Introduction and Principles

Unit 27 Web Server Scripting

Extended Diploma in ICT

Outcomes

- Understand the principles of web server scripting
- Be able to use web server scripting
- Understand the issues affecting web server scripting

Assignments

- 3

  - P1, M1 D1: Theory – principles
  - P2, P3, P4, M2, M3: Practical - code snippets
  - P5, M4, D2: Practical - make an application
  - P6, D3: Theory – issues and security

Server side Scripting

- What does the title to this slide actually mean?

- Can you name and server side scripting technology?

Unit brief introduction

- When designing and building websites, a key issue for developers is the amount of control they can exert over how tasks are carried out. *Client-side* scripting embedded in web pages can give additional functionality but, because the code is executed after the page has been loaded, there is little control and this approach can lead to hacking vulnerabilities and errors.

Unit brief introduction

- Web server scripting is code written ‘server-side’ and executed before the page is loaded. This means that complex tasks can be created and programming is generally more secure. The skills and knowledge developed in this unit are particularly valuable because security and reliability are common issues for businesses.

What is server side scripting?

- **Server-side scripting** is a web server technology in which a user's request is verified by running a script directly on the web server to generate dynamic web pages.
  (http://en.wikipedia.org/wiki/Server-side_scripting)
- It is used to create dynamic web pages
- Unlike client side scripting, the code is secure as it cannot be viewed on the client

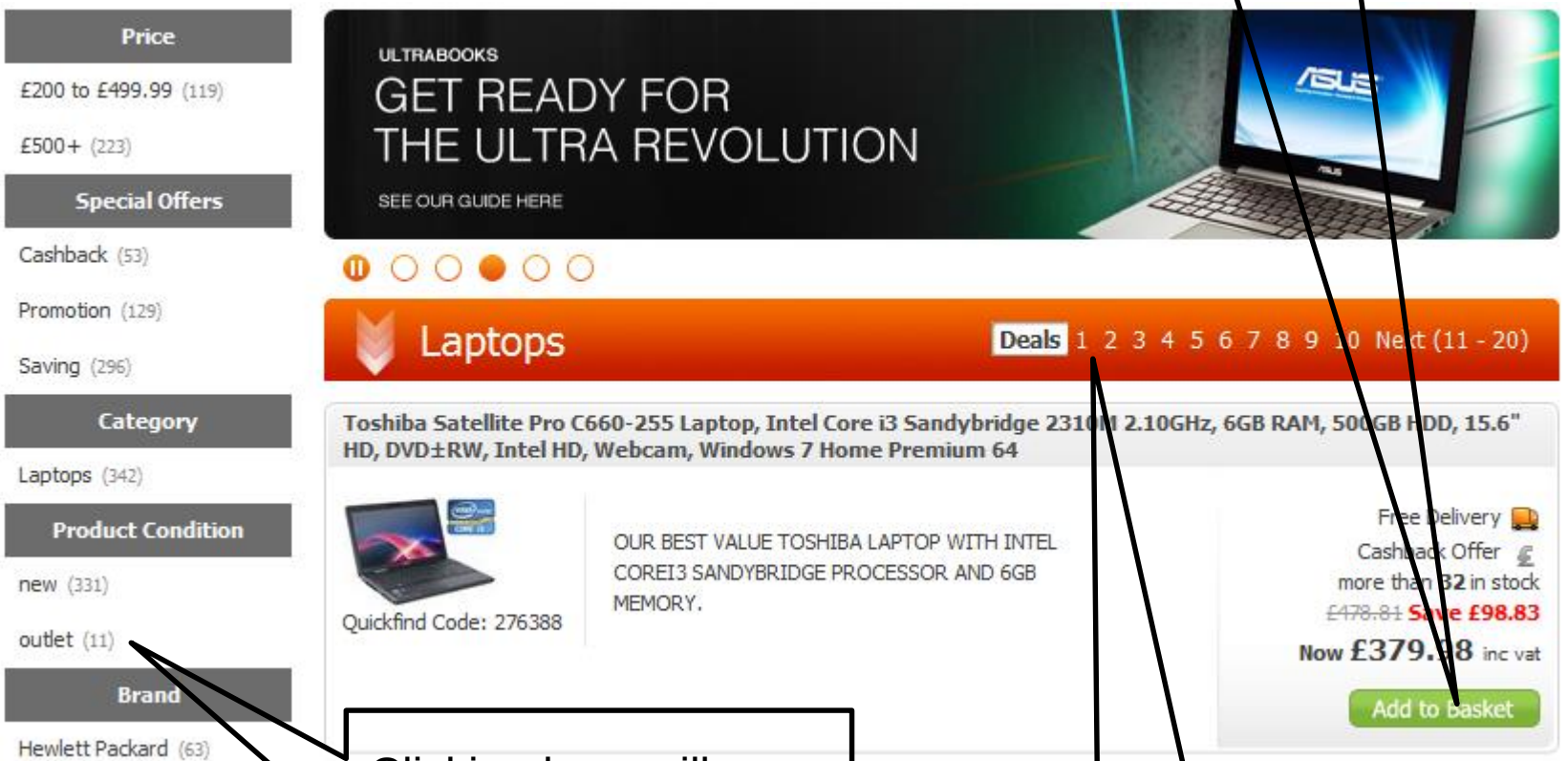
Static web pages

- The server contain .HTML files
- These are served to the client browser on request
- The local browser may have some interaction via client-side scripting (JavaScript)
- The HTML pages are fixed and cannot be altered to match individuals requests

Dynamic model

- HTML pages are generated when they are requested
- The basic page is done in a scripting language and uses HTML and CSS to determine static content and appearance
- The variable data is generated by scripts (an interpreted program) running on the web server

An example



Clicking here will run a script to add the product to a basket and update the basket status on this page

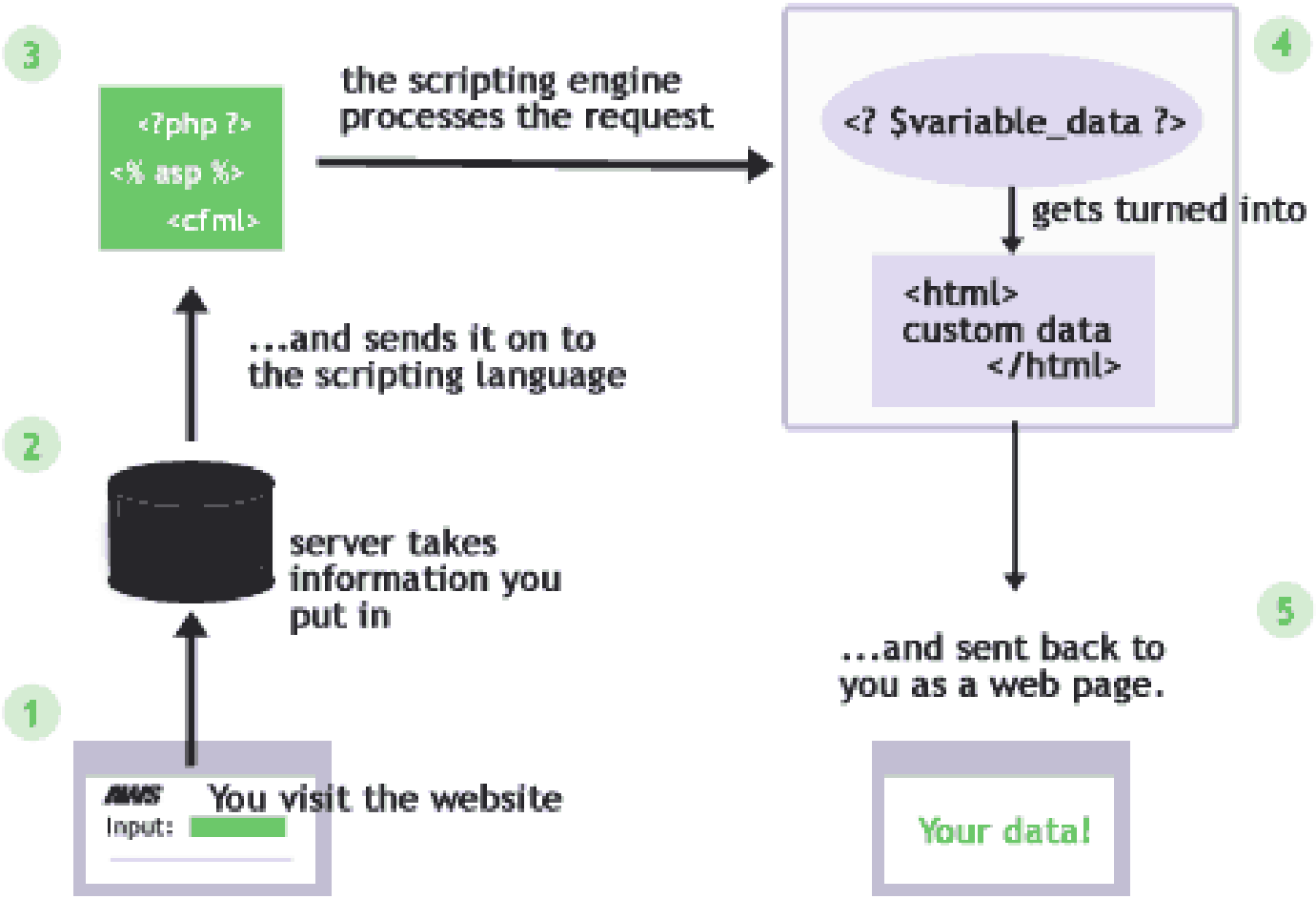
Clicking here will run a script using the new parameters to generate a new list

List generated by scripts

Other uses

- Updating databases
- Interacting with forums
- Gathering user statistics
- Managing profiles
- Blogging
- Sharing (upload to Web 2.0 applications)

How does it work?



Activity

- There are a variety of web server scripting languages
- Find 5 scripting languages and create a table which shows:
  - Open or proprietary
  - Free or paid for
  - Main features
  - Claimed advantages
- You must include PHP

ASP

- ***Active Server Pages (ASP)***
- a server-side scripting environment from Microsoft
- Uses a file extension of *.asp*.
- Scripts are usually written in VBScript
- ASP will normally run only on Microsoft servers

ASP.NET

- The successor to Microsoft's ASP
- Allows programmers to create web applications using any scripting or programming language supported by the .NET Framework.
- The main building blocks are pages known as *web forms*, which contain definitions for server-side *Web Controls* and *User Controls*,
- Web forms have the file extension *.aspx*

ColdFusion Markup Language (CFML)

- A scripting language originally introduced by Adobe Systems in 1995
- Enables web developers to embed database commands and other server-side scripting elements within standard HTML or XHTML
- Pages in a ColdFusion web application are pre-processed by the *ColdFusion Application Server* when requested by a client browser

Perl

- A high-level, interpreted programming language
- a procedural programming language loosely based on C
- Used for the creation of web applications, especially those where database access is required.
- Perl is free software

JavaServer Pages (JSP)

- A Java technology similar to ASP
- Used to create dynamically generated web pages by embedding Java programming code in HTML or XHTML documents
- A *JavaServerPage* is compiled into a Java *servlet* by an application server, rather than being interpreted
- (a servlet is a Java program that executes on the server to create dynamic web pages).

PHP

- a widely-used scripting language
- PHP is free software released
- PHP code can be embedded into HTML or XHTML documents
- It is executed on the server to generate dynamic web content.
- PHP is frequently used together with MySQL, and is one of the key technologies in the

What we will use for this unit

- Web server scripts written in PHP
- Database commands written in SQL (structured query language)
- Client scripts written in JavaScript
- Apache Web server
- MySQL database managed with PHPMyAdmin
- PHP interpreter add on to Apache
- PHPStorm IDE (or …….?)

College set up

- Each student# login has
  - A folder for storing the web site
  - MySQL
  - PHP
  - FTP access
  - Apache
- For home you can use XAMPP

College set up

- Setting up your hosting account –
- From the wiki –

Tools

- Your choice of browser
- Contextual editors
  - Notepad++ (Windows)
  - GEDIT (Linux)
  - phpStorm
- FTP
  - Filezilla
  - Windows File Explorer

A simple PHP script

```
<html>
  <head>
    <title>Today&rsquo;s Date</title>
    <meta http-equiv="content-type“
        content="text/html; charset=utf-8"/>
  </head>
  <body>
    <p>Today&rsquo;s date (according to this web server) is
      <?php
      echo date('l, F dS Y.');
       ?>
    </p>
  </body>
</html>
```

<?PHP is the opening tag
(?> is the closing tag)

Using the PHP date function to get the date set on the server and echo it to the screen

Practicalities

- Use an FTP client (I use Filezilla) to create a subdirectory on your student account
  - student#.computing.hct.ac.uk/Unit27
- Use a contextual editor (Notepad++ or gedit to create the file we just used and save it with a file type of .php (I called it Unit27PHPTest.php) in the subdirectory
- Access it using your browser
  - http://studentnn.computing.hct.ac.uk/Unit27/Unit27PHPTest.php
- Make sure it works

Results

- Browser displays: Today's date (according to this web server) is Tuesday, March 27th 2012.
- Source is:

```
<body>
<p>Today&rsquo;s date (according to this
web server) is Tuesday, March 27th 2012.
</p>
</body>
```

Assignment 1 criteria

- P1 Explain the principles of web server scripting
- M1 Compare server-side and client-side scripting
- D1 Evaluate the combined use of client and web server scripting

Client side and server side

- Client side
  - Reduces web traffic as processing done on client
  - Good for
    - Form validation
    - User aids (drop downs, hover etc)
    - Local navigation
- Server side
  - Can use centralised server resources eg databases
  - No client plug-ins required, runs in any browser
  - No load on client, can use powerful server hardware
  - Applications do not need distributing

Combined use

- Build efficient applications
- Choose to do the processing in the most appropriate location
- Choose where to keep up to date multi use data
- Choose how to optimise the user experience:
  - Performance
  - Appearance
  - Function

Summary: web server scripting

Advantages

- Interpreted, low processing overhead
- Interpreter is integrated into web server, so it is fast
- Secure, code runs on server
- Content based on live data
- Can use basic, eg phone, browsers as processing is server side

Disadvantages

- Debugging tools can be scarce
- Can be more difficult to develop
- Requires a running server to test
- Has no direct control over the user interface