

# Introduction to PHP

Χριστόφορος Παναγιώτου

# Why Use PHP

- ▶ If you like **free** software or need a free solution
- ▶ If you need a solution that's **portable** across multiple platforms (e.g. Red Hat Linux to Windows 2000)
- ▶ If you want to add **dynamic content** to your pages
- ▶ If you want to make your pages easier to maintain
- ▶ Examples of uses of PHP :
  - Surveys – Polls
  - Small/Medium Portals
  - Small/Medium Web–Mails
  - Content Management

# Overview of PHP

- ▶ Open Source server-side scripting language designed specifically for the web.
- ▶ In-line scripting
- ▶ Conceived in 1994, now used on +10 million web sites. Now in version 5.3 (5.4 as RC)
- ▶ Outputs not only HTML but can output XML, images (JPG & PNG), PDF files and even Flash movies (using libswf and Ming) all generated on the fly. Can write these files to the file system.
- ▶ Supports a wide-range of databases (inherently or via ODBC).
- ▶ PHP also has support for talking to other services using protocols such as LDAP, IMAP, SNMP, NNTP, POP3, HTTP.
- ▶ Supports OO programming
- ▶ Perl- and C-like syntax. Relatively easy to learn.
- ▶ Website @ <http://www.php.net/>

# PHP Files

- ▶ A PHP file
  - may contain text, HTML tags and scripts
  - is returned to the browser as plain HTML
  - have a file extension of ".php", ".php3", or ".phtml"

# PHP and HTML

## ▶ Embedding PHP in HTML

```
<html>
<body>
  <strong>Hello World!</strong><br />
  <?
  echo 'This is a PHP introductory course!';
  ?>
</body>
</html>
```

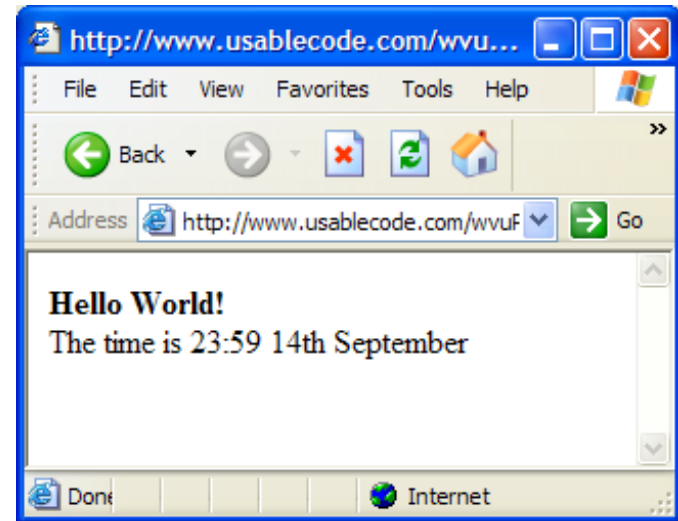
## ▶ PHP tag styles:

- Short: `<?php ?>`, `<? ?>`,
- Script: `<script language='php'></script>`

# PHP and Dynamic Content

- ▶ Adding dynamic content by adding the date to the page.

```
<html>
<body>
  <strong>Hello World!</strong><br />
  <?
  echo 'The time is';
  echo date('H:i jS F');
  ?>
</body>
</html>
```



- ▶ Date()  
<http://www.php.net/manual/en/function.date.php>
- ▶ PHP Function Reference  
<http://www.php.net/manual/en/funcref.php>

# PHP Inclusions

- ▶ Including code/HTML into your page

```
<?php
include '../includes/header.html';
?>
<center>
content of your web page
</center>
<?php
include 'http://cs.ucy.ac.cy/php/footer.html';
?>
```

- ▶ Content can be included from a local or remote source via such protocols as HTTP, HTTPS, FTP, and FTPS

# PHP Variables

- ▶ Variables: Are the symbols we use to represent data.
- ▶ Variable names can be of any length; can include letters, numbers and underscores; cannot start with a digit; case-sensitive; and can have the same name as a function.
- ▶ To assign values to variables:
  - `$foo = 'bar'`; Data Type: String
  - `$foo = 1`; Data Type: integer
  - `$foo = 5.34`; Data Type: Double
  - `$foo = array("bar","united")`; Data Type: Array
- ▶ Data Types are automatically assigned though you can force a data type by type casting. For example:
  - `$foo = 'Hello'`;
  - `$bar = (int)$foo`;
  - `$bar` now equals 0
- ▶ Almost all variables are local (page). Globals include `$_Session`



# PHP Operators

- ▶ Operators: Operators are symbols that you can use to manipulate values and variables by performing an operation on them.
- ▶ Web Site @ <http://www.php.net/manual/en/language.operators.php>
- ▶ Includes:
  - Assignment (e.g. =, +=, \*=)
  - Arithmetic (e.g. +, -, \*, /, ++)
  - Comparison (e.g. <, >, >=, ==, !=)
  - Logical (e.g. !, &&, ||)

# PHP Basics

## ▶ Concatenation

```
<html>
  <body>
    <?php
      $txt1="Hello World";
      $txt2="1234";
      echo $txt1 . " " . $txt2 ;
    ?>
  </body>
</html>
```

## • Comments

```
<html>
  <body>
    <?php
      //This is a comment

      /* This is a comment block
      */
    ?>
  </body> </html>
```

# PHP Basics

- ▶ Control Structures: Are the structures within a language that allow us to control the flow of execution through a program or script.
- ▶ Grouped into conditional (branching) structures (e.g. if/else) and repetition structures (e.g. while loops).
- ▶ Example if/elseif/else statement:

```
if ($foo == 0) {  
    echo 'The variable foo is equal to 0';  
}  
else if (($foo > 0) && ($foo <= 5)) {  
    echo 'The variable foo is between 1 and 5';  
}  
else {  
    echo 'The variable foo is equal to ' . $foo;  
}
```

- ▶ Good code will use indents and comments.

# PHP Basics

- ▶ Example of “switch” statement

```
<html><body>
<?php
    switch ($x)
    { case 1:
      echo "Number 1";
      break;
    case 2:
      echo "Number 2";
      break;
    case 3:
      echo "Number 3";
      break;
    default:
      echo "No number between 1 and 3";
    }
?>
</body></html>
```

# PHP Basics

- ▶ The “While” statement

```
while (condition)  
{  
    code to be executed ;  
}
```

- ▶ The “do...while” Statement

```
do  
{  
    code to be executed;  
}  
while (condition);
```

- ▶ The “for” Statement

```
for (initialization; condition; increment)  
{  
    code to be executed;  
}
```

# PHP Basics

## ▶ PHP built-in functions

- **phpinfo()**: Outputs PHP information like version and configuration – used for troubleshooting
- **PHP server variables**: Holds information like the user's browser, which URL the user came from e.t.c. The server variables have **global scope**

```
<html><body>
<?php
echo "Browser: " . $_SERVER["HTTP_USER_AGENT"] . "<br />";
echo "User's IP address: " . $_SERVER["REMOTE_ADDR"] .
    "<br />";
echo " Server address: " . $_SERVER["SERVER_ADDR"] ;
?>
</body></html>
```

# PHP Basics

## ▶ PHP Header() Function

- Sends Raw HTTP headers over the HTTP protocol
- Must be called before anything is written to the page (including HTML). Call it before the <HTML> tag.

```
<?php
```

```
//Redirect browser
```

```
header("Location: http://www.cs.ucy.ac.cy/");
```

```
?>
```

```
<html>
```

```
<body>.....</body>
```

```
</html>
```

# PHP Basics

- ▶ Accessing Form Variables
- ▶ Three methods
  - Short: `$varfoo`,
  - Medium: `$_POST['varfoo']`, (recommended for versions of PHP +4.1)
  - Long: `$HTTP_POST_VARS['varfoo']`
- ▶ Tip: For checkbox variables your variable name should end with `[]`
  - Checkbox results are automatically put into an array
  - Example: `<input type=checkbox name=foo[] value=Y>`
- ▶ Accessing Querystring Variables
  - <http://cs.ucy.ac.cy/PHP/qrystring.php?FName=Marios&LName=Tziakouris>
  - `echo $_Get['FName'] . " " . $_Get['LName'];`



# PHP Basics

- ▶ Form processing example

```
<?php
if (!empty($_POST['campus'])) {
    echo "Welcome to {$_POST['campus']}";
}
else
    echo "Welcome to UCY";
?>
```

```
<form action="<?php echo$_SERVER['PHP_SELF']; ?>"
method="post">
```

```
Enter your campus: <input type="text"
name="campus">
```

```
<input type="submit">
```

# PHP Sessions

## ▶ Session Handling

- The idea of a session is to track a user during a single session on a web site. This enables customized web pages, single login during a session, shopping cart applications, and tracking users behavior
- Cryptographically generated to be a unique session id
- Session ID is stored as a cookie on the client box or passed along through URL's.
- Session variable values are stored in the 'superglobal' associative array '`$_SESSION`'
- The values are actually stored at the server and are accessed via the session id from your cookie.
- On the client side the session ID expires when connection is broken.

# PHP Sessions

## ▶ Session handling example

### ◦ Page 1

```
<?php
    session_start();
    $_SESSION['FName'] = $_Get['FName'];
    $_SESSION['LName'] = $_Get['LName'];
    include '../includes/header.html';
?>
```

### ◦ Page 2

```
<?php
    session_start();
    echo $_SESSION['FName'] . " " . $_SESSION['LName'];
?>
```

# PHP Cookies

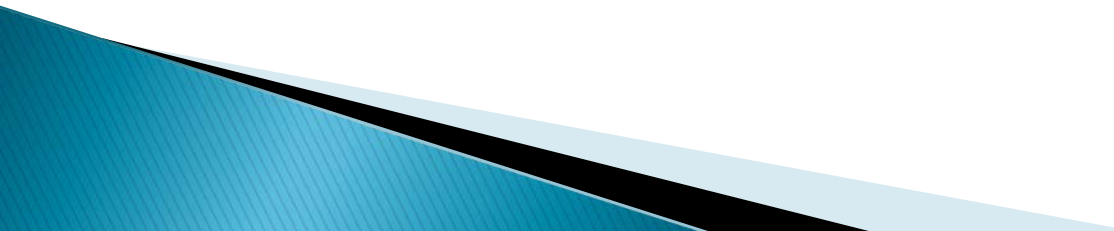
## ▶ Cookies

- Cookies are little text file that a web site stores in the client's computer to maintain information about that client
- Cookies are sent along with the rest of the HTTP headers
- Like other headers, cookies must be sent *before* any output from your script (this is a protocol restriction).
- This requires that you place calls to this function prior to any output, including `<html>` and `<head>` tags

# PHP Cookies

- ▶ Setting a cookie
  - `setcookie("TestCookie", "Ing=en");`
- ▶ Setting a cookie with expiration
  - `setcookie("TestCookie", "Ing=en", time()+3600);`  
`/* expire in 1 hour */`
- ▶ Access and print a cookie
  - `echo $_COOKIE['TestCookie']`
- ▶ Delete a cookie
  - `setcookie ("TestCookie", "", time() - 3600);`
  - set the expiration time to an hour ago

# PHP and Databases

- ▶ PHP and MySQL are a perfect companion
  - ▶ Largely because they are both free and they have numerous capabilities
  - ▶ PHP as of version 3 supports inherently MySQL i.e. specialized build-in functions handle the database interactions
  - ▶ Same goes with ORACLE but not with Microsoft databases (Access, SQL Server)
- 

# Using PHP to Query a MySQL Database

```
<html>
<body>
<h1>A List of Users Who Have Signed Up For ....</h1>
<?
    $dbh = mysql_connect("localhost","dbusername","dbpassword")
        or die("Couldn't connect to database.");
    $db = mysql_select_db("dbname", $dbh)
        or die("Couldn't select database.");
    $sql = "SELECT username, email FROM userspool";
    $result = mysql_query($sql)
        or die("Something is wrong with your SQL statement.");

    while ($row = mysql_fetch_array($result)) {
        $username = $row['username'];
        $email = $row['email'];
        echo '<a href="mailto:'. $email.'">'. $username.'</a><br />';
    }
?>
</body>
</html>
```

# Using PHP to Query a MySQL Database (cont.)

- ▶ Notes for previous slide example:
  - The first option in `mysql_connect` can be an IP address.
  - `mysql_query` returns a small table with your results in it. The while loop then goes through each record of that small table and pulls out the attributes/fields you selected in your SQL statement.
  - `die( )` will kill the script. Make sure that that text is informative.
  - If you use a function in your SQL query then it has to be a part of the `$row` statement. For example, `UNIX_TIMESTAMP(datefield)` would be `$row['UNIX_TIMESTAMP(datefield)']`
  - `\n` stands for a new line so that your source code will look a little neater:
  - PHP MySQL functions @
    - url: <http://www.php.net/manual/en/ref.mysql.php>



# Using PHP to Query an MS Access Database

- ▶ PHP does not provide a connection library for MS Access or MS SQL Server
  - Only for MySQL, Oracle
  - All other databases are accessed through ODBC
- ▶ ODBC: Open Database Connectivity is a standard method of connecting an application or system to a database.
  - Most database vendors provide ODBC drivers so that you can use ODBC as a method of connecting to and querying their database.
- ▶ DSN: Data Source Name is a joining point between the database server and any application wishing to query the database.
  - Programs wishing to connect to and query a database using ODBC can reference this DSN

# Using PHP to Query an MS Access Database

```
<?php

//connect to database
$conn=odbc_connect('northwind','test','test2');

//SQL query
$sql="SELECT * FROM customers";

//Get the result-set into $rs
$rs=odbc_exec($conn,$sql);

//Return the value of the first field for the current record
$compname=odbc_result($rs,1);

//Return the value of the field called "CompanyName"
$compname=odbc_result($rs,"CompanyName");

//disconnect from database
odbc_close($connectionstring);

?>
```

# Using PHP to Query an MS Access Database

```
<html><body>
  <?php
    $conn=odbc_connect('northwind','');
    if (!$conn) { exit("Connection Failed: " . $conn);
    }
    $sql="SELECT * FROM customers";
    $rs=odbc_exec($conn,$sql);
    if (!$rs)
    {
      exit("Error in SQL");
    }
    echo "<table><tr>";
    echo "<th>Companyname</th>";
    echo "<th>Contactname</th></tr>";

    while (odbc_fetch_row($rs))
    {
      $compname=odbc_result($rs,"CompanyName");
      $conname=odbc_result($rs,"ContactName");
      echo "<tr><td>$compname</td>";
      echo "<td>$conname</td></tr>";
    }
    odbc_close($conn);
    echo "</table>";
  ?>
</body></html>
```

# Using PHP to insert records in MS Access Database

```
<?php
```

```
//connect to database
```

```
$conn=odbc_connect('northwind','');
```

```
//SQL query
```

```
$sql="INSERT INTO Customers (FName, LName, Address) Values ('Andreas', 'Protopapas', '1821 Nicosia')";
```

```
//Execute SQL
```

```
$updateresult=odbc_exec($conn,$sql);
```

```
//disconnect from database
```

```
odbc_close($conn);
```

```
?>
```



# PHP & ODBC Resources

## ▶ Web Sites

- <http://www.php.net/> – Manual is available here
- <http://www.zend.com> – PHP 4–5 engine
- <http://www.phpbuilder.com/>
- <http://www.devshed.com/>
- <http://www.phpmyadmin.net/>
- <http://www.hotscripts.com/PHP/>
- <http://www.mysql.com/>
- <http://www.owasp.org/>
- <http://www.zend.com/zend/tut/odbc.php>

## ▶ Google Search

## ▶ Books

- PHP and MySQL Web Development 2nd Edition, Welling & Thomson
- Web Database Applications with PHP & MySQL, O'Reilly Publishers
- PHP Cookbook, O'Reilly Publishers