



SWINBURNE  
UNIVERSITY OF  
TECHNOLOGY

# Advanced Web Development: An Overview and Introduction to Advanced Web Development

Week 1



# Outline

---

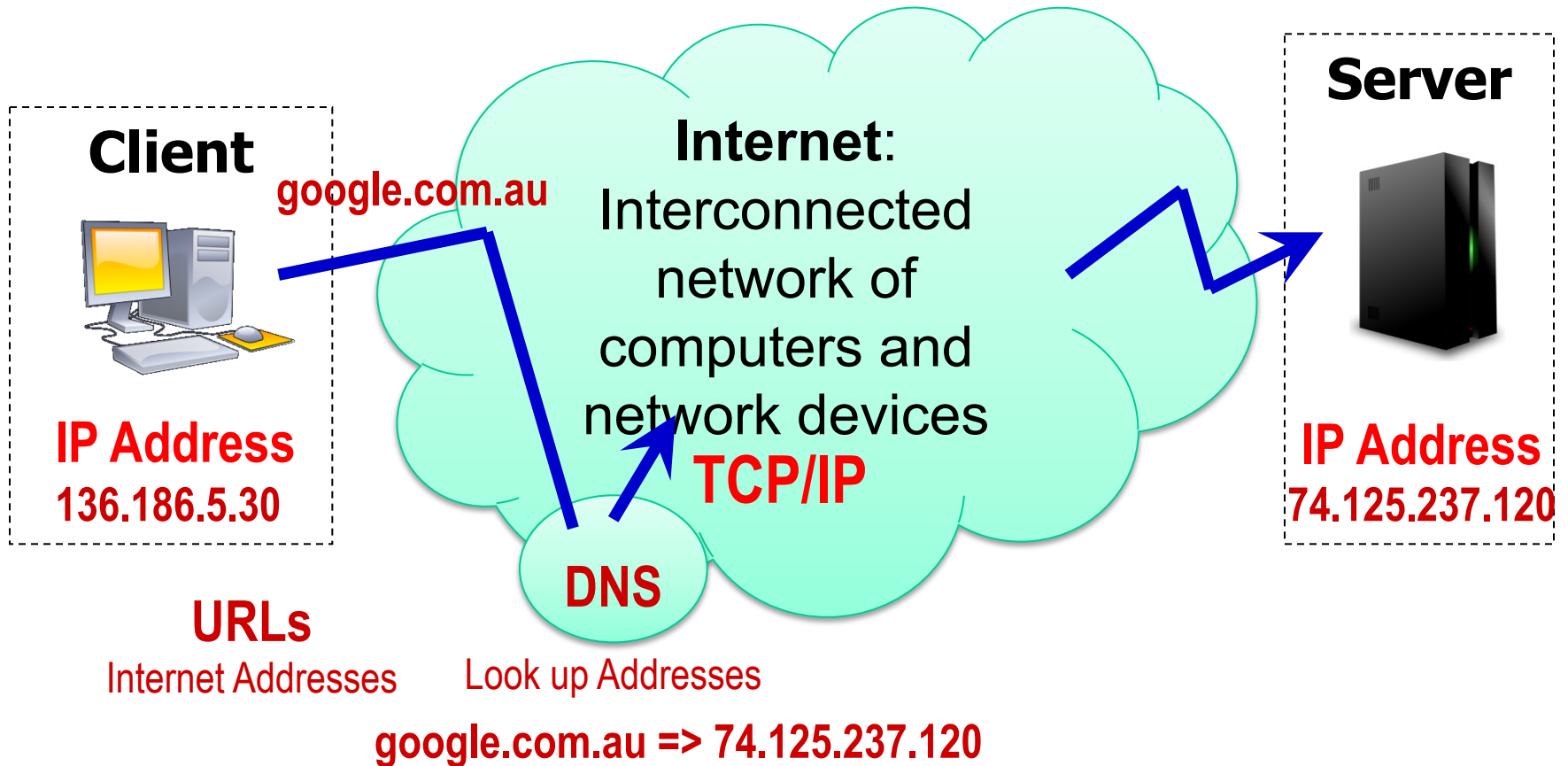


- Introduce the unit (see syllabus)
- Revisions
- Create basic PHP scripts



# Internet

## DNS, URLs



# Domain Name Registration

---



- Pick a domain name that is similar to your business name or that describes your Web site
- You cannot use a domain name that is already in use or a trademarked name
- Contact a domain name registrar to find out the availability of a domain name and register it
- Domain names are stored in a master database that is maintained by the InterNIC



# Client/Server Architecture

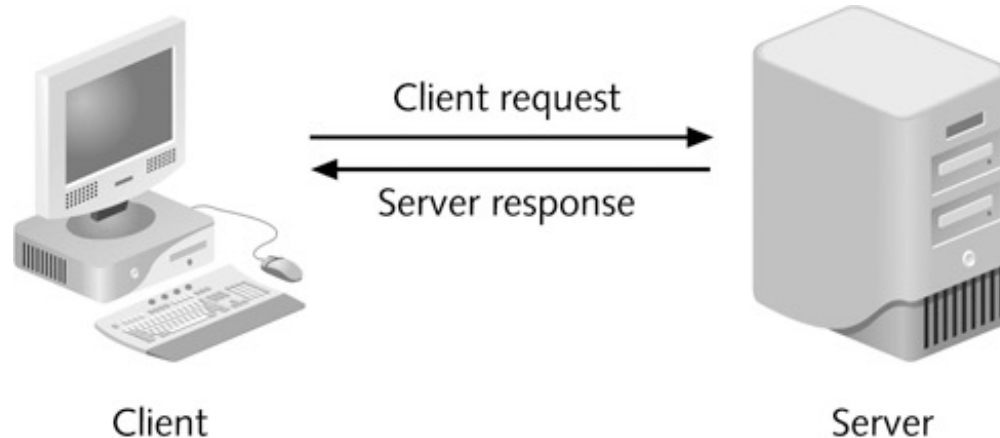
---

- Server (“back end”):
  - A database from which a client requests information
  - Fulfills a request for information by managing the request or serving the requested information to the client
  - Responsible for data storage and management
- A system consisting of a client and a server is known as a two-tier system

# Client/Server Architecture (continued)



- Client (“front end”):
  - Presents an interface to the user
  - Gathers information from the user, submits it to a server, then receives, formats, and presents the results returned from the server



**The design of a two-tier client/server system**

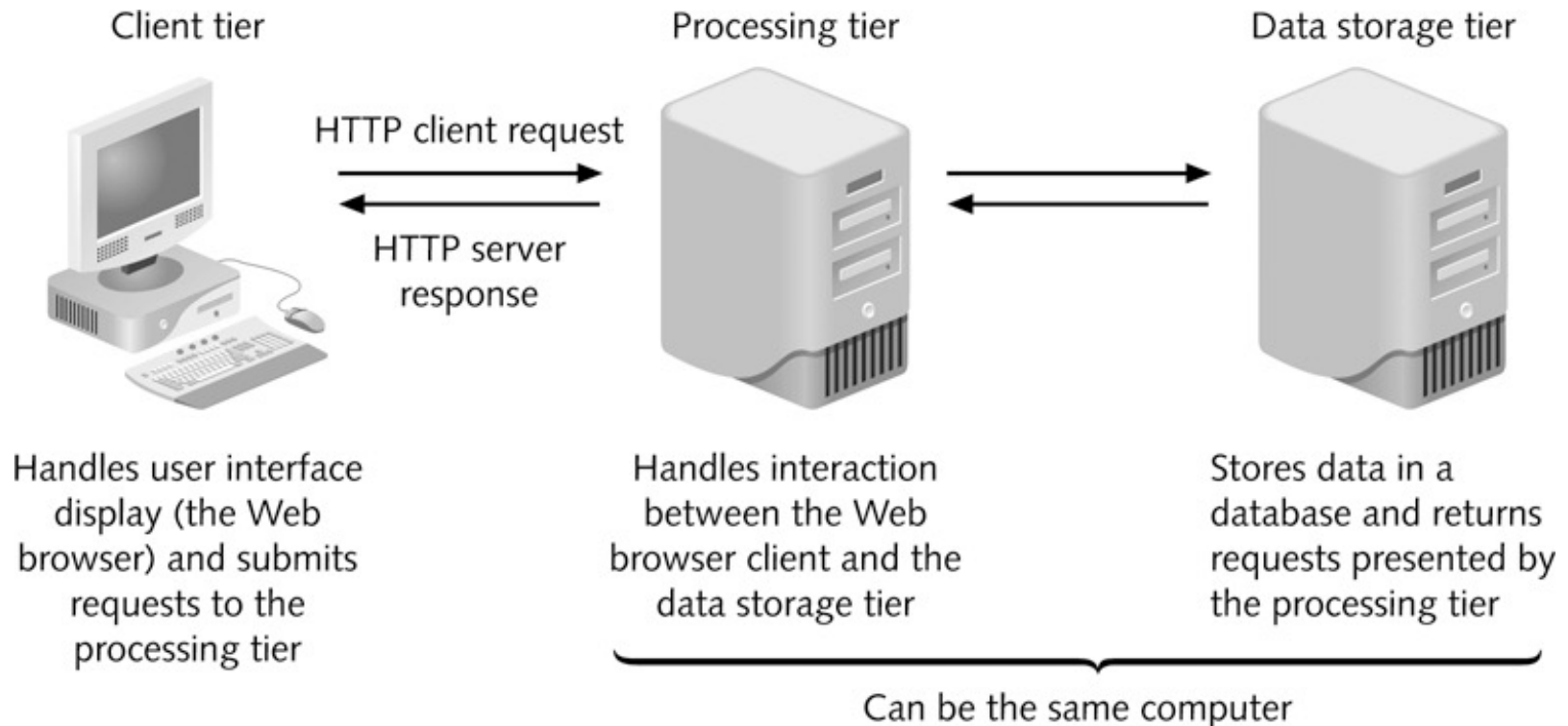
# Client/Server Architecture (continued)

---



- A **three-tier**, or **multi-tier**, client/server system consists of three distinct pieces:
  - **Client tier**, or **user interface tier**, is the Web browser
  - **Processing tier**, or **middle tier**, handles the interaction between the Web browser client and the data storage tier
    - Performs necessary processing or calculations based on the request from the client tier
    - Handles the return of any information to the client tier

# Client/Server Architecture (continued)



## The design of a three-tier client/server system



# Client-Side Scripting and JavaScript

---



- **Client-side scripting** is a language that runs on a local browser (on the client tier) instead of on a Web server (on the processing tier)
- **JavaScript** allows you to:
  - ☐ Turn static Web pages into applications such as games or calculators
  - ☐ Change the contents of a Web page after a browser has rendered it
  - ☐ Create visual effects such as animation
  - ☐ Control the Web browser window itself

# Server-Side Scripting and PHP

---



- **Server-side scripting** refers to a scripting language that is executed from a Web server
- **Hypertext Preprocessor (PHP)** is a server-side scripting language that is used to develop interactive Web sites
  - ☐ Is easy to learn
  - ☐ Includes object-oriented programming capabilities
  - ☐ Supports many types of databases (MySQL, Oracle, Sybase, ODBC-compliant)

# Server-Side Scripting and PHP (continued)

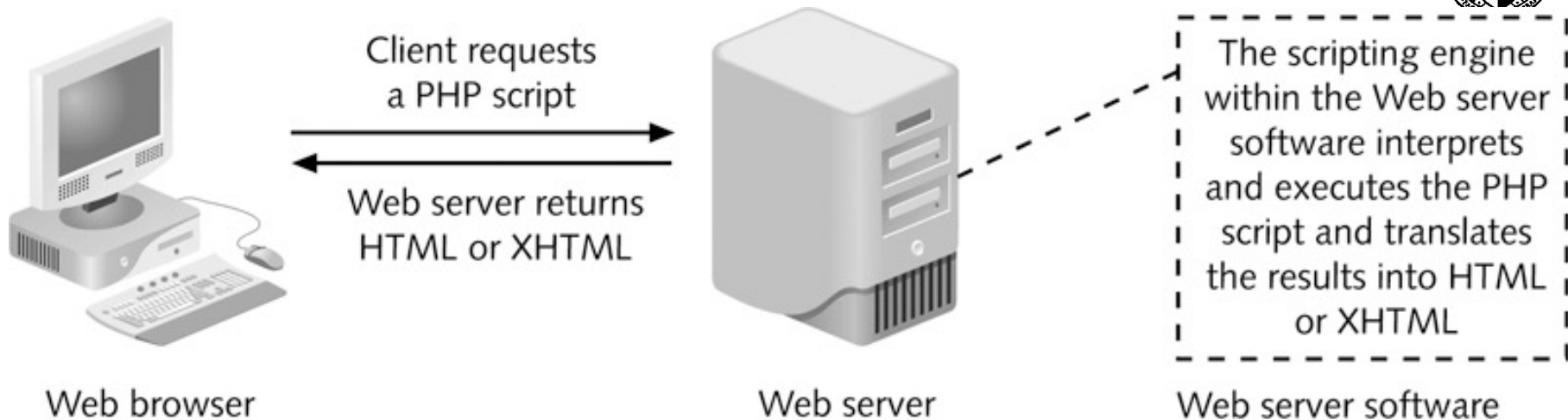
---



## ■ PHP (continued):

- ☐ PHP is an open source programming language
  - ☐ Open source refers to software where source code can be freely used and modified
- ☐ Can't access or manipulate a Web browser like JavaScript
- ☐ Exists and executes solely on a Web server, where it performs various types of processing or accesses databases

# Server-Side Scripting and PHP (continued)



## How a Web server processes a PHP script

- General rule: Use client-side scripting to handle user interface processing and light processing, such as validation; use server-side scripting for intensive calculations and data storage



# Basic PHP Scripts

# A Web Development Environment

---

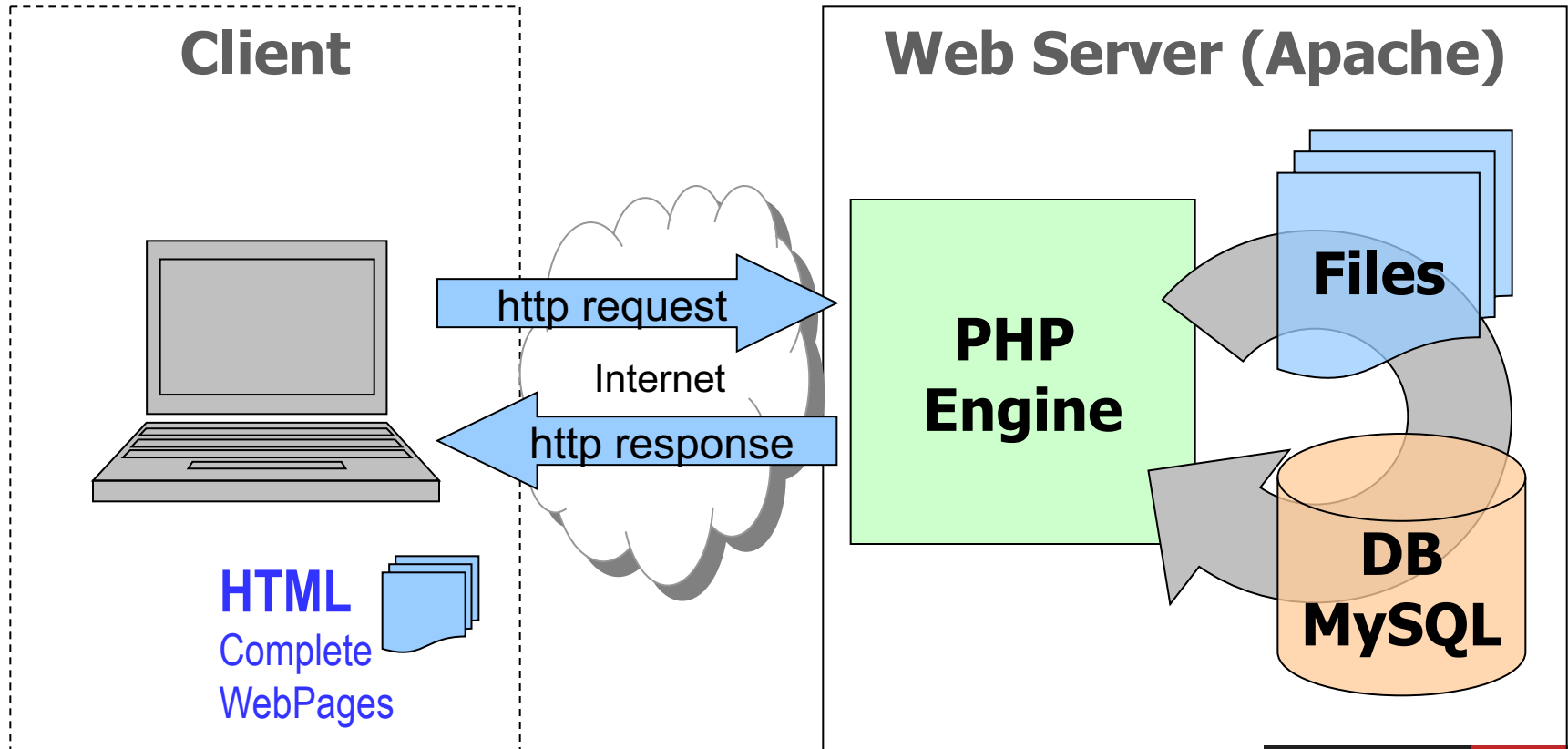


- A Web browser
  - e.g. IE, Firefox, Safari
- A Web server
  - e.g. Apache
- The PHP software
- A database
  - e.g. MySQL



# Embedded Scripting

## ■ Apache/PHP/MySQL example





# Creating Basic PHP Scripts

---

- **Embedded scripting languages** (JavaScript or PHP) refer to code that is embedded within a Web page (either an HTML or XHTML document)
- This code is typed directly into a Web page as a separate section
- A Web page document containing PHP code must have an extension of .php
- PHP code is never sent to a client's Web browser



# Creating Basic PHP Scripts (continued)

---



- The Web page generated from the PHP code, and HTML or XHTML elements found within the PHP file, is returned to the client
- A PHP file that does not contain any PHP code should have an **.html** extension
- **.php** is the default extension that most Web servers use to process PHP scripts



# Creating PHP Code Blocks

- **Code declaration blocks** are separate sections within a Web page that are interpreted by the scripting engine
- There are four types of code declaration blocks:
  - **Standard PHP script delimiters**
    - `<?php` statements; `?>`
  - (The `<script>` element)
    - `<script language =“php”>` statements; `</script>`
  - (Short PHP script delimiters)
    - `<?>` statements; `?>`
  - (ASP-style script delimiters)
    - `<%>` statements; `%>`

# Standard PHP Script Delimiters

---



- A **delimiter** is a character or sequence of characters used to mark the beginning and end of a code segment
- The standard method of writing PHP code declaration blocks is to use the `<?php` and `?>` script delimiters
- The individual lines of code that make up a PHP script are called **statements**

```
<?php  
    statements;  
?>
```

e.g. `echo "Hello World!";` as a statement

# Understanding Functions

---



- A **function** refers to a procedure that performs a specific task
- To execute a function, you must invoke, or **call**, it from somewhere in the script
- A **function call** is the function name followed by any data that the function needs
- The data (in parentheses following the function name) are called **arguments** or **actual parameters**
- Sending data to a called function is called **passing arguments**

e.g. `<?php phpinfo () ; ?>`



# Displaying Script Results

---

- To return to the client the results of any processing that occurs within a PHP code block, you must use an `echo()` statement or the `print()` statement
- The **`echo()`** and **`print()`** statements create new text on a Web page that is returned as a response to a client

# Displaying Script Results (continued)

---



- The `echo()` and `print()` statements are language constructs of the PHP programming language, not functions.
- A **programming language construct** refers to a built-in feature of a programming language
- The `echo()` and `print()` statements are virtually identical except:
  - The `print()` statement returns a value of 1 if it is successful
  - It returns a value of 0 if it is not successful

# Displaying Script Results (continued)



- Use the `echo()` and `print()` statements to return the results of a PHP script within a Web page that is returned to a client
- A **text string**, or **literal string**, is text that is contained within double or single quotation marks
- To pass multiple arguments to the `echo()` and `print()` statements, separate them with commas like arguments passed to a function

```
<?php echo "Explore Africa,", "South America,", "and  
Australia!"; ?>
```

# Creating Multiple Code Declaration Blocks



For multiple script sections in a document, include a separate code declaration block for each section

```
...
</head>
<body>
    <h1>Multiple Script Sections</h1>
    <h2>First Script Section</h2>
    <?php echo "<p>Output from the first script section.</p>";?>
    <h2>Second Script Section</h2>
    <?php echo "<p>Output from the second script section.</p>";?>
</body>
</html>
```





# Creating Multiple Code Declaration Blocks

(continued)

PHP code declaration blocks execute on a Web server  
***before*** the Web page is sent to a client

...

```
</head>
```

```
<body>
```

```
    <h1>Multiple Script Sections</h1>
```

```
    <h2>First Script Section</h2>
```

```
    <p>Output from the first script section.</p>
```

```
    <h2>Second Script Section</h2>
```

```
    <p>Output from the second script section.</p>
```

```
</body>
```

```
</html>
```

# Creating Multiple Code Declaration Blocks

(continued)



```
</head>
<body>
  <h1>Multiple Script Sections</h1>
  <h2>First Script Section</h2>
  <?php echo "<p>Output from the first script section.</p>";?>
  <h2>Second Script Section</h2>
  <?php echo "<p>Output from the second script section.</p>";?>
</body>
</html>

</head>
<body>
  <h1>Multiple Script Sections</h1>
  <h2>First Script Section</h2>
  <p>Output from the first script section.</p>
  <h2>Second Script Section</h2>
  <p>Output from the second script section.</p>
</body>
</html>
```

# Creating Multiple Code Declaration Blocks

(continued)



---

## Multiple Script Sections

### First Script Section

Output from the first script section.

### Second Script Section

Output from the second script section.

**Output of a document with two PHP script sections**



SWINBURNE  
UNIVERSITY OF  
TECHNOLOGY



# Case Sensitivity in PHP

- Programming language constructs in PHP are **mostly** case **insensitive**

```
<?php
```

```
    echo "<p>Explore <strong>Africa</strong>, <br />";
```

```
    Echo "<strong>South America</strong>, <br />";
```

```
    ECHO " and <strong>Australia</strong>!</p>";
```

```
?>
```

- However, ***code consistently***.
- Exceptions to case insensitivity:  
**variable** and **constant names** which **are** case sensitive and are studied later.

# Adding Comments to a PHP Script

---



- **Comments** are nonprinting lines placed in code such as:
  - ☐ The name of the script
  - ☐ Your name and the date you created the program
  - ☐ Notes to yourself
  - ☐ Instructions to future programmers who might need to modify your work
- Line comments hide a single line of code
  - ☐ Add `//` or `#` before the text
- Block comments hide multiple lines of code
  - ☐ Add `/*` to the first line of code
  - ☐ And `*/` after the last character in the code

# Adding Comments to a PHP Script (continued)



```
<?php
```

```
/*
```

```
This line is part of the block comment.
```

```
This line is also part of the block comment.
```

```
*/
```

```
echo "<h1>Comments Example</h1>"; // Line comments
```

```
// This line comment takes up an entire line.
```

```
# This is another way of creating a line comment.
```

```
/* This is another way of creating
```

```
a block comment. */
```

```
?>
```



---

# Summary

# Summary

---



- JavaScript is a client-side scripting language that allows Web page authors to develop interactive Web pages and sites
- Hypertext Preprocessor (PHP) is a server-side scripting language that is used for developing interactive Web sites
- Open source refers to software for which the source code can be freely used and modified



# Summary (continued)

---



- You write PHP scripts within code declaration blocks, which are separate sections within a Web page that are interpreted by the scripting engine
- The individual lines of code that make up a PHP script are called statements
- The term function refers to a procedure (or individual statements grouped into a logical unit) that perform a specific task

# Summary (continued)

---



- The term programming language construct refers to a built-in feature of a programming language
- Programming language constructs in PHP are mostly case insensitive, although there are some exceptions
- Comments are nonprinting lines that you place in code to contain various types of remarks