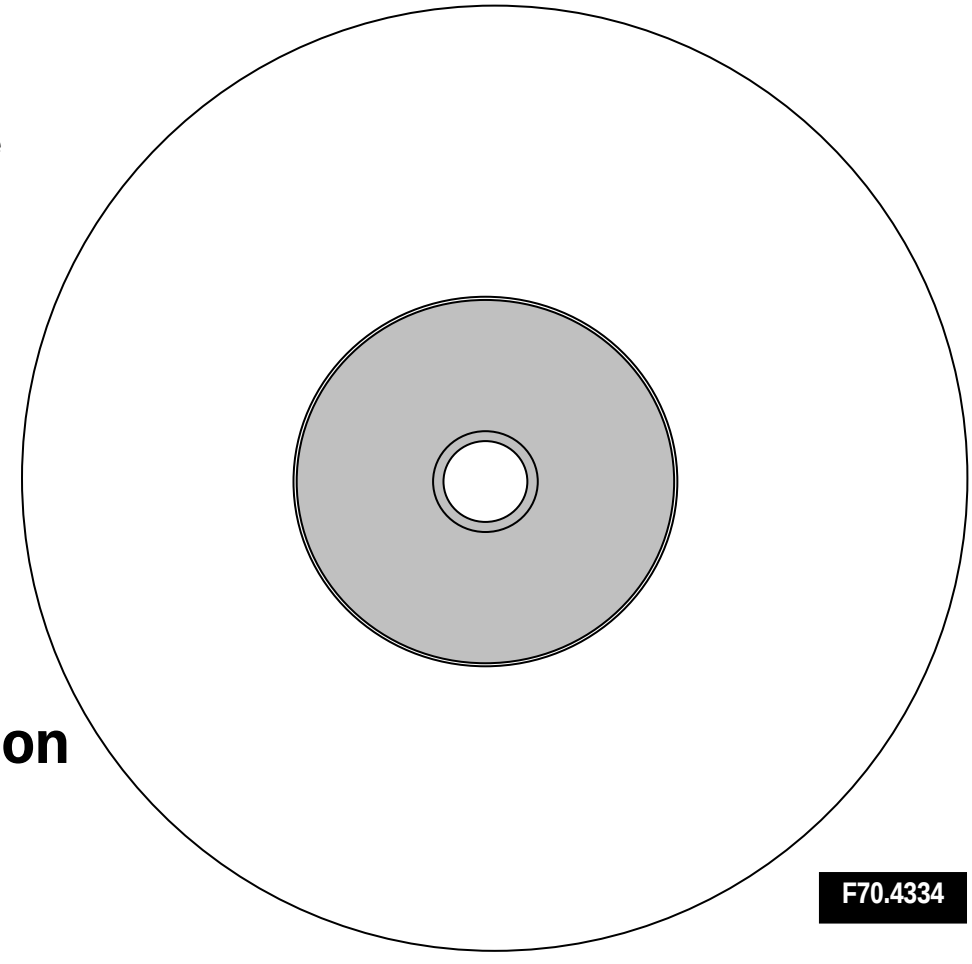
php on a Stick

This unique all-in-one tool allows students to learn about, develop and run their own PHP and SQLite projects straight from a USB memory stick!

The master files are provided on this CD-ROM. These files can be copied directly onto each student's memory sticks, although it is suggested that you copy them to your site's network beforehand.

For guidance on installing and using *PHP on a Stick*, read pages 2–9.



ZigZag Education
ict@zigzageducation.co.uk
zigzageducation.co.uk

Author: Daren Craddock

F70.4334

Contents

Thank You

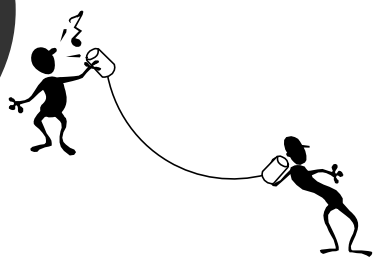
for choosing ZigZag Education!

Talk to Us!

Love it as it is?
Let the author and other teachers know what you think

Got a suggestion?
If your improvement leads to an update we will send you a new copy for free

Found a problem?
We will fix it and send you a free updated copy



We ♥ your feedback – let us know what you think using the feedback sheet on the next page
➥ ***£10 ZigZag Voucher for detailed & complete reviews***

For more resources go to **zigzag.at/more**
– where you can preview every page before you buy

ZigZag is a large community of over 6000 teachers & educationalists.
Review new titles or publish your own work

Fancy being involved?

Then register at...
publishmenow.co.uk
The Professional Publishing Community

Alternatively email new resource ideas directly to...
publishmenow@zigzageducation.co.uk

ZigZag Education
Unit 3, Greenway Business Centre
Doncaster Road
Bristol
BS10 5PY

0117 950 3199
Real Person

0117 959 1695
Fax

zigzageducation
.co.uk

Teacher Feedback Opportunity

£10 ZigZag Voucher for detailed & complete reviews!
Use for problems/areas for improvement/positive feedback

Resource ID & name	**4334** *PHP on a Stick*	**Your Name**	
School Name		**Position**	

Overall, what did you think about this resource? _____

_____

_____

_____

I particularly like this resource because… _____

_____

_____

How does it help you or your students? _____

_____

It is better than some other resources because… _____

_____

_____

What might you say to a colleague in a neighbouring school to persuade them to use this resource? _____

_____

_____

_____

How well does it match your specification (& which specification is this)? _____

_____

Other comments, suggestions for improvements, errors found (please give page numbers) etc. _____

_____

_____

_____

Resources **I would like published***:*

Resources **I might write***, or have written,* **for consideration for publication***:*

Fax: **0117 959 1695**
Email: **feedback@ZigZagEducation.co.uk**
Post: **ZigZag Education, Unit 3, Greenway Business Centre, Doncaster Road, Bristol BS10 5PY**

INTERNAL USE ONLY
Feedback logged: ✓☐
Complete & detailed: Y /N
If detailed, £10 sent: ✓☐

Terms and Conditions of Use

Terms and Conditions

Please note that the **Terms and Conditions** of this resource include point 5.8, which states:

"You acknowledge that you rely on your own skill and judgement in determining the suitability of the Goods for any particular purpose."

"We do not warrant: that any of the Goods are suitable for any particular purpose (e.g. any particular qualification), or the results that may be obtained from the use of any publication, or expected exam grades, or that we are affiliated with any educational institution, or that any publication is authorised by, associated with, sponsored by or endorsed by any educational institution."

Copyright Information

Disclaimer

It is the teacher's own responsibility to assess the suitability of publications for use in their school. ZigZag Education is not affiliated with MIT, DfES, Edexcel, OCR, AQA, WJEC or CCEA in any way nor is this publication authorised by, associated with, sponsored by or endorsed by these institutions unless explicitly stated on the front cover of this publication.

PHP on a Stick utilises a number of different applications. Besides the content that is unique to PHP on a Stick (including the interactive PHP and SQLite tutorials, the fully tested and working demonstrations, and the Sandbox tool), there are also a number of third party open-source applications (free to distribute) that have been modified to run directly from a USB memory stick and integrate with PHPonaStick.

These include: *Uniform Server, Apache, PHP, SQLite, EditArea, SQLiteManager, PHPMailer, PHP Zip Class, Firefox, FirefoxPortable Launcher, Simple Tree Menu (JS), Colour Picker (JS)*

When you pay for PHP on a Stick you are not paying for these free open source applications. Rather, you are paying for the interactive PHP and SQLite tutorials, the fully tested and working demonstrations, the Sandbox tool, and the time and effort that has gone into getting these all to work together in a single package.

All the third party applications used are free to redistribute, and the licences for each of these applications can be found in the provided electronic files (inside the *licences* folder). ZigZag Education and Daren Craddock accept no responsibility for any lack of function or loss of any kind (electronic, personal, etc.) as a result of using these third party applications. They are provided on an "as is" basis. In using PHPonaStick you automatically agree to this disclaimer.

Teacher's Introduction

What is *PHP on a Stick*?
PHP on a Stick is a unique tool for learning about PHP and SQLite, and will enable your students to build, develop and even run PHP projects straight from a USB memory device. Created by a successful and practising Head of Computing, *PHP on a Stick* enables students of all abilities to enjoy immediate success with their programming.

Everything you and your students need is included: comprehensive step by step tutorials with dozens of tested examples, fully working demonstration projects, a unique Sandbox tool for writing and running PHP programs, all in a single package that runs straight from any USB memory stick. No other software or Internet connection is required. *PHP on a Stick* has been tested on all versions of Windows[1] and works equally well on stand alone computers or on a school network.

Suitable for all exam boards at GCSE or A level, *PHP on a Stick* is the ***complete*** solution for teaching and learning programming. Ideal for all ICT and Computing courses.

Key Features and Benefits

- 32 complete tutorials on all aspects of programming and databases.
- 8 working demonstration projects with detailed explanations.
- Unique Sandbox tool to build and develop projects.

✓ Ready to go – all the software needed to run the application is included.
✓ Run PHP applications locally – no web server required.
✓ Easy to transfer work between school and home – ideal for project work.
✓ Builds confidence with immediate success and quick results.

Why PHP?
PHP is now an accepted programming language for all exam boards. It is easy to learn, and your students will soon be building their own exciting database driven projects. PHP programmers are in great demand for the development of commercial websites, so as well as being fully prepared for examinations and projects, your students will be gaining a vocationally relevant skill to add to their CVs and UCAS references.

How to use *PHP on a Stick*
The comprehensive tutorials on PHP programming and SQLite databases are structured progressively, and can be used as the basis for delivering a taught programming course. Teachers and students can follow the tutorials in order of chapter, or return to particular chapters to build confidence and consolidate their knowledge and skills. Each chapter begins with a summary and concludes with a number of challenging exercises for students to try.

PHP on a Stick can also be issued to students to work independently, with little or no teacher intervention, to learn about programming and build exciting projects that really work.

A quick note about Microsoft Access
Although the application has been designed for projects that utilise PHP files and SQLite databases, *PHP on a Stick* can also be used to link PHP scripts with new or existing Access databases.

For this to work, Access must be installed, and you must ensure that the PHP scripts use absolute file paths to the required database upon connection.

The following pages have information on how to get up and running with *PHP on a Stick*.
This information is also accessible via the *PHP on a Stick* interface.

[1] *PHP on a Stick* has been tested successfully on Windows XP, Windows Vista and Windows 7

Getting Started with *PHP on a Stick*

Installing *PHP on a Stick*

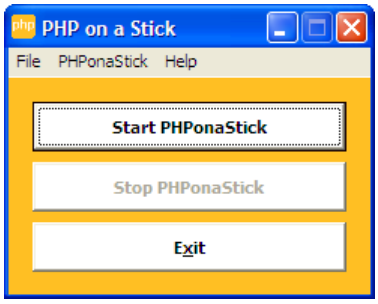
All that is required is a USB memory stick. The files needed to make *PHP on a Stick* work require approximately 150MB of storage space. Therefore, it is recommended that the USB stick has a capacity stick of at least 256MB.

Copy all files/folders into the ROOT of the USB memory stick from the CD-ROM (or from a computer/network location if you have already made a copy there). The device may be usable on other types of re-writable storage medium, e.g. MP3 players, but due to the range of different devices available, support on these is not guaranteed.

Note that any modifications to any file contents and/or directory structure may cause *PHP on a Stick* to malfunction, or not work at all. It is advisable that students use a memory stick solely for storing *PHP on a Stick*, although this is not essential.

Loading *PHP on a Stick* via Control Panel

With the USB memory stick inserted, navigate to the relevant drive (in Windows, this may be labelled as *'Removable Disk'*, with a yellow icon).



Inside the drive, double-click on **phponastick.exe**

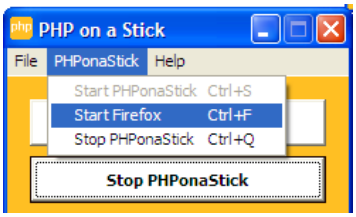
The *PHP on a Stick* Control Panel will appear (right). From here you can start and stop the application. To keep *PHP on a Stick* running, the Control Panel needs to be kept open.

Click **Start PHPonaStick**

At this point, a command box will open temporarily and will attempt to start the Apache service, before disappearing and loading the *PHP on a Stick* start screen in FireFox (see below).

If you receive a notification regarding security and your firewall, talk to your network administrator.

Note that the *PHP on a Stick* interface runs in FireFox Portable, which is provided with the software. If this web browser is accidentally closed, the quickest way to restore the interface is by clicking on the **PHP on a Stick** menu and selecting **Start Firefox** (or simply by pressing CTRL + F).



Start Screen

This is the starting place whenever *PHP on a Stick* is run. From here, a number of options are available:

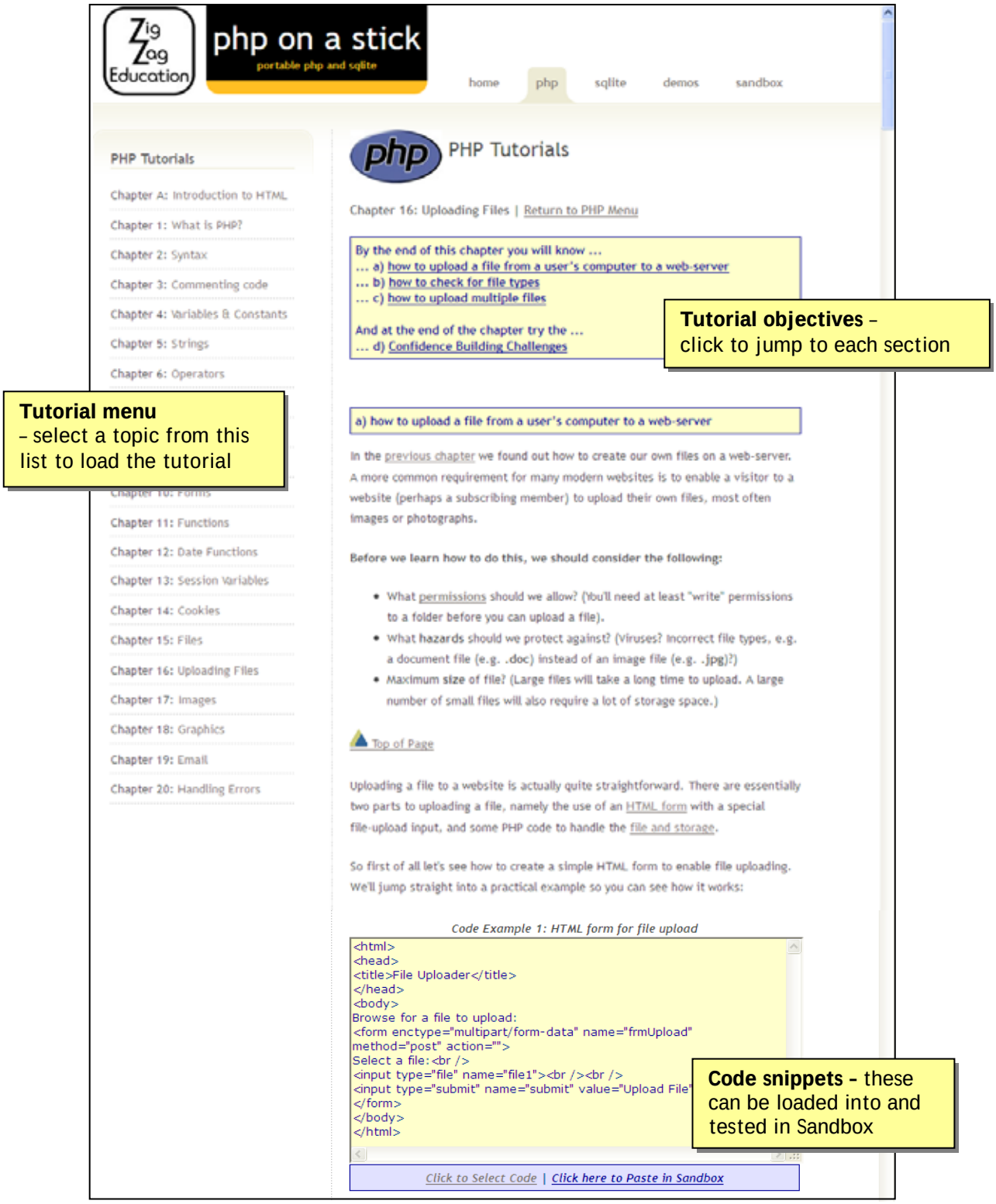
- **Introduction** – things to know before starting
- **Learn PHP** – 20 interactive PHP tutorials
- **Learn SQLite** – 12 interactive SQL tutorials
- **Project Demos** – PHP & SQL working examples
- **Sandbox** – build, save and run projects
- **SQLite Manager** – control panel for SQLite database

Learning PHP and SQLite

Learning PHP
If you're new to PHP then it is recommended that you use the PHP tutorials in the given order, as each chapter builds on the knowledge and skills learned in previous chapters. Otherwise feel free to jump straight to a chapter of your choice: you can return to a chapter at any time to refresh your PHP knowledge.

Learning SQLite
Before you start these tutorials you'll need to have at least a basic understanding of PHP, so don't begin with SQLite if you're new to PHP!

Figure 1: The Tutorial Interface (applies to PHP and SQLite)



Overview of PHP Tutorials

Chapter 1: **What is PHP?**

Chapter 2: **Syntax**
- *a bit about HTML tags*
- *how to add PHP script to an HTML webpage*
- *how to use PHP to write HTML information onto a webpage*

Chapter 3: **Commenting code**
- *why we add comments to our PHP code*
- *how to add single line comments*
- *how to add multi-line comments*

Chapter 4: **Variables & Constants**
- *the difference between variables and constants*
- *how to use variables in PHP*
- *how to use constants in PHP*

Chapter 5: **Strings**
- *how to define strings*
- *how to concatenate strings*
- *about some useful in-built PHP string functions*

Chapter 6: **Operators**
- *how to use arithmetic operators to do mathematical operations*
- *how to use assigment operators to give values to variables*
- *how to use comparison operators to compare variables*
- *how to use logic operators to check the result of conditions*

Chapter 7: **If, If-Else, Switch**
- *how to use IF-ELSE to make decisions in programming*
- *how to give further options by using ELSE IF*
- *how to use SWITCH to handle multiple conditions*

Chapter 8: **Looping**
- *FOR loops to repeat things a fixed number of times*
- *WHILE loops to repeat if or while a condition is true*
- *DO-WHILE loops to repeat at least once while a condition is true*

Chapter 9: **Arrays**
- *what an array is*
- *how to use numeric arrays*
- *how to use associative arrays*
- *about some useful array functions*
- *how to use multiple dimensional arrays*

Chapter 10: **Forms**
- *how to use HTML forms to enable a user to enter data onto a webpage*
- *how to send data from one webpage to another using the GET method*
- *how to send data from one webpage to another using the POST method*
- *how to use the REQUEST method*
- *about htmlentities*

Chapter 11: **Functions**
- *about user defined functions*
- *how to create your own functions*
- *some more about functions*

Chapter 12: **Date Functions**
- *how to use the inbuilt Date() function*
- *how to use the inbuilt mktime() function*
- *how to use the inbuilt time() function*
- *how to find the sunrise and sunset times for today!*

Chapter 13: **Session Variables**
- *what session variables are*
- *how to use session variables to share data between PHP files*

Chapter 14: **Cookies**
- *what cookies are*
- *how to use cookies to store data for return visits to a website*

Chapter 15: **Files**
- *how to create files using PHP*
- *how to write to and append files*
- *how to read files*
- *how to delete files*
- *how to create and delete folders (directories)*
- *how to change permissions of files and folders*
- *how to include files*
- *how to copy files*
- *how to check file type*

Chapter 16: **Uploading Files**
- *how to upload a file from a user's computer to a web-server*
- *how to check for file types*
- *how to upload multiple files*

Chapter 17: **Images**
- *about GD2*
- *how to transform images*
- *how to write text onto an image*
- *how to create screenshots*

Chapter 18: **Graphics**
- *how to create a simple image*
- *how to draw using pixels*
- *how to draw a line*
- *how to draw a rectangle*
- *how to draw an ellipse*
- *how to draw a polygon*
- *how to add graphics to an existing image*
- *how to create a dynamic bar chart*

Chapter 19: **Email**
- *how to set up a free email account*
- *the additional free files you'll need*
- *how to use PHPMailer to send an email*

Chapter 20: **Handling Errors**
- *about basic error handling*
- *how to create a custom error handler*
- *how to report errors*
- *how to trigger errors*

Overview of SQLite Tutorials

Chapter 1: **Introduction to Databases**
- *what a database is*
- *about flat file and relational databases*

Chapter 2: **Database Design & Normalisation**
- *about 3rd normal form*
- *how to design a normalised database*

Chapter 3: **Introduction to SQL**
- *what SQL is*
- *about Data Definition Language (DDL)*
- *about Data Manipulation Language (DML)*

Chapter 4: **Introduction to SQLite**
- *what SQLite is*
- *when to use SQLite instead of MySQL*
- *the limitations of SQLite*

Chapter 5: **Create Databases**
- *how to create an SQLite database*
- *how to open an existing SQLite database*
- *how to delete an existing SQLite database*

Chapter 6: **Create Tables**
- *about SQLite data-types*
- *how to create a new table into an existing SQLite database*
- *how to empty an existing table*
- *how to drop an existing table*

Chapter 7: **Insert Records**
- *how to insert a record into a table*
- *how to use an HTML form to insert a record*

Chapter 8: **Select Records**
- *how to select all records from a table*
- *how to use a filter to select specific records from a table*
- *how to order the way records are selected*
- *how to limit the results of a select search*
- *how to use pagination to improve the display of search results*

Chapter 9: **Update Records**
- *how to update specific records in a table*
- *how to update all records in a table*

Chapter 10: **Delete Records**
- *how to delete specific records in a table*
- *how to delete all records in a table*

Chapter 11: **SQL Functions**
- *how to use aggregate functions*
- *how to use scalar functions*

Chapter 12: **SQLite Manager**
- *what SQLite Manager is and where to get it from*
- *how to start SQLite Manager*
- *how to create a new database*
- *how to create and modify tables*
- *how to insert and update records*
- *how to export an SQLite database*
- *how to upload an existing SQLite database*

Using the Sandbox

The Sandbox is a unique and powerful tool that will enable you to build, save and test your own PHP files. It's really easy to use and there's no need to use any other 3rd party software to build your PHP projects: everything you need is right here. The best way to learn is to get stuck right in, and you'll soon be using the Sandbox with confidence to develop your own PHP applications.

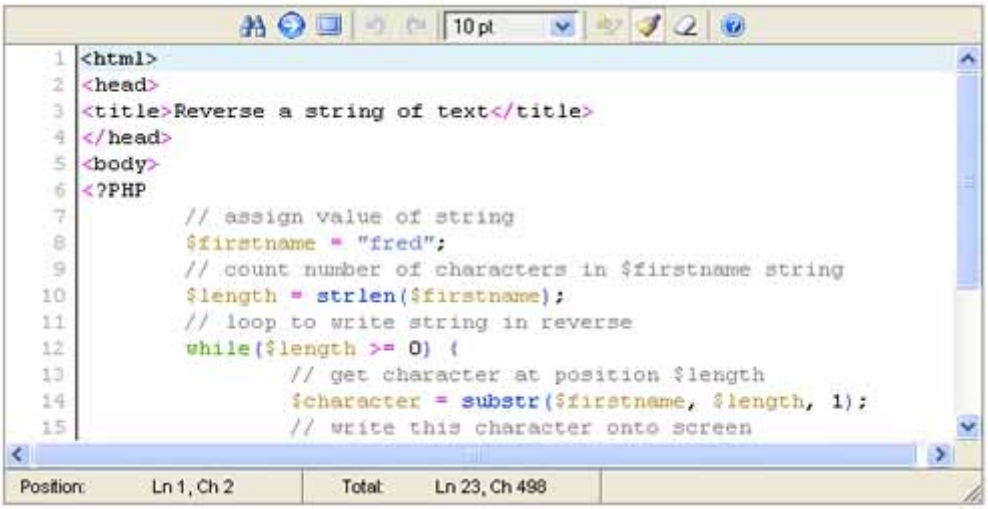
Figure 2: The Sandbox [in Editor Mode]

There are 5 parts to the Sandbox, as follows:

1. The editor

This is the place where you write your PHP. You can either type in your PHP code directly, or paste in code from elsewhere (e.g. use CTRL and V keys to paste). The Sandbox will also automatically paste code from examples in the **PHP** and **SQLite** tutorials! This will save you time and enable you to try out the code examples quickly and efficiently.

Figure 3: The Editor



As you can see in Figure 3 the editor has in-built syntax-highlighting. This means that it recognises keywords from within your PHP code (and HTML, JavaScript and CSS) and will display such keywords in different colours. If you click on a line of code then the whole line will be highlighted. Notice also that line numbers are shown automatically on the left-hand-side of the editor.

The editor supports tabbed-indentation, which will make your code easier to follow (see how the PHP code on the previous page is indented from the left).

At the top of the editor is an array of useful tools, including:

- **Search -** search (and replace if desired) any text.
- **Goto Line -** go to a specific line of code.
- **Full Screen -** expand editor to full screen size.
- **Undo -** undo last action in editor.
- **Redo -** redo last action in editor.
- **Highlight -** toggle syntax-highlighting on/off.
- **Reset Highlight -** reset highlighting.

You'll see a **font-size-adjust tool** with a range of text sizes to make reading of code easier - useful for large screens and projecting for the benefit of an audience. In fact, for such uses it's a good idea to select the largest font size.

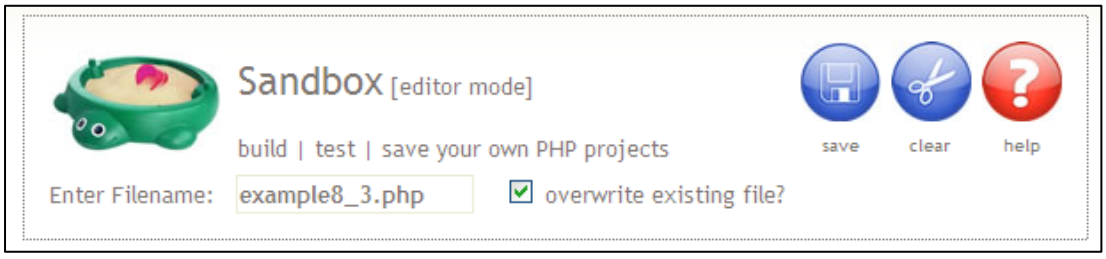
There's also a very useful tool at the foot of the editor:

- **Status Bar Resize Tool -** click to stretch editor to any custom size.

2. Saving your files

You'll need to save your files if you want to test them, and also to return to them later on. Saving files is easy, simply enter a **filename** (which must have a suitable file extension such as **.php** e.g. index.php) and click on the blue **save** button as shown below in Figure 4:

Figure 4: Saving Files



If you don't enter a filename, or if there's nothing in the editor, then you won't be able to save a file.

Note also the **overwrite** checkbox in Figure 4. By default this is checked, and will allow overwriting of existing files with the same name. If however you do not wish to overwrite an existing file, then click the **overwrite** checkbox.

All files created in the Sandbox will be saved in a directory on your USB stick named **myfiles**. The Sandbox can enable you to create sub-directories (folders) within the **myfiles** folder; we'll see how to do this next.

Hint: Using Windows Explorer, create a shortcut to the *myfiles* folder (**PHP on a Stick\udrive\www\Tutorials\sandbox\myfiles**) to allow easy access to the project files:

3. Opening and testing your files

At the left-hand-side of the Sandbox screen is the **file organiser**. Here you can open files for testing, click to edit or delete existing files, and open, create and delete directories. See Figure 5 below:

The file organiser shows all your saved files (if any) in a **tree-view** structure. Clicking on a folder icon will show the contents of that particular folder.

Clicking on a file will open that file in the editor.

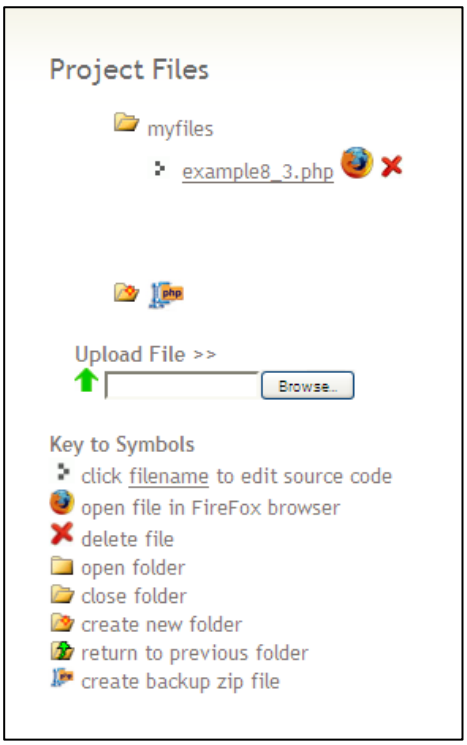
Clicking on the button will run the PHP file in a web browser (remember to save the file after making changes to see these changes in the browser).

Long file names will be truncated (shortened) so as not to adversely affect the screen display.

Please note that the following actions are not possible:

- × deleting a non-empty folder
- × creating a folder with the same name as an existing folder in the same directory
- × Using invalid characters in the name of a folder. If you wish to delete a folder then you'll first need to delete any files and folders within it.

Figure 5: File Organiser



4. Uploading files

You can also import files from any other location to the Sandbox.

Figure 6: Uploading Files

Click the **Browse...** button (as shown in Figure 6 above) to search for the file to be uploaded, and then click on the green arrow: to upload your selected file to the Sandbox. After a moment or two (depending on the size of file being uploaded) the Sandbox screen will reload and you'll be able to see the uploaded file in the tree-view under **Saved Files**.

The Sandbox can execute (run) the following types of file: PHP and HTML (file extensions **.php** and **.html**). It will also enable you to edit and create text files (**.txt** file extension, also CSS **.css** and JavaScript files **.js**) and open/view image files (**.gif** and **.jpg**).

It is possible to upload other file types (e.g. SQLite database files **.sqlite**) but these are not editable in the Sandbox, although these can be deleted.

Alternatively, you can also move files directly into the *myfiles* folder using Windows Explorer:

PHP on a Stick\udrive\www\Tutorials\sandbox\myfiles

They will automatically appear on the file list when the page is refreshed.

5. Backing up your saved files

It's always good practice to regularly backup your files, particularly when using a USB memory stick device. Such devices are easily lost and may be damaged if accidentally put through the wash in your back pocket!

Your own files and data will be more valuable to you than the *PHP on a Stick* application (that can easily be re-installed on a new USB memory stick). Therefore the Sandbox has an inbuilt backup facility that Zips all of your files and folders into a Zip file (maintaining the original folder tree structure) ready for you to save in a new location (e.g. onto the C: drive, your Desktop or just about anywhere else).

It is recommend that prior to using the Zip Backup Facility you download and install a Zip application, such as the free and excellent jZip **(**www.jZip.com).

If you have at least one file or folder saved in the Saved Files, then the **Backup to Zip** icon will appear as follows: 

Simply click on this icon and the Sandbox will create a Zip archive and then open a save-dialogue window giving you the option to open the Zip archive or save it to a new location. It is recommended that you always select the **Save** option and save the Zip archive to a new location (other than your USB memory stick).

Figure 7: Backing up Files

The Zip file will be given the unique name: **phpbackupDDMMYY_hhmmss.zip**

Where **DDMMYY** are the values of the current date (day, month, year), and **hhmmss** is the time that the backup was created (hours, minutes, seconds).

For example, if a backup was made on 22nd July 2011 at 3.30pm (and 29 seconds) then the Zip file archive would be named: **phpbackup220711_153029.zip**

In this way, no two backup files will have the same name, and it is easy to identify when backups were created.

Restoring backup files is easy. Simply unzip them (using a good Zip application, it is recommended that you use a Zip application (e.g. jZip) and then upload individual files to the Sandbox.

