="application"
  width="800" height="680"
  frameborder="0"
  title="Jigsaw puzzle"
  src=
"http://raymondhill.net/puzzle-rhill/jigsawpuzzle-rhill.
php?puzzleKey=eyJzcmMiOiJodHRwOi8vZmFybTMuc3RhdGljLmZsaWNrci5jb20
vMjM4MCM4yMTg0OTE3NDEwX2E4YmE4NTA4MWRfZC5qcGciLCJjdXQiOiJzdHJhaWdod
CIsInNjcmlVbGlNpemUiOiJoNjAwIiwibnVtUGllY2VzIjo2LCJjb21wbGV4aXR5I
jowLCCJudW1Sb3RhdGVtdGVwcyI6MX0">
  [Requires iframe.]
</iframe>
```

Below you can see what the snippet will look like in the Moodle editor. Note that line-breaks have been added to the mark-up for clarity.



4. Press the **Update** button to close the **Edit HTML Source** dialog. Then scroll to the end of the form and press the button to **Save and display**.

### ***What just happened?***

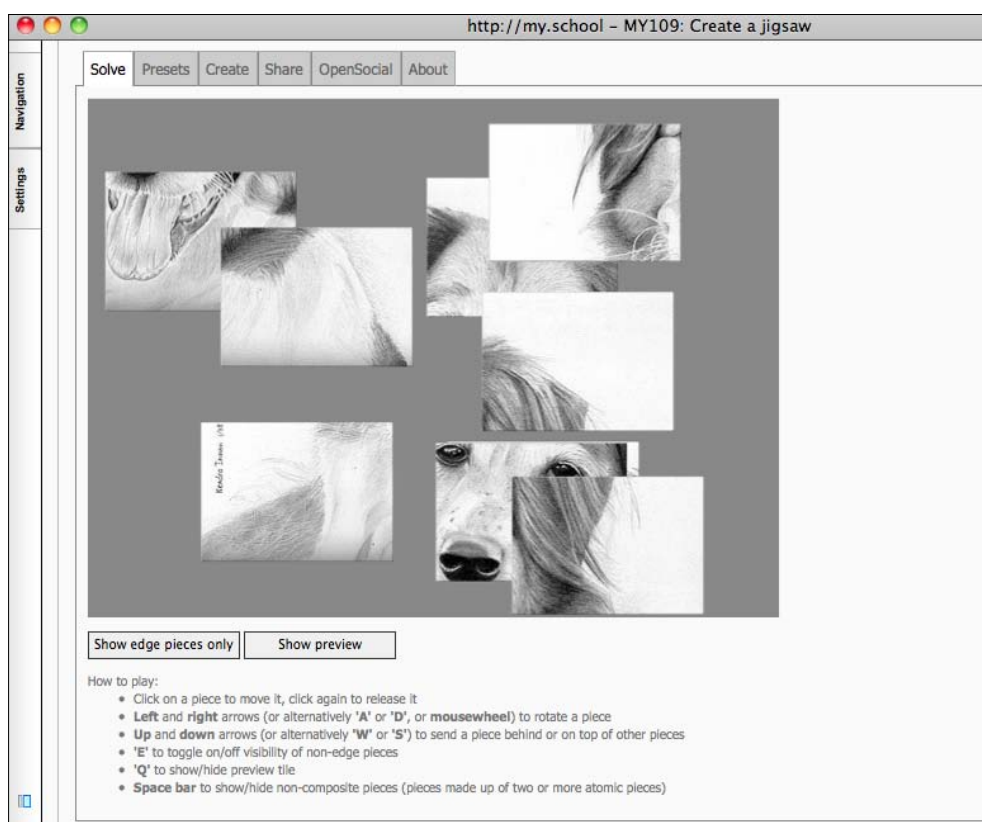
We added a page resource to our course, and embedded our jigsaw within it. We used a HTML `<iframe>`, which is a flexible way of embedding one web resource within another. **Raymond Hill's** jigsaw is well suited to an `<iframe>` approach as the page does not contain much around the jigsaw—the other main features are the tabs at the top labeled **Solve**, **Presets**, and so on.

We added a title attribute to the `<iframe>` to improve its accessibility, and set the `frameborder` to 0 to make it look more integrated. The `<iframe>` is large at 800 pixels wide, and the width and height were found from experimentation and are sufficient for the jigsaw.



Note that an `<iframe>` cannot be used in activities like **Forum** or **Wiki**, and if you try to include one, Moodle will simply remove it when you save the activity. Also, your students will not be able to use this mark-up in their submissions on the site.

To incorporate this activity into your teaching, try linking to it from a forum. Use the forum to set a challenge for your class. They should try your example jigsaw. Then they can use a site like Flickr; for example a search for **house** looks like <http://flickr.com/search/?q=house>. Note that at the time of writing, if the user is not logged in, a Flickr-search is set to **safe**, so potentially harmful content is filtered. Once they have selected an image, and copied the link, your students should create a jigsaw, and post the permalink for it to the forum. A classmate can try the jigsaw.



So there we have it. A final embedding activity, which will allow the class to share their creativity while having fun.

## Have a go hero

We used an `<iframe>` hack to embed a HTML5 jigsaw puzzle. There is a lot happening in the HTML5 area, so venture out and explore. Find another open-source game, or a game that does not preclude embedding. Explore how you would embed the game, and how you would incorporate it into your teaching.



Hint: Avoid games whose terms and conditions use phrases such as "...you may use the Websites... only for personal... purposes" and "You may not... frame any portion of the Websites or content...".

Possible solution: an example would be **Processed Tower Defense** (<http://ptdef.com/>) by Will Larson and Pete Burns, available under an open-source MIT License.

## Pop quiz

Here are some quick questions to check your understanding of this chapter. Careful, there may be more than one correct answer.

1. Both the **Scratch** and **PhET** activities in this chapter used a particular technology. What was it?
  - a. JavaScript
  - b. Flash
  - c. Java
2. We found that the Scratch embed filter had a number of benefits. What were they?
  - a. Students can securely embed a Scratch project
  - b. The filter obscures the embed code for the reader
  - c. It makes embedding more usable and less error-prone

## Summary

We learned a great deal in this chapter about how to embed interesting bits of the wider Web directly into our courses. And we explored some ideas about how to incorporate the activities into our classroom teaching.

Specifically, we:

- ◆ Introduced the Scratch programming environment
- ◆ Embedded a Scratch project in a course using some copied HTML embed code
- ◆ Used the Scratch embed filter for Moodle to facilitate the embedding of Scratch projects by the whole class
- ◆ Added an RSS feed to our Scratch activity
- ◆ Linked to a PhET science simulation from our Moodle course
- ◆ Embedded a HTML5 Jigsaw puzzle in a course resource

We also discussed security, musical applets, side-blocks, Yahoo! Pipes, and iframes.

We've learned about embedding resources from the Web. And throughout the chapters we've developed many useful learning activities. So, we should really look at course backup and restore, and administration. This is the topic of the final chapter.

# 10

## Administration

*Throughout this book and as you get to grips with Moodle, you will create many valuable resources in your course websites. It is important to be able to back up and restore them. You may wish to record private notes about your students' progress, write blog posts, and so on. In this chapter, we will explore a number of administrative, reporting, and communication tools within Moodle.*

Specifically, in this chapter we will:

- ◆ Learn how to create a course backup
- ◆ Explore how to restore a course backup
- ◆ Create teacher notes
- ◆ Explore blogs
- ◆ Introduce the Gradebook
- ◆ Learn about the community and plug-ins

So let's get on with it...

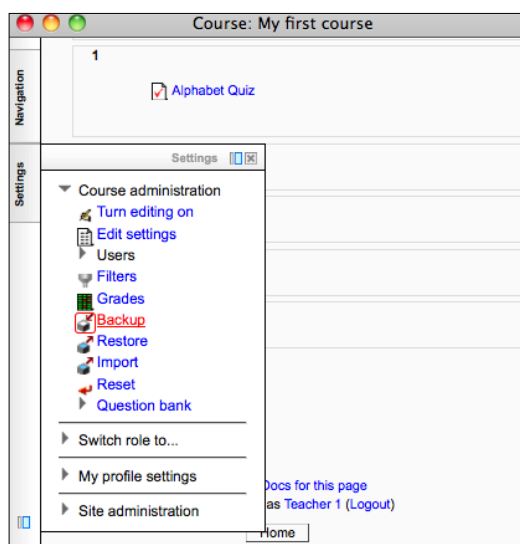
### Course backup

Course and activity backup is a useful tool to give you peace of mind as an online teacher. As we will find out, the role and permissions that you have in a course determine what you can back up.

## Time for action – creating a course backup

These are the steps to follow to create a backup of a course in Moodle 2:

1. Log in to your Moodle system as a teacher.
2. Return to the first course you created, **MY101**.
3. In the **Settings** side-block, choose **Course administration**, then **Backup**.



4. You will find yourself on a page called **Initial settings**. This is stage one of five. You may not have permissions to change the first option, **Include enrolled users**. A padlock and a red cross on the **Initial settings** page will indicate this.

### Roles and permissions

You may find that you can't tick any of the un-ticked boxes in the backup **Initial settings** page. If this is the case, in Moodle 2 ask your IT support to assign the system role of **Manager** to you. This is in addition to the **Course creator** role.

Your IT support or site administrator should use the **Site administration** block and go to **Users | Permissions | Assign system roles**.

They may also wish to visit the **User policies** form via **Site administration | Users | Permissions**, and look at the **Creators' role in new courses** setting. Consider changing this from **Teacher (editingteacher)** to **Manager (manager)**.

If your permissions cannot be changed you will be able to back up the course without users and course history.





5. The checkboxes labeled **Include activities**, **Include blocks** and **Include filters** are already ticked and should remain so.
6. You will probably wish to leave the top two options, **Anonymize user information** and **Include user role assignments** un-checked. You may well wish to tick **Include user files**, **Include comments**, **Include user completion details**, **Include course logs** and **Include grade history**. Do this now.

Moodle 4.9 > My courses > MY101 > Backup > Initial settings

1. Initial settings and review    2. Schema settings    3. Confirmation    4. Perform backup    5. Complete

**Backup settings**

- Include ☒ enrolled users
- Anonymize ☐ user information
- Include ☒ user role assignments
- Include ☒ user files
- Include ☒ activities
- Include ☒ blocks
- Include ☒ filters
- Include ☒ comments
- Include ☒ user completion details
- Include ☒ course logs
- Include ☒ grade history

Cancel    Next

7. Then press the **Next** button to the bottom-right of the page.
8. On the **Schema settings** page, two columns of checkboxes will already be ticked for you. Leave them, and press **Next**.

9. Stage three is called **Confirmation and review**. There will be a **Filename**, for example, **backup-moodle2-course-my101-20110422-1532.mbz**. Under **Backup settings** there will be a series of ticks and one cross (X). Under **Included items** there will be two columns of ticks. Scroll down and press **Perform backup**.

http://my.school - Backup course: MY101: Confirmation and review

Filename\* backup-moodle2-course-my101-20110422-1532.mbz

**Backup settings**

- Include ☒
- enrolled users ☒
- Anonymize ☐ user information
- Include user ☒ role assignments
- Include user ☒ files
- Include ☒ activities
- Include blocks ☒
- Include filters ☒
- Include ☒ comments
- Include user ☒ completion details
- Include ☒ course logs
- Include grade ☒ history

**Included items:**

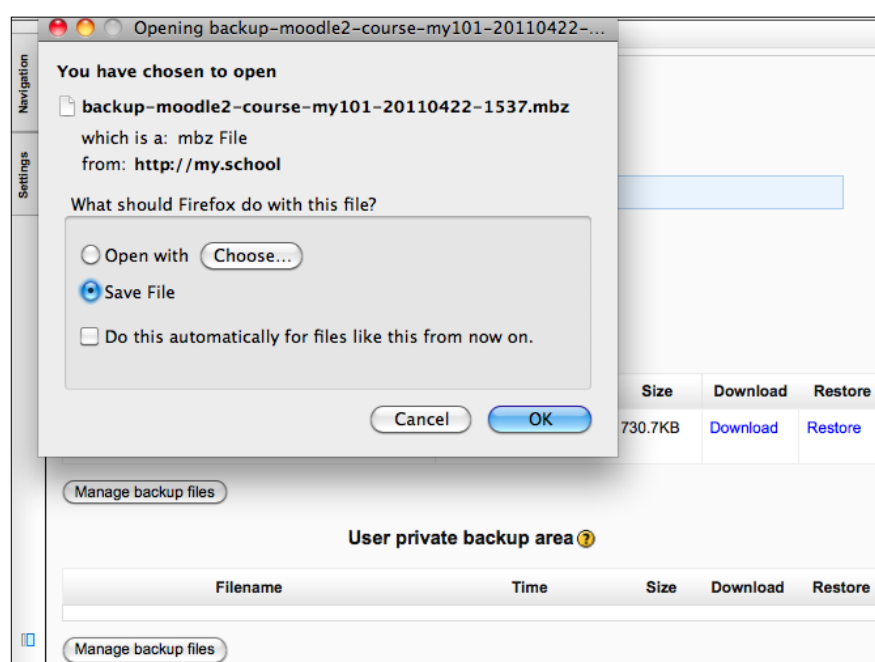
|  |   |
|--|---|
| <b>General</b> <input checked="" type="checkbox"/> | User data <input checked="" type="checkbox"/> |
| News forum <input checked="" type="checkbox"/>     | - <input checked="" type="checkbox"/>         |
| <b>Topic 1</b> <input checked="" type="checkbox"/> | User data <input checked="" type="checkbox"/> |
| Alphabet Quiz <input checked="" type="checkbox"/>  | - <input checked="" type="checkbox"/>         |
| <b>Topic 2</b> <input checked="" type="checkbox"/> | User data <input checked="" type="checkbox"/> |
| <b>Topic 3</b> <input checked="" type="checkbox"/> | User data <input checked="" type="checkbox"/> |

10. If all goes to plan you will see a page with the message, **The backup file was successfully created**. Hit the **Continue** button.

## What just happened?

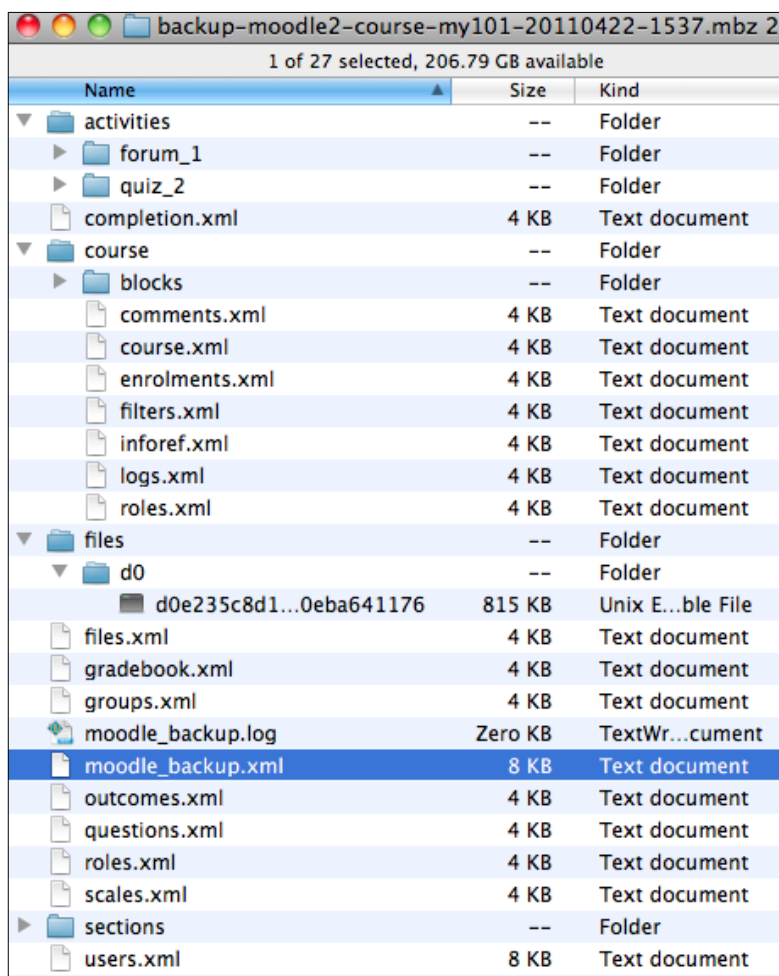
As a teacher, with the role **editingteacher**, you will have the permissions to backup course content. We started by ensuring that you have the **manager** role, so that we can try out backing up student data and student progress too. Note that the manager role is new to Moodle 2.

Having achieved this, we went through the five stages of backing up a course. We landed on a page called **Restore**. This has the headings **Import a backup file**, **Course backup area** and **User private backup area**. Our file will be listed under **Course backup area**. Press the **Download** link next to it, and when prompted save the file to your computer.



When you view the downloaded backup it will have `.mbz` file extension. In Moodle 1.9, the procedure looks very similar, but it will have a `.zip` file extension. In either case, we can look inside the archive.

Copy the **mbz** (Moodle backup Zip) file on your computer, and give the copy a **.zip** file extension. Uncompress the new Zip file, and you will see something like the file and directory structure shown in the following screenshot:



| Name                    | Size    | Kind              |
|-------------------------|---------|-------------------|
| activities              | --      | Folder            |
| forum_1                 | --      | Folder            |
| quiz_2                  | --      | Folder            |
| completion.xml          | 4 KB    | Text document     |
| course                  | --      | Folder            |
| blocks                  | --      | Folder            |
| comments.xml            | 4 KB    | Text document     |
| course.xml              | 4 KB    | Text document     |
| enrolments.xml          | 4 KB    | Text document     |
| filters.xml             | 4 KB    | Text document     |
| inforef.xml             | 4 KB    | Text document     |
| logs.xml                | 4 KB    | Text document     |
| roles.xml               | 4 KB    | Text document     |
| files                   | --      | Folder            |
| d0                      | --      | Folder            |
| d0e235c8d1...0eba641176 | 815 KB  | Unix E...ble File |
| files.xml               | 4 KB    | Text document     |
| gradebook.xml           | 4 KB    | Text document     |
| groups.xml              | 4 KB    | Text document     |
| moodle_backup.log       | Zero KB | TextWr...cument   |
| moodle_backup.xml       | 8 KB    | Text document     |
| outcomes.xml            | 4 KB    | Text document     |
| questions.xml           | 4 KB    | Text document     |
| roles.xml               | 4 KB    | Text document     |
| scales.xml              | 4 KB    | Text document     |
| sections                | --      | Folder            |
| users.xml               | 8 KB    | Text document     |

At first glance this array of items may seem overwhelming; however, there are a few points to note:

- ◆ Normally you will not need to look inside the **mbz** archive!
- ◆ The course backup structure changed between Moodle 1.9 and Moodle 2.
- ◆ The backup archive comprises course, plugin, and user information and content. It does not contain the software plugins themselves.

- ◆ There are sub-directories named `activities`, `course`, `files`, and `sections`.
- ◆ The majority of the files are Moodle XML documents with an `.xml` file extension. These can be viewed in a web browser such as Internet Explorer or Firefox, or a text editor such as Notepad2 or TextWrangler (on Mac OS X).
- ◆ The key file is named `moodle_backup.xml`.
- ◆ Other significant files are `course/course.xml` and `activities/quiz_2/quiz.xml`, `files.xml`, `questions.xml`, and `users.xml`.

The file `moodle_backup.xml` contains information or metadata about the backup itself, the course, and the Moodle system that is the source of the backup. It ties all the other XML files together. The file `course/course.xml` contains further metadata specific to the course itself, and files like `activities/quiz_2/quiz.xml` tell us that there is a quiz activity in section two of the course. Section two equates to **topic 1** of the course, as section one is the summary section at the top of the course.

Note that `activities/quiz_t/quiz.xml` comprises metadata about the quiz and what questions it includes. It does not contain the questions themselves, as these are in the `questions.xml` file, which holds the question bank for the course. This allows questions to be reused between quiz activities.

`files.xml` is composed of metadata about files from the course and from your private file area. This includes all images, multimedia files, documents, and so on that are uploaded.

The `users.xml` file contains the metadata including the username and e-mail address of each person enrolled in the course. It does not include passwords, however it is still sensitive. So keep your backups somewhere secure.

There are a number of reasons you may wish to perform backups:

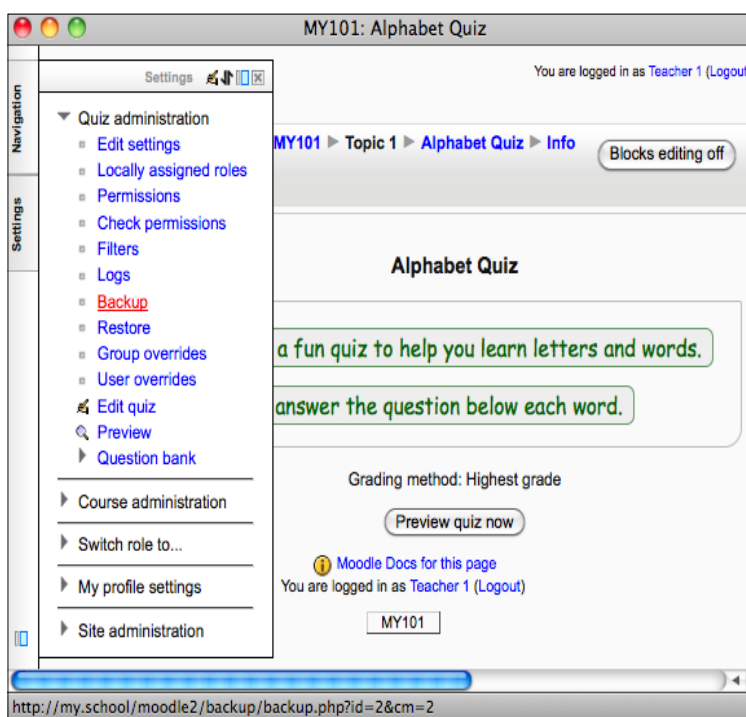
- ◆ For peace of mind. Your system administrator or IT support should be scheduling backups of the whole system, but you may wish to perform your own course-level backups.
- ◆ To roll activities and resources forward to a new course.
- ◆ To transfer courses to a new Moodle system. Note, at the time of writing, Moodle 1.9 courses can be restored into a system running Moodle 2.1, but not 2.0.

## Have a go hero

Try this exercise to increase your understanding of the topic.

Challenge 1. In the previous sections, we stepped through the process of backing up a whole course. It is possible to backup or export individual activities and components too. Look at the following example, where we start the process of backing up an individual quiz activity.

Try backing up some of the individual activities that were created in previous chapters. Return to one of your courses containing questions. Export the question bank and import it into another course.



Now that we can backup a course, we should try restoring it.

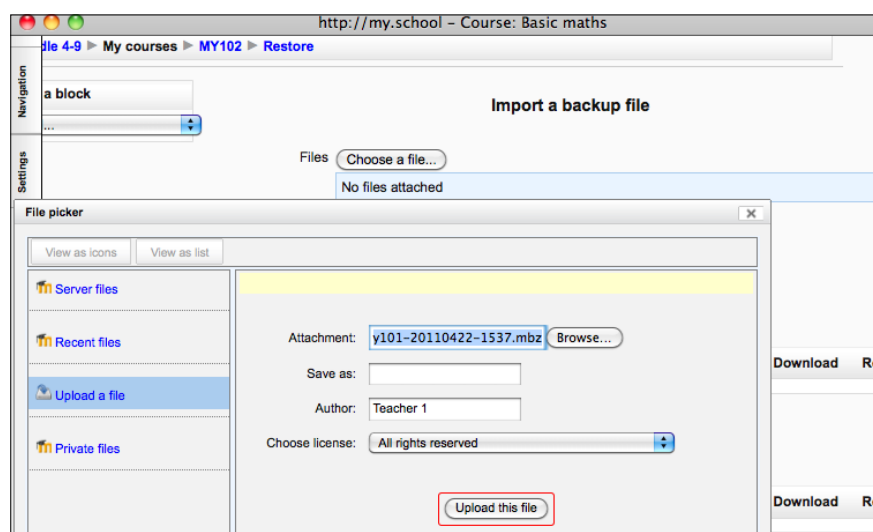
## Restoring a course

In the previous section, we discovered how to perform course-backups. Now we'll turn our hand to restoring a course using the backup archive. This can be done in a variety of ways. We will be restoring into a new course.

## Time for action – course restore

These are the steps to follow to restore a course in Moodle 2. We assume that you have a Moodle backup archive with an .mbz extension on your computer.

1. Login to Moodle as a teacher, and go to an existing course, for example your **MY102** course.
2. In the **Settings** side-block, under **Course administration** follow the link labeled **Restore**.
3. Under the heading **Import a backup file**, press the button labeled **Choose....** In the **File picker** dialog follow the **Upload a file** link in the left-hand menu.
4. Next to the **Attachment:** field press the **Browse** button. Find the **MBZ** file on your computer, then press **OK** in the dialog. Returning to the **File picker** dialog press the button to **Upload this file**.



5. The filename, for example backup-moodle2-course-my101-20110422-1537.mbz will appear below the **Choose a file...** button. Beneath the filename press the **Restore** button.

We are now at the first stage of a seven-stage process. Never fear!

1. The first stage is labeled **Confirm**. You will see a long page with the headings **Backup details**, **Backup settings**, **Course details**, and **Course sections**. This summarizes the metadata and contents of the backup file with tick icons to indicate the included items. No action is required, so press **Continue** at the foot of the page.
2. Stage two is labeled **Destination**. As shown in the following screenshot, there are three options, **Restore as a new course**, **Restore into this course** and **Restore into an existing course**. You will see further sub-options for the latter two, **Merge the backup course into this course** and **Delete the contents of this course and then restore**.

The screenshot shows a web application window titled "Basic maths: Destination". It features a progress bar at the top with seven stages: 1. Confirm, 2. Destination (active), 3. Settings, 4. Schema, 5. Review, and 6. Complete. The main content area is divided into three sections: "Restore as a new course", "Restore into this course", and "Restore into an existing course". In the "Restore as a new course" section, the "Restore as a new course" radio button is selected. Below it, there is a "Select a category" section with a table showing "Miscellaneous" as the selected category. A "Search" button is also present. A "Continue" button is highlighted with a red box. In the "Restore into this course" section, the "Merge the backup course into this course" radio button is selected. In the "Restore into an existing course" section, the "Merge the backup course into the existing course" radio button is selected, and a table of existing courses is displayed.

| Course short name | Course full name    |
|-------------------|---------------------|
| MY109             | Embedding the Web   |
| MY108             | Stories revisited   |
| MY107             | Interactive puzzles |
| MY106             | Fun Games           |
| MY105             | Setting Homework    |

3. Beneath the heading **Restore into an existing course** is a table of your existing courses. We will take the first route, **Restore as a new course**, which means that a new course will be created. Check the radio-button next to the **Miscellaneous** category, then press the first **Continue** button.



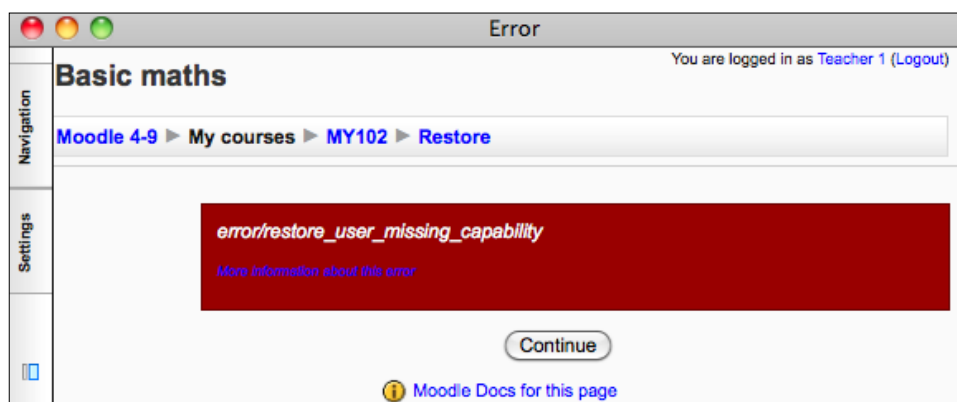
4. The next stage is labeled **Restore settings**. You will see a column of checkboxes, starting with **Include enrolled users** and ending with **Include grade history**. Leave all the boxes checked and press the **Next** button.
5. Stage four is labeled **Schema**. I changed the **Course name** to **My first course - COPY**, and the **Course startdate**. I left the checkboxes ticked and pressed the **Next** button again. Nearly there!
6. Next we have the **Review** stage. There is a column of **Backup settings** - you will see **Include enrolled users** and **Include grade history** again. Then there is the **Course settings** section. If everything looks OK, scroll down the page and press the button to **Restore now**.

### ***What just happened?***

We went to a course main page, **MY102** in our example. We went to the **Restore** page and uploaded the **MBZ** Moodle backup archive that we created previously. Then we followed the process to restore it, choosing to restore into a new course.

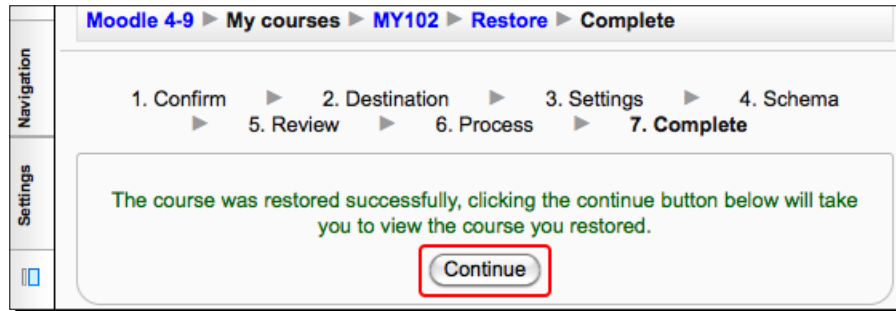
We can see that if we wish to restore by creating a new course it does not matter which of our courses we are in to commence the restore process.

You need sufficient privileges to restore the specific backup. So if it contains user data, as the archive we created previously did, then you need to have the **manager** role to restore it. Otherwise, you will see a similar error as shown in the following screenshot—**error/restore\_user\_missing\_capability**:



A **course creator** can restore courses without user data; someone with the **manager** role can restore courses with user data. If you run into problems, ask your friendly IT support person to check your system role.

When the restore is finished you will see the phrase **The course was restored successfully**, as shown in the following screenshot. You can press the **Continue** button.



Return to the home page for your Moodle system and you will see the restored course **My first course - COPY**. Note that as you were a teacher in the original course you will be listed as a teacher in this course. Although we restored enrolled users, your user account is not duplicated. And, all the files that existed in the original course, for example images, are copied for our duplicate course, even if it lives on the same Moodle site.

We move from the useful course backup and restore functions to the equally useful capability to record notes on the progress of your class.

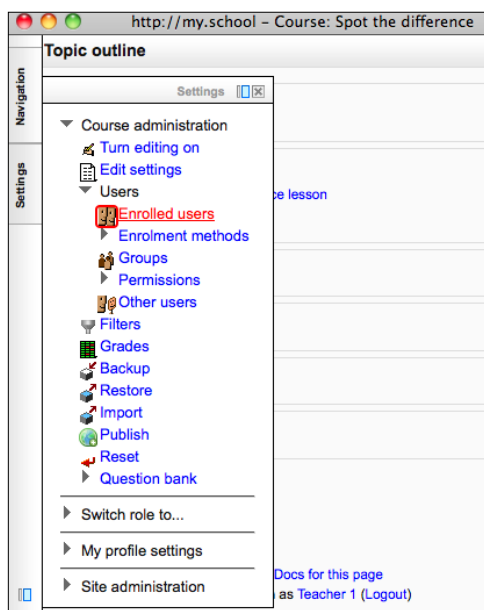
## Recording and tracking progress

While using Moodle to supplement your face-to-face teaching, you may wish to record how your class is progressing, and where problems lie. And it will be useful to view reports on their activity. Read on.

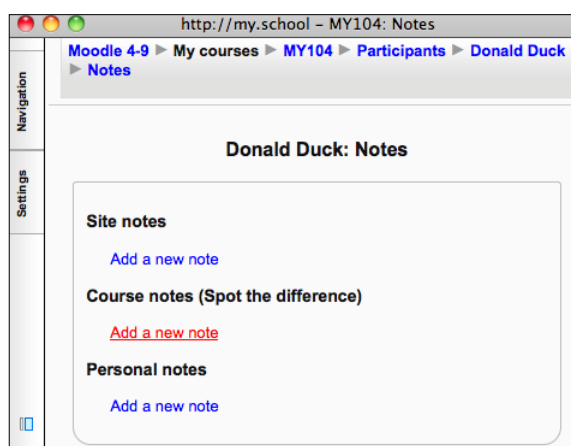
### Time for action – making notes

To create a note about a student in Moodle 2 (similar functionality is available in Moodle 1.9):

1. Go to a course containing students, for example **MY104**. In the **Settings** side block, choose **Course administration** | **Users** | **Enrolled Users**.



2. On the **Enrolled users** page click on a user image on the left of the table, for example the one next to **Donald Duck**.
3. Now, in the **Navigation** side-block go to **My courses | MY104 | Participants | Donald Duck | Notes**. Follow the link.
4. You will come to a page with three headings. **Site notes** are visible to all teachers in the site (strictly speaking users with the **notes:view** capability). **Course notes** are visible to all teachers in a course, while **Personal notes** are only visible to you. Click the link **Add a new note** under **Course notes**.



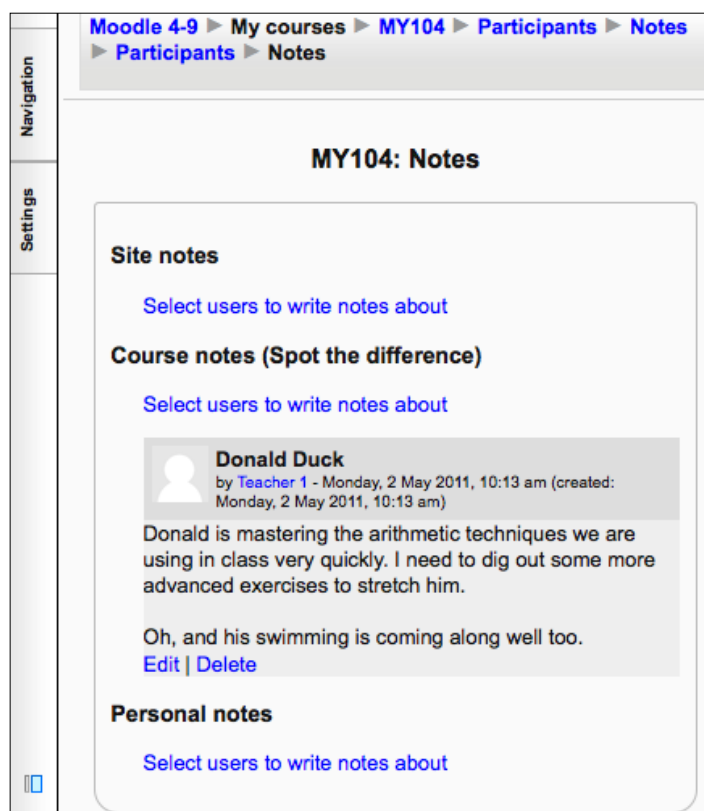
5. You will see a simple form under the heading **Donald Duck**. Enter your notes in the text area labeled **Content**, and press the **Save changes** button when you have finished.

## ***What just happened?***

We returned to our **Spot the difference** course (**MY104**), which was the course where we imported and enrolled users. We found the table of course participants, and then navigated to a simple form for adding notes. We added a **Course note**, only visible to teachers in this course.

If you can't see the links to the Notes functionality, ask your friendly IT support person to check whether they are enabled under **Advanced features** in the **Administration** interface.

As you can see in the following screenshot, the date and time that you create the note are recorded for you. Like much of the content in Moodle you have the ability to edit and delete notes at will. And when you edit a note you can change its visibility, for example from a **Personal note** to a Course note.



We've found a simple way to record thoughts about the progress of individuals in our class, and potentially share them privately with fellow teachers. What if we wish to communicate online with parents, and more widely with the teaching community?

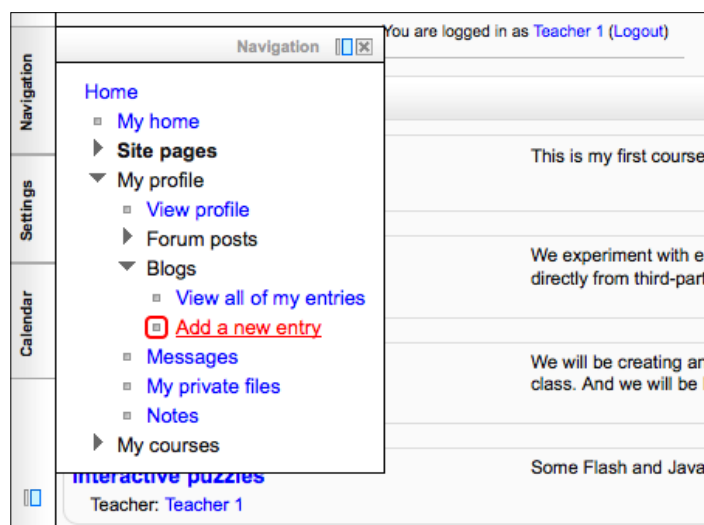
## Blogging

Blogging may sound scary, however as you get into it, you will hopefully find it a liberating experience. Moodle has the facilities for you (and your class) to create personal blogs, and to display content from an external blog. We will start off with the former option.

### Time for action – writing a blog post

These are the steps to follow to create your first blog post in Moodle:

1. Log in to your Moodle system. In the **Navigation** side-block go to **My profile, Blogs** and follow the link to **Add a new entry**.



2. You will see a form with the **Entry title** and a rich-editor for the **Blog entry body**. I called my post **Reflections on Moodle**, and I started jotting down some thoughts about writing this book.
3. There is a drop-down menu labeled **Publish to**. If your site is configured to allow world visibility there will be three options, **Yourself (draft)**, **Anyone on this site** (the default) and **Anyone in the world**. Choose one of the latter two options, as shown in the next screenshot.



Note, you will only be able to publish to **Anyone in the world** if your IT support or system administrator has enabled it. In the **Settings** side-block go to **Site administration | Advanced features**, and scroll down to **Blog visibility**.

Moodle 4.9 ► My profile ► Blogs ► Add a new entry

**General**

Entry title\* Reflections on Moodle

Blog entry body\*

Font family Font size Paragraph

**B I U ABC X<sub>2</sub> X<sup>2</sup>**

Through writing this book, I've learnt more about the wealth of features that Moodle offers to teachers and learners to help in their blended and distance education.

I hope that I've shown that Moodle is a great platform for experimenting, so that you can try new ways of teaching and your class can discover different ways of learning.

Path: p

HTML format

Attachment Add... Maximum size for new files: 128MB

No files attached

Publish to ?

Anyone on this site

Yourself (draft)

Anyone on this site

Anyone in the world

**Tags**

Tags Official tags (Manage official tags)

None

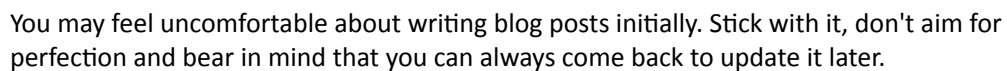
Other tags (enter tags separated by commas)

reflection, Moodle

4. Below the section labeled **General**, is one labeled **Tags**. Under **Other tags** you may wish to enter some tags or keywords, separated by commas. I put **reflection, Moodle**.
5. We can take tags much further. Visit the **Wordle** site created by Jonathan Feinberg. You will find it at, <http://www.wordle.net/create>. Enter some text that is relevant to your post. For example, you can copy the main body of the text available here: [http://docs.moodle.org/en/About\\_Moodle](http://docs.moodle.org/en/About_Moodle). Paste it into the form in Wordle, and press **Go**.

- Wordle uses a Java applet to display tag clouds. You will need Java installed on your computer, and enabled in your Web browser.

We created a blog entry. This consists of a title, body, and any number of keywords or tags. And we created a **Wordle** tag cloud as a fun way to illustrate our blog post.





To view the blog of another teacher in your Moodle system, in the **Navigation** side-block choose **My courses** | a course, for example **MY105** | **Participants**. Choose a person from the list, for example **Teacher 1** (<http://my.school/moodle2/user/view.php?id=3&course=7>). Then, in the **Navigation** block choose **My courses** | **MY105** | **Participants** | **Teacher 1** | **Blogs** | **View all entries by Teacher 1** (<http://my.school/moodle2/blog/index.php?userid=3>). You will see all the blog entries for the user, with the most recent first.

You can also encourage your students to blog and set blogging as a class exercise. This is a great way of encouraging creativity and self-expression. Naturally, this depends on the maturity of your class.

## Tags

When we save the blog post above, the tags are turned into links of the form <http://my.school/moodle2/tag/?tag=reflection>. Follow the link and you will be taken to a page as the one shown in the following screenshot. This lists any courses, blog entries and users tagged **reflection**. There is a link labeled **Add "reflection" to my interests**. Click on this and your user profile will be listed on the page for the reflection tag.



Tags can be useful tools for you and your class. They appear widely across the web, and allow people to collaborate to organize information. Tags should not be random. A useful tip is to observe what tags others are using, and reuse the ones that seem most appropriate, adding new ones sparingly.

## Have a go hero

Challenge 2. We found out how to add **tags** to things such as blog posts and our user profile. Set an exercise for your class where they create a description in their use profile, add a photograph and tags describing themselves. Explore the use of tags on another site such as **Flickr** or **Twitter**.

>> Suggestions: you and your class can tag your user profiles based on themes, for instance, favorite food, or membership of a sports team.

Explore **Clusters** around the tag `moodle` in Flickr, <http://flickr.com/photos/tags/moodle/clusters/>.

Try a search for the hash tag `#moodle` on Twitter, <http://twitter.com/search/%23moodle>.

## Gradebook

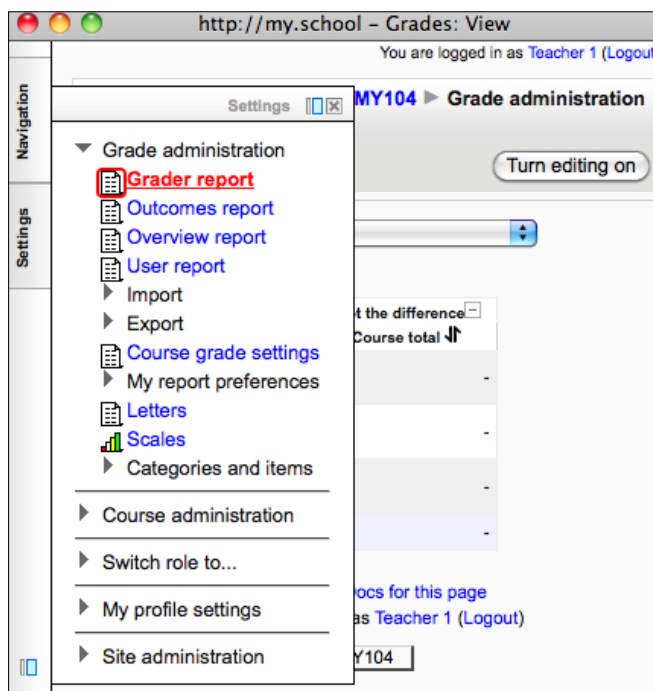
When we created the quiz and lesson activities way back in the first chapters, we needed to assign a grade to the correct and incorrect responses. The grades that the students achieve during these and other activities finish up in the **Gradebook**.

## Time for action – viewing grades

To view the grades for a course in Moodle 2 follow this procedure (the process is similar for Moodle 1.9):

1. Login to Moodle as a teacher and choose one of the courses we created previously, for example **MY104** (Spot the difference).
2. In the **Settings** side-block, under **Course administration**, choose **Grades**.
3. The system will take you to the **Grader report**.
4. You will see the **Spot the difference** lesson activity and the list of three students in the report. Next to the student names are icons, for example to view **Grades for Donald Duck**.
5. Under the **Course total** column there are icons to **Edit grade** for a student. Click on the icon and you will go to a form in which you can manually override a grade and enter **Feedback**.

6. In the **Settings** block, under **Grade administration** there are also **Outcomes**, **Overview**, and **User reports**, as you can see in the next screenshot:



### ***What just happened?***

We took a tour through the **Grader report** for one of our example courses and found one way to manually alter a grade. When your class has participated in some activities it will inevitably be easier to explore the results and reports.

Grades can be exported in a variety of spreadsheet and XML formats, via the **Settings** block, **Grade administration** | **Export**. And you can create manual grades under **Grade administration** | **Categories and items** | **Simple view**. Just press **Add grade item**.

This concludes the exploration of Moodle software features for the book. We'll finish by discussing some features of the ecosystem that has grown up around Moodle and that is integral to the software.

## Community

**Moodle** is a large, open source project. As such, a big community of interested people has grown up, centered around the main site at <http://moodle.org>. The community is diverse and consists of teachers, trainers and learning technologists, as well as developers. Moodle powers the main site, and the principle is that everyone is learning about Moodle by participating in courses such as **Using Moodle**, and collaboratively creating the software.

Inevitably as you explore and use Moodle you will have queries. Feel free to join the community for free at <http://moodle.org/community>, browse the forums and the documentation at <http://docs.moodle.org>. If you can't find an answer, pick the most suitable forum and pose your question. Keep in mind that people are generally helping others in their free time. Be patient and courteous.

In time you will become more confident, and you will be able to help others.

### Have a go hero

Challenge 3. Throughout this book we have concentrated on teaching with Moodle. However, as touched on in the **Community** section, Moodle can be used for staff development and community building. Try creating a course for you and your colleagues. You could use it for planning, sharing ideas, socializing, and so on.

>> Suggestions: Browse the courses on the Moodle community site, <http://moodle.org/course/> for ideas. **Moodle Lounge** is interesting in that it is primarily a social hub. You can use a topics-based course, with activities such as forums and a Wiki.

## Contributed plugins

At a number of points in this book we have installed and used third-party contributed plugins, such as the **Book** module (by Petr Škoda), **Game** module (by Vasilis Daloukas), and the **Calculated Objects** question type (by the author). There are many more plugins available, via <http://moodle.org/modules>. These can all enhance teaching and learning, and increase your productivity.

A word of caution is in order. Consider the long-term maintainability of your courses. Will the plugins and themes be updated for new versions of Moodle? You may be able to gauge the viability of a plugin by searching Moodle.org and the web. View activity in the code repository which may be at <https://github.com/search?q=moodle&type=Repositories>, <http://cvs.moodle.org/contrib/>, or elsewhere. Also look in the bug or issue tracker (<http://tracker.moodle.org/browse/CONTRIB>) and in the Moodle forums.

## Pop quiz

Have a go at these questions to test your knowledge of the topics in this chapter. Watch out! There may be more than one correct answer.

1. Which is the minimum role that you need in order to backup and restore Moodle 2 courses including user data?
  - a. Editing teacher
  - b. Manager
  - c. Administrator
2. By default, who can create blog posts in Moodle?
  - a. Guests
  - b. Teachers
  - c. Students
  - d. Administrators

## Summary

We covered a lot of ground in this chapter to do with some administrative, reporting, and communication functions of Moodle.

Specifically, we dealt with:

- ◆ Backing up a course—including the permissions required to backup user data
- ◆ Restoring a course in Moodle
- ◆ Creating teacher notes
- ◆ Writing a blog and tagging
- ◆ Gradebook
- ◆ Community interaction

We also discussed backing up and restoring single activities within a course, Wordle tag-clouds and contributed plugins.

This concludes our journey for this book. We have come a long way from our alphabet quiz in *Chapter 1, Getting Started*. Your journey with Moodle will continue, as you learn and explore the rich set of tools it provides for teaching and learning.

Happy Moodling!





# Accessibility for Online Teaching

*Throughout this book we have explored the use of Moodle to create learning resources and activities to help you in your teaching. You may find it odd that there is an appendix on accessibility. What do we mean by web accessibility? What does this have to do with my teaching? Why should I be concerned with this? These may be some of the questions you have at this point.*

In this Appendix, we will cover various topics:

- ◆ What is Web accessibility?
- ◆ Know the law—reasonable adjustments (UK, USA and international).
- ◆ Why accessibility for teachers and course designers?
- ◆ Principles—perceivable, operable, usable and robust (POUR).
- ◆ Accessibility by example.
- ◆ Sources of further information.

So let's press on...

## What is accessibility?

The Web Accessibility Initiative at the World Wide Web Consortium, say in their introduction:

*"Web accessibility means that people with disabilities can use the Web. More specifically, Web accessibility means that people with disabilities can perceive, understand, navigate, and interact with the Web, and that they can contribute to the Web."*— <http://w3.org/WAI/intro/accessibility.php>

The previous quote may suggest that making websites accessible to those with disabilities is exclusively the domain of web developers, designers, and the like—the people who create websites. If we look at a definition on Wikipedia though, we see two further details. Wikipedia refers to making websites *"usable by people of all abilities"*, and editing is included as part of the process of making a website accessible.

*"Web accessibility refers to the inclusive practice of making websites usable by people of all abilities and disabilities. When sites are correctly designed, developed and edited, all users can have equal access to information and functionality."*— [http://en.wikipedia.org/wiki/Web\\_accessibility](http://en.wikipedia.org/wiki/Web_accessibility) (accessed 23 April 2011)

The right to equal access to education including online education is enshrined in various national laws. Section 508 is an amendment of the Rehabilitation Act in the United States of America (<http://section508.gov>), which came into force in 1998. And in the United Kingdom there is the Special Educational Needs and Disability Act 2001. In contrast to Section 508 which is prescriptive, SENDA talks about making reasonable adjustments so as not to discriminate against someone who is disabled ([http://www.abilitynet.org.uk/edu\\_senda](http://www.abilitynet.org.uk/edu_senda)). This is a fuzzy term, which recognizes for example that a primary school may not have the same resources to accommodate a variety of needs as a large university or training body. However, this does not make a primary school exempt, and as we'll see there are many practical steps you can take.

Many other countries have similar legislation. See for example, Web Accessibility in Mind's world laws pages—<http://webaim.org/articles/laws/world/> and the Web Accessibility Initiative's policies page—<http://w3.org/WAI/Policy/>.



At an international level UNESCO has this to say:

*"The right to education is universal and must extend to all children, youth, and adults with disabilities. This right is enshrined in the Convention on the Rights of the Child (1989) and addressed in several significant, internationally approved declarations, including the World Declaration for Education for All (1990)..."—  
[http://www.unesco.org/education/efa/know\\_sharing/flagship\\_initiatives/disability\\_last\\_version.shtml](http://www.unesco.org/education/efa/know_sharing/flagship_initiatives/disability_last_version.shtml) (accessed 21 May 2011)*

In this Appendix, we will be referring to *assistive technologies*, which are software or equipment that help someone with a disability achieve something. A walking stick or glasses can be considered assistive technology, as can specialist computer keyboards, and software like a screen reader.

## Why accessibility for teachers?

So why does this affect me? You may say that no one in your class is blind or in a wheelchair. To answer this, consider that:

- ◆ Disabilities including dyslexia, color blindness, learning difficulties, and a stammer can affect a person's interaction with online resources, and with other people in an online context.
- ◆ It's easier and less time-consuming to build in accessibility from the start, so developing an appreciation and knowledge at this stage can increase the longevity of the online resources and activities that you create.
- ◆ Inclusive web design is generally considered to make a more usable Web for everyone, not just those who consider themselves disabled.
- ◆ As a teacher, you are helping to develop the next generation of Web citizens including online editors, authors, computer programmers, and teachers. Learning to author accessible content can be considered part of your class' ICT development.

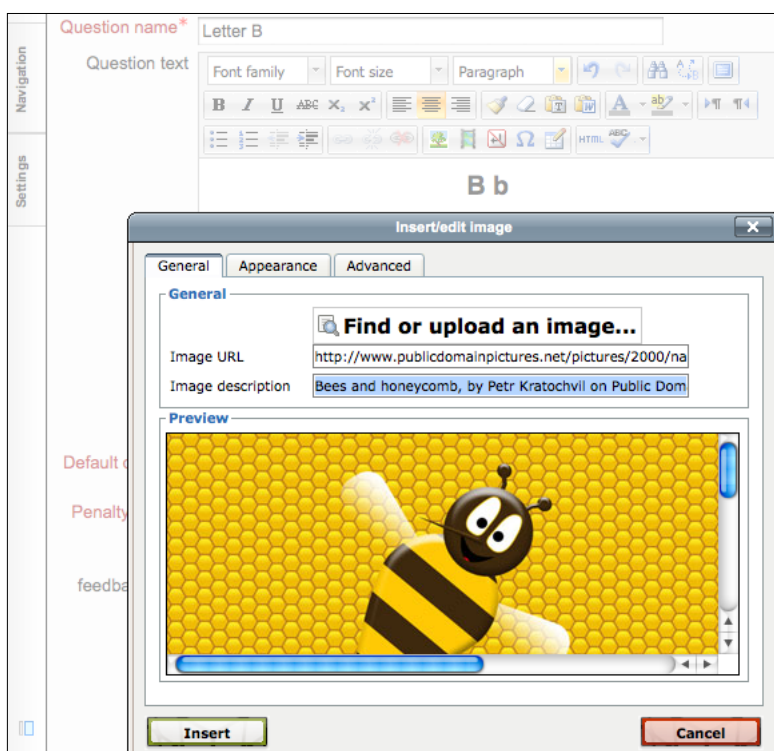
As we've already suggested, this last point is significant. Web accessibility is not just the domain of computer programmers and web designers. It involves those who write content, including learning resources and those who commission websites or learning content, and so on. On the modern Web anyone can publish content. It is the responsibility of us all.

## Guidance for teachers and online authors

The cornerstone of the Web Content Accessibility Guidelines version 2 (WCAG20) developed by the World Wide Web Consortium (<http://w3.org/TR/wcag20>) is the acronym **POUR**. This reminds us that we should make resources on the web **perceivable, operable, usable** and **robust**. What does this mean in the context of our practice as authors of online content and activities? A range of examples and techniques follow. Note that this is not designed to be an exhaustive set of guidelines. There are links at the end to further readings.

### Perceivable

The content we create should be perceivable in a variety of ways. This includes providing so-called alternative text for the images that we embed in most contexts. In the following screenshot, we go right back to the alphabet quiz we created in *Chapter 1, Getting Started*. When you insert an image using the rich editor, you are required to include an **Image description** below the **Image URL**. In this example, I have entered **Bees and honeycomb, by Petr Kratochvil on Public Domain Pictures.net**. This describes what the image is, or what purpose the image serves in the context of our quiz, as well as attributing the creator of the image.



The HTML markup generated when we insert an image contains our description in the so-called `alt` attribute; hence, we talk about **ALT** (alternative) text.

```

```

Note that when an image is the only thing inside a link, the ALT text is doubly important and should describe where the link goes.

The alternative text is particularly useful for those who are blind or have low vision. It can be read by a **screen reader**, software that converts text on the screen to synthesized speech or Braille, which can be output to a Braille display.

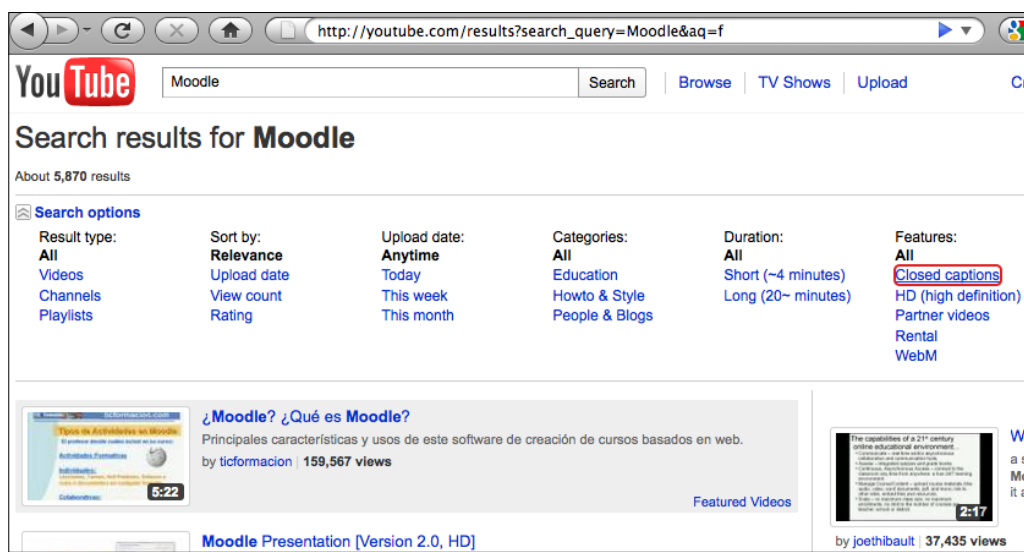
Alternative text is also useful for mobile browsers, who may choose not to download images to save on bandwidth.

In some contexts, for example in a picture recognition quiz, care must be taken to not "*give the game away*". You could use some fairly generic text such as **What's in the picture?** to keep this as a valuable exercise for the majority of your class. You can go on to create an **equivalent learning experience** for low vision students, for example, a sound recognition quiz (<http://freesound.org>). Of course, a sound-recognition exercise would be interesting for many in your class, not just those with dyslexia or low vision!

In general, text should be used in place of images of text. It is acceptable to use images of text for things such as logos, where the way that the text is presented is important in that context. So, for example, you might expect images of text in classes on art, product design, marketing, and typography.

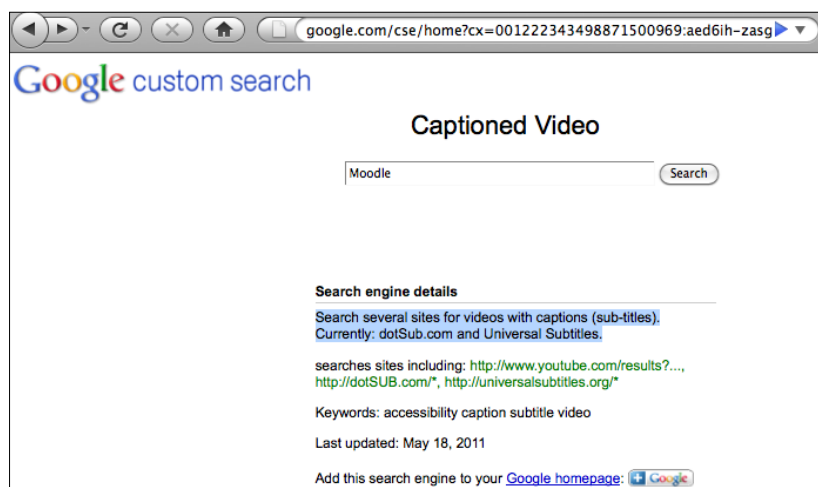
You should provide captions for pre-recorded video. This can be a challenge and is potentially time consuming in the context of a busy teaching schedule. However, there are various sources of help. It is possible to search some video sites such as **YouTube** specifically for content that has been captioned. For example, here is a search for videos on Moodle that have captions:

[http://youtube.com/results?search\\_type=videos&closed\\_captions=1&search\\_query=Moodle](http://youtube.com/results?search_type=videos&closed_captions=1&search_query=Moodle)



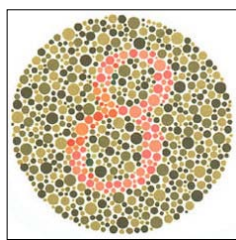
There are also websites that help to crowd-source the production of captions, notably **dotSub** (<http://dotsub.com>) and **Universal Subtitles** (<http://universalsubtitles.org>). Here is a **Google Custom Search** that will help you search several sites for captioned video:

<http://google.com/cse?cx=001222343498871500969:aed6ih-zasg>



The other ways to make your learning resources perceivable by the widest possible audience are to ensure there is sufficient color-contrast and that information is not conveyed by color alone. **Red-green** and **blue-yellow** are two of the more common forms of color blindness, so these are obvious combinations to avoid. And, there are tools available to test contrast against the thresholds specified in the WCAG20 (<http://paciellogroup.com/resources/contrast-analyser.html>). An example of conveying information by color and other means is to color an error message red, and to also prefix it with the word **error**.

We finish the section with **Ishihara plate 02** by **Vasile Tomoiagă** and hosted on Flickr released under a Creative Commons Attribution License, <http://flickr.com/photos/tomoiaga/3201541450/>.



What number can you see?

## Operable

What does it mean to us as authors to make the elements of a web page **operable**?

When adding hyperlinks to text, try to make the purpose of a link obvious when the link is taken out of context. An example of poor link-text is **click here**,

```
<a href="http://moodle.org"> Click here </a> to visit Moodle.org
```

Better link text would be the following,

```
<a href="http://moodle.org"> Visit the Moodle web site </a>
```

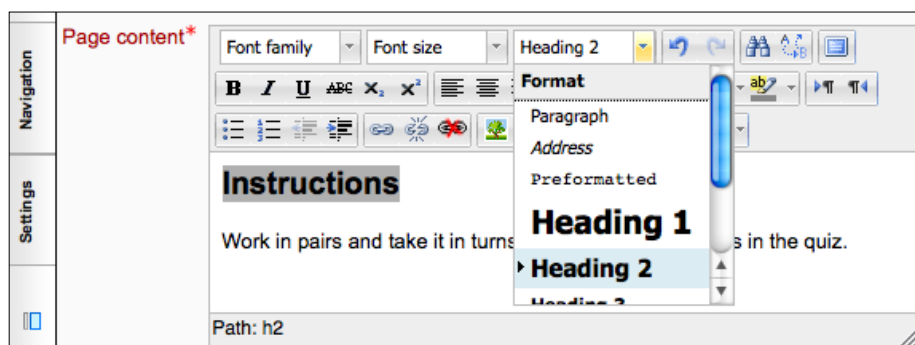
And, use the appropriate semantics provided by HTML to mark up items such as headings. Moodle uses a system of themes to style text, so your aim when writing text is to indicate structure. A poor way to indicate a heading is:

```
<b>Instructions</b>
```

An example of correct markup is:

```
<h2>Instructions</h2>
```

The following screenshot shows how to use the rich-editor in Moodle 2 to mark some text as a level 2 heading.



Appropriate link-text and correctly marked headings are particularly important as some assistive technologies provide shortcuts to allow the user to skip between items like links and headings on a page.

Note that though we have discussed headings in the context of web pages, they are equally useful in, for example, word processing documents.

We've made our resources **perceivable** and **operable**. What's next?

## Usable

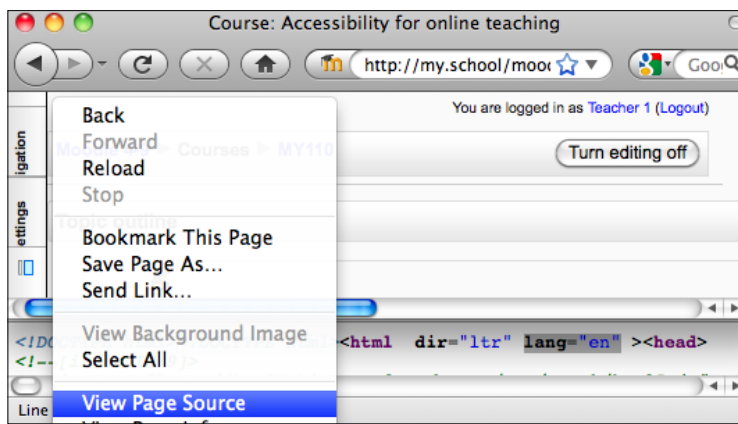
We've all at some point read a book or article that used obscure or confusing language, so that you had to work harder to understand what the author was trying to say. This is an example of poor **usability** and is something that we can all empathize with.

There are some general points to consider:

- ◆ Explain unusual words and abbreviations
- ◆ Write to the reading level of your class
- ◆ Provide help with pronunciation

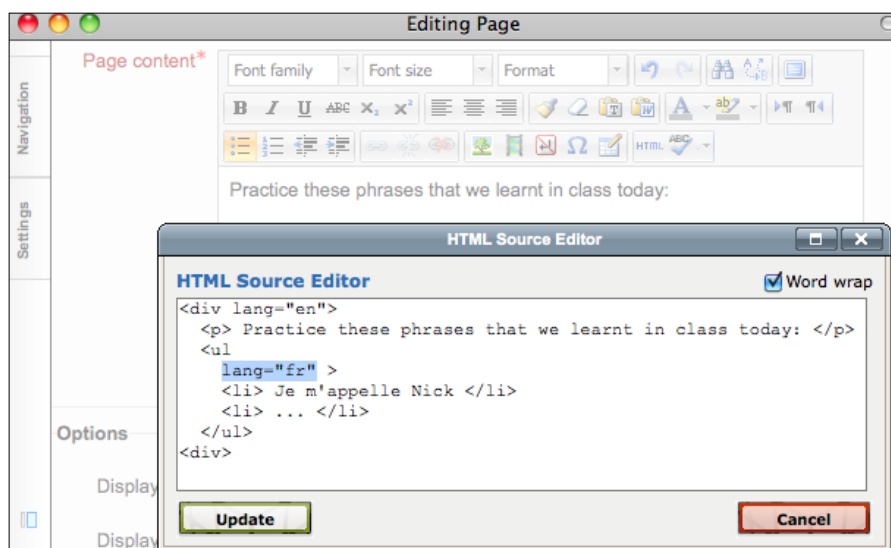
Moodle's **Glossary** module is a useful tool to help you and your class to make resources usable—we used Glossary in *Chapter 6, Fun Games*. There are third-party tools like the **Pop up Dictionary filter** by Patrick Thibaudeau (<http://moodle.org/mod/data/view.php?rid=1751>), as used in *Chapter 8, Stories Revisited*. And **SimpleSpeak filter** (<http://moodle.org/mod/data/view.php?rid=4778>) by the author, which we employed in *Chapter 1, Getting Started*.

In the context of foreign language teaching, there are specific techniques to note. View the **HTML source** of a page in Moodle as shown in the following screenshot:



You will see HTML markup like `<html lang="en">`, which indicates that in this case the primary language of the page is English. Other language codes include `fr` for French, `de` for German and `es` for Spanish. A full list of the **ISO-639-1** two-letter codes is available on sites such as Wikipedia ([http://en.wikipedia.org/wiki/List\\_of\\_ISO\\_639-1\\_codes](http://en.wikipedia.org/wiki/List_of_ISO_639-1_codes)).

When you mix content in different languages on the same page you should indicate the **change** of language. In the following example, the enclosing HTML element is in English `<div lang=en>`, while the inner list contains French text `<ul lang=fr>`.



There may be specialist plugins on your Moodle site to help markup text in different languages. However, you can always press the **Edit HTML Source** button in the rich-editor to manually add the `lang` attributes, like so:

```
<html lang="en">
```

The page header...

```
<p> Practice these phrases that we learnt in class today: </p>
```

```
<ul
```

```
  lang="fr" >
```

```
  >
```

```
    <li> Je m'appelle Nick </li>
```

```
    <li> ... </li>
```

```
</ul>
```

The page footer...

```
</html>
```

When changes in the language of text are indicated, tools such as screen readers have the opportunity to correctly pronounce the text. Try **Google Translate** to hear speech synthesizers for different languages in action, for example,  
<http://translate.google.com/#en|fr|I+am+called+Rosie>.



You may also wish to try **NVDA**, a free and open source screen reader for Windows,  
<http://nvda-project.org>.

As you can see, improving the usability of a resource is an activity that everyone can benefit from.

## Robust

Considerations of robustness are principally the domain of Web developers. Robustness is in part about using the semantics available in HTML and associated standards to indicate items on a web page such as links, buttons, sliders, and so on, in a way that can be understood by assistive technologies such as screen readers.

A number of improvements have been made to the accessibility of the underlying Moodle software in recent years. Find out more on the Moodle Docs Wiki via,  
<http://docs.moodle.org/en/Category:Accessibility>.

## Useful links

Here are a number of resources that you may find useful:

- ◆ Dive Into Accessibility, 30 days to a more accessible website, by Mark Pilgrim (2002),  
<http://diveintoaccessibility.org/>.
- ◆ Web Accessibility Initiative, at the World Wide Web Consortium (W3C),  
<http://w3.org/WAI/>.
- ◆ Web Content Accessibility Guidelines 2.0, B Caldwell, M Cooper, LG Reid G Vanderheiden, Editors, W3C Recommendation, 11 December 2008,  
<http://w3.org/TR/2008/REC-WCAG20-20081211/>.
- ◆ Make your teaching inclusive, by The Open University, CARS project funded by HEFCE (2006), <http://www.open.ac.uk/inclusiveteaching/>.
- ◆ Accessibility of eLearning, by The Open University, a 15 hour module released under a Creative Commons Attribution Share-Alike Non-Commercial License on OpenLearn, [http://openlearn.open.ac.uk/H807\\_1](http://openlearn.open.ac.uk/H807_1).
- ◆ Moodle accessibility forum—a useful place to seek advice and discuss issues,  
<http://moodle.org/mod/forum/view.php?id=5752>.
- ◆ Web Accessibility in Mind community, hosted by Utah State University,  
<http://webaim.org/community/>.
- ◆ Further accessibility resources to accompany the book,  
<http://freear.org.uk/moodle/accessibility>.



Packt has published *Moodle 1.9 for Teaching Special Education Children (5-10): Beginner's Guide* by Vanesa S. Olsen. Vanesa explores many imaginative ways to create activities that are inclusive, and that use and develop all the senses of your pupils. Find out more at <http://packtpub.com/moodle-1-9-for-teaching-special-education-children-5-10/book>.

## Final words

In this Appendix, we've explored the relevance of accessibility to the role of the online teacher.

Specifically, we covered:

- ◆ The meaning of Web accessibility, that is, making online resources accessible to those with disabilities
- ◆ The legal situation regarding the right to education and accessibility in several jurisdictions around the world
- ◆ Why you should be interested in accessibility as an online teacher
- ◆ How to make resources **perceivable**—alternative text for images, captions for video, considerations of color contrast, and the appropriate use of color
- ◆ Making resources **operable**, including the use of semantic markup for headings and the appropriate text for links
- ◆ How to improve the **usability** of online resources, by choosing language suitable to the audience, explaining new words and indicating changes in language, especially in foreign language teaching
- ◆ **Robustness**—we found that this is mostly the concern of programmers and designers

During our discussions we have demonstrated that:

- ◆ There are practical steps that you can take as a teacher
- ◆ Enhancements to the accessibility of your teaching resources will often benefit your whole class
- ◆ There is common ground between learning to author accessible content and you and your class developing effective ICT and computer skills

# B

## Pop Quiz Answers

### Chapter 1

#### Getting Started

---

|   |  |
|---|--|
| 1 | c  |
| 2 | b  |
| 3 | b and d  |
| 4 | b (This makes a task simpler for you, or makes a potentially insecure operation, safe. For instance, adding JavaScript)) |

---

### Chapter 2

#### Basic Math in Moodle

---

|   |  |
|---|--|
| 1 | c and d (as of 2011)   |
| 2 | b and c  |
| 3 | Numerical and Calculated (Calculated Objects is a third-party plug-in, not part of the core) |

---

## Chapter 3

### Telling Stories

- 
- |   |   |
|---|---|
| 1 | a and b   |
| 2 | c (press the button <b>Reset template</b> )   |
| 3 | b, encoding audio files, including as MP3s (Audacity is the desktop software which records and edits) |
- 

## Chapter 4

### Spot the Difference

- 
- |   |   |
|---|---|
| 1 | a and c   |
| 2 | b   |
| 3 | b, it describes the benefit of vector graphics such as SVG over raster/bitmap formats like JPEG for line art and other types of image. (Points a and b are good for other reasons!) |
| 4 | c (Note, it would be insecure to allow all pupils to share a password!)   |
- 

## Chapter 5

### Setting Homework

- 
- |   |   |
|---|---|
| 1 | a                                       |
| 2 | b and c                                 |
| 3 | c (using built-in Moodle functionality) |
-

## Chapter 6

### Fun Games

---

|   |         |
|---|---------|
| 1 | b and e |
| 2 | b       |
| 3 | c       |

---

## Chapter 7

### Interactive Puzzles

---

|   |         |
|---|---------|
| 1 | b       |
| 2 | c       |
| 3 | a and c |

---

## Chapter 8

### Stories Revisited

---

|   |   |
|---|---|
| 1 | b or e, and c (most Project Gutenberg books have expired copyright, but occasionally authors or rights-owners submit their works. And Project Gutenberg relies on volunteer effort) |
| 2 | a, it generates a table of contents   |

---

## Chapter 9

### Embedding the Web

- 
- |   |   |
|---|---|
| 1 | c, (Java and JavaScript are sometimes confused, but are completely unrelated)                               |
| 2 | a and c. (The embed code remains as visible to the reader—they can just click View source in their browser) |
- 

## Chapter 10

### Administration

- 
- |   |   |
|---|---|
| 1 | b   |
| 2 | b, c, and d (All are authenticated users) |
-

# Index

## Symbols

`<iframe>` HTML element 46  
`<param>` tags 195, 207  
`<wordsearch>` element 188  
.zip file extension 270

## A

**accessibility**  
about 290  
for teachers 291  
**Add an activity drop-down menu** 18  
**Add a new course button** 14  
**Add a question page button** 97  
**Add a resource menu** 18  
**Adobe® Flash® Player plug-in**  
URL 185  
**ALT (alternative) text** 293  
**ALTER TABLE statement** 165  
**Ant**  
URL 161  
**assignments**  
about 19  
creating 146, 147  
**Audacity**  
about 78  
installing 78  
installing, on Mac OS X 78  
**audio**  
recording 81, 83  
testing 80  
test recording, performing 80

## audio file

embedding 88  
files, uploading 87  
integrating, with Moodle 83  
multimedia filter, enabling 83  
uploading 84, 85

## B

**blogging**  
about 279  
blog post, writing 279-283  
tags 283  
**book**  
creating 214  
creating, steps 215-217  
custom styles, adding 219-223  
finding 210-212  
improving 219  
pictures, adding 223-225  
**book module**  
about 286  
activity module, installing 212-214  
installing 212  
**branch, lesson activity**  
adding 116, 117

## C

**Calculated Objects** 286  
**calculated objects question**  
creating 56-58  
**calculated objects question type**  
installing 55, 56

- calculated question**
  - creating 52-54
- Category field 14**
- chat 19**
- choice 19**
- Cloud**
  - URL 161
- cluster 95, 284**
- CMYK (cyan-magenta-yellow-K) 107**
- comment**
  - story, adding as 76, 77
- community 286**
- Confirmation and review 268**
- course**
  - restoring 272-276
- course backup**
  - about 265
  - creating 266-271
- Creative Commons 36**
- Creative Commons Attribution Share-Alike License**
  - URL 240
- cryptex**
  - adding 170
  - word search puzzle, adding 171, 172
- custom styles, book**
  - adding 219-222

**D**

- database 19**
- database activity**
  - creating 64-69
- database entry**
  - commenting on 76
- database fields**
  - planning 68
- dictionary**
  - installing 228
  - popup dictionary, installing 228, 229
  - service, integrating 230
- dictionary service**
  - integrating 230
  - using 230, 231
- docked Calendar block 12**
- dotSub 294**
- Draconian error handling 133**

- Drum Kit 2.0 240**
- Drum Kit 2.0 project 237**
- Duncan Keith's Word Search puzzle**
  - URL 186

## E

- embed filter, Scratch**
  - about 240
  - installing 240
  - using 241, 242
- end of branch 95**
- end of cluster 95**
- enrolment key**
  - configuring 118
  - configuring, in Moodle 2, steps 118, 119
  - password generator, URL 118
  - self-enrolment settings, steps 119, 120
  - users, uploading 120, 121
  - user upload, used for enrolling 120
  - using, steps 120
- entry**
  - adding, to stories database 73-75

## F

- Flash games**
  - URL 207
- flashvars (Flash variables) variable 204**
- Flash, word search puzzle**
  - embedding 193-195
- follow-on activity**
  - creating 136
  - forum, adding 136, 138
- forum 19**
- forum, follow-on activity**
  - adding 136
- Full name field 14**

## G

- game**
  - creating 166
  - snakes and ladders, adding 166-170
- Game module 286**
- Games module**
  - about 163
  - installing 163



**Geograph**

URL 174

**Github**

URL 163

**glossary**

about 19

creating 154

creating, steps 155, 156

**glossary auto-linking**

about 162

using 162

**glossary entries**

adding 158, 160

**Google Chrome Frame**

URL 256

**Google Custom Search 294****Google Translate**

Google TranslateURL 27

**graceful degradation 51****Gradebook**

grades, viewing 284, 285

**guidance**

for online authors 292-299

for teachers 292-299

**H****height attributes 188****hidden picture**

adding 174-176

questions, adding 177, 178

**hidden picture game**

adding 179-182

**HTML5 jigsaw**

about 256

creating 259

exploring 256-259

iframe, used 260, 261

**hue-saturation-luminance 107****Hyper-Text Markup Language (HTML) 26****I****iframe**

using, in HTML5 jigsaw 260

**image editor**

Inkscape, installing 102, 103

installing 101

**images**

adding, to lesson 109, 110

editing 103, 104

inserting 24

inserting, steps 25, 26

SVG image, editing 104-108

**information**

adding, to timeline XML file 138

**Inkscape**

installing 102, 103

**J****jigsaw puzzle**

about 199

jigsaw picture, uploading 199, 200

**L****label**

creating 39

resource, creating 39-42

**LAME**

installing 78

installing, for Audacity on Windows 78, 79

installing, on Mac OS X 79

**lesson 19****lesson activity**

adding 95, 96

branch, adding 116, 117

images, adding 109

question page adding, steps 97, 98

wrapping 116

**lesson module**

about 94, 95

cluster 95

end of branch 95

end of cluster 95

**M**

Massachusetts Institute of Technology. *See* MIT

**math questions**

creating 48-51

**MathsNet 197****mbz (Moodle backup Zip) 270**

**Michael Badger** 236  
**MIME type (Multi-purpose Internet Mail Extensions)** 195  
**MIT** 36, 126, 236  
**moddata** 87  
**Moodle**  
   about 10  
   default activities 19  
   exploring 11, 12, 13  
   game, creating 166  
   games 153  
   glossary, auto-linking 162, 163  
   glossary, creating 154  
   glossary entries, adding 158  
   hidden picture 174  
   hidden picture, questions 177  
   logging in 10  
   puzzle, creating 179  
   SimpleSpeak filter, using 29-31  
**Moodle course**  
   creating 14, 15  
   roles, exploring 16, 17  
**Moodle, default activities**  
   chats 19  
   choice 19  
   database 19  
   forum 19  
   glossary 19  
   lesson 19  
   quiz 19  
   SCORM/AICC 19  
   survey 19  
   wiki 19  
   workshop 19  
**MoodleDocs site**  
   URL 154  
**Moodle Lounge** 286  
**Mozilla Firefox 3.0 onwards**  
   URL 256  
**Multiple choice type** 98

## N

**Notepad++** 128  
**Notepad2**  
   about 128  
   installing 128-130

**Notepad2 text editor** 188  
**numbers typography** 36  
**numerical questions**  
   creating 46, 48

## O

**OCAL**  
   about 99  
   images, finding online 99, 100  
**online authors, guidance for** 292-299  
**online book**  
   completing 225  
   remaining parts, adding 225-228  
**online jigsaw**  
   creating 201  
   jigsaw class, exercise 206  
   jigsaw, preparing 201-204  
**online video tutorials** 249  
**online word search generator**  
   about 197  
   using 197  
**Open Clip Art Library.** *See* OCAL  
**open content**  
   searching for 36-38  
**Open Directory Project** 248  
**Opera 10 onwards**  
   URL 256  
**operable, guidance for online authors** 296  
**operable, guidance for teachers** 296

## P

**Patrick Thibaudeau** 209  
**perceivable, guidance for online authors** 292-295  
**perceivable, guidance for teachers** 292-295  
**PhET project** 249  
**pictures**  
   adding, to book 223-225  
**plugins** 286  
**POUR** 292  
**progress**  
   notes, making 276-278  
   recording 276  
   tracking 276  
**Project Gutenberg** 210  
**Public Domain** 100

**puzzle**  
  creating 179  
  hidden picture game, adding 179-182  
**puzzle XML**  
  uploading 190  
  XML file, uploading 191, 192

## Q

**question page**  
  adding, to lesson activity 97, 98  
**questions**  
  adding 113-115  
**quiz**  
  about 19  
  activity, creating 20-22  
  assignments 19  
  creating 18  
  incorporating, in teaching 32

## R

**Restore page** 269  
**robust, guidance for online authors** 299  
**robust, guidance for teachers** 299  
**Roles and permissions** 266  
**roles, Moodle course**  
  exploring 16, 17  
**RSS feed**  
  about 244  
  adding 244, 245  
  CBBC Newsround 248  
  CBBC Newsround Pictures 248  
  feeds 248, 249  
  improving 246  
  iSpot 249  
  World News for Schools (podcasts) 249  
  Yahoo! Pipes, using 246

## S

**Safari 3.0 onwards** 256  
**Scalable Vector Graphic (SVG)** 100  
**SCORM** 19  
**SCORM/AICC** 19  
**Scratch**  
  about 236  
  development software, downloading 249

  embed filter 240  
  projects, embedding 236-240  
**Scratch embed filter.** *See* **embed filter, Scratch**  
**screen reader** 293  
**SENDER** 290  
**Sharable Content Object Reference Model.** *See*  
  **SCORM**  
**Short name field** 14  
**SIMILE**  
  URL 127  
**SimpleSpeak filter** 297  
  installing 27, 28  
  troubleshooting 31, 32  
  using, in Moodle 29-31  
**sketchr/ Kendradeann**  
  URL 199  
**snakes and ladders**  
  adding 166-170  
**sound recorder**  
  Audacity, installing 78  
  Audacity, installing on Mac OS X 78  
  installing 77  
  LAME, installing 78  
  LAME, installing for Audacity 78, 79  
  LAME, installing on Mac OS X 79  
**spot-the-difference images**  
  finding 99  
  finding, on OCAL 99-101  
**Stephen Ostermiller** 203  
**stories database**  
  entry, adding 73-75  
**story**  
  adding, as comment 76, 77  
**students**  
  enrolling 118  
**survey** 19  
**SVG image**  
  editing 104-108

## T

**tags** 283  
**teachers, guidance for** 292-299  
**templates**  
  editing 71, 72  
**text and picture-based online book**  
  URL 227

**text editor**

- installing 127, 128
- Notepad++ 128
- Notepad2 128
- Notepad2, installing 128-130
- TextWrangler 128
- types 128

**text filter**

- installing 27
- SimpleSpeak filter, installing 27, 28

**TextWrangler 128****third-party book module 209****timeline data file**

- creating 131
- no element found, error 134
- not well-formed, error 133
- troubleshooting 133, 134
- undefined entity, error 134
- XML, creating 131-133

**Timeline Widget filter**

- installing 134, 135

**timeline XML file**

- assignment, creating 146, 147
- editing 139, 141
- embedding 142-145
- generating, dynamically 150, 151
- information, adding to 138
- integrating, with teaching 146

**Turn editing on button 17****U****Universal Subtitles 294****URLs 299****usable, guidance for online authors 297, 298, 299****usable, guidance for teachers 297-299****user upload**

- used, for enrolling 120

**V****Vasilis Daloukas**

- URL 153

**video**

- embedding 43
- embedding, in Moodle course 43-45

**virtual learning environment or learning man-**

agement system. *See* VLE / LMS

**VLE / LMS 10****W**

**Web accessibility.** *See* accessibility

**Web Content Accessibility Guidelines version 2 (WCAG20) 292****What-You-See-Is-What-You-Get (WYSIWYG) editor 22****wiki 19****Wikimedia Commons**

- URL 187

**Wikipedia**

- URL 145

**Wordle! 281****Wordle site 280****word search**

- about 186
- planning 186, 187
- XML, creating 188-190

**word search puzzle**

- adding 171, 172
- creating 193
- Flash, embedding 193-196

**workshop 19****World Wide Web Consortium (W3C) 101****X****XML parser 133****XML, word search**

- creating 188-190

**Y****Yahoo! Pipes**

- using 246-248



## **Thank you for buying Moodle 2.0 for Teaching 4-9 Year Olds Beginner's Guide**

### **About Packt Publishing**

Packt, pronounced 'packed', published its first book "*Mastering phpMyAdmin for Effective MySQL Management*" in April 2004 and subsequently continued to specialize in publishing highly focused books on specific technologies and solutions.

Our books and publications share the experiences of your fellow IT professionals in adapting and customizing today's systems, applications, and frameworks. Our solution based books give you the knowledge and power to customize the software and technologies you're using to get the job done. Packt books are more specific and less general than the IT books you have seen in the past. Our unique business model allows us to bring you more focused information, giving you more of what you need to know, and less of what you don't.

Packt is a modern, yet unique publishing company, which focuses on producing quality, cutting-edge books for communities of developers, administrators, and newbies alike. For more information, please visit our website: [www.packtpub.com](http://www.packtpub.com).

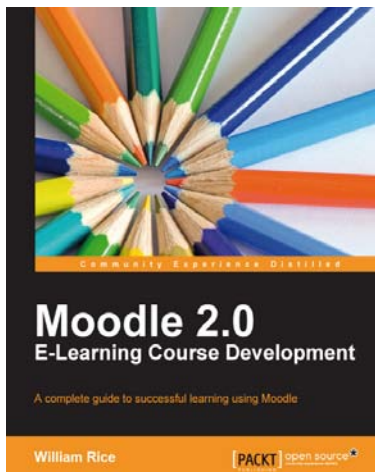
### **About Packt Open Source**

In 2010, Packt launched two new brands, Packt Open Source and Packt Enterprise, in order to continue its focus on specialization. This book is part of the Packt Open Source brand, home to books published on software built around Open Source licences, and offering information to anybody from advanced developers to budding web designers. The Open Source brand also runs Packt's Open Source Royalty Scheme, by which Packt gives a royalty to each Open Source project about whose software a book is sold.

### **Writing for Packt**

We welcome all inquiries from people who are interested in authoring. Book proposals should be sent to [author@packtpub.com](mailto:author@packtpub.com). If your book idea is still at an early stage and you would like to discuss it first before writing a formal book proposal, contact us; one of our commissioning editors will get in touch with you.

We're not just looking for published authors; if you have strong technical skills but no writing experience, our experienced editors can help you develop a writing career, or simply get some additional reward for your expertise.

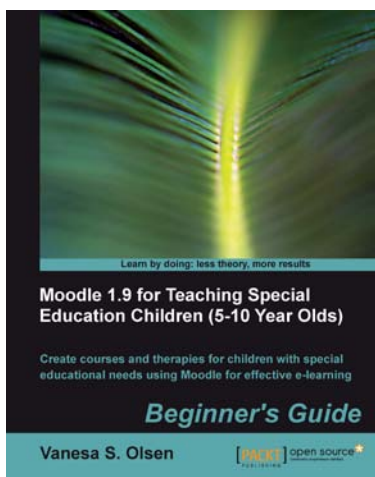


### **Moodle 2.0 E-Learning Course Development**

ISBN: 978-1-849515-26-9      Paperback: 344 pages

This book and eBook is a complete guide to successful learning and e-learning using Moodle

1. The new book and ebook edition of the best selling introduction to using Moodle for teaching and e-learning, updated for Moodle 2.0
2. Straightforward coverage of installing and using the Moodle system, suitable for newcomers as well as existing Moodle users who want to get a few tips
3. Configure site settings, set up the front page, create user accounts, and create courses



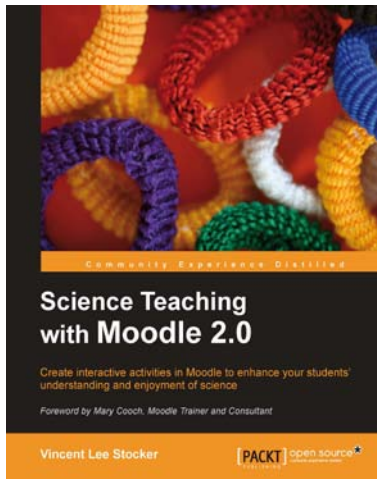
### **Moodle 1.9 for Teaching Special Education Children (5-10): Beginner's Guide**

ISBN: 978-1-849510-94-3      Paperback: 324 pages

Create courses and therapies for children with special educational needs using Moodle for effective e-learning

1. Focus on building rich and interactive activities to create least restrictive environments, your students will want to go on day after day
2. Learn to use netbooks, gamepads, touch screens, microphones, and Webcams to motivate children
3. Take advantage of all the new possibilities offered by the creation of rich User Experiences in your activities

Please check [www.PacktPub.com](http://www.PacktPub.com) for information on our titles



## **Science Teaching with Moodle 2.0**

ISBN: 978-1-849511-48-3      Paperback: 296 pages

Create interactive lessons and activities in Moodle to enhance your students' understanding and enjoyment of science

1. Follow a sample course to see how lessons, groups, and forums are created
2. Make your student's homework more exciting by enabling them to watch videos, participate in group discussions, and complete quizzes from home
3. Simplify the teaching of difficult scientific notation using animations
4. Monitor your students' progress using Gradebook



## **Alice 3 Cookbook**

ISBN: 978-1-849514-92-7      Paperback: 340 pages

79 recipes to harness the power of Alice 3 for teaching students to build attractive and interactive 3D scenes and videos

1. A cookbook for teachers who would like to demonstrate the basics of programming by creating marvelous animated scenes
2. Work with the new Alice 3 environment to add instances and inspect their properties, procedures, and functions
3. Develop interactive 3D scenes and videos with Alice 3
4. The book covers the new Sims module incorporated in Alice 3

Please check [www.PacktPub.com](http://www.PacktPub.com) for information on our titles

