

Introduction to PHP

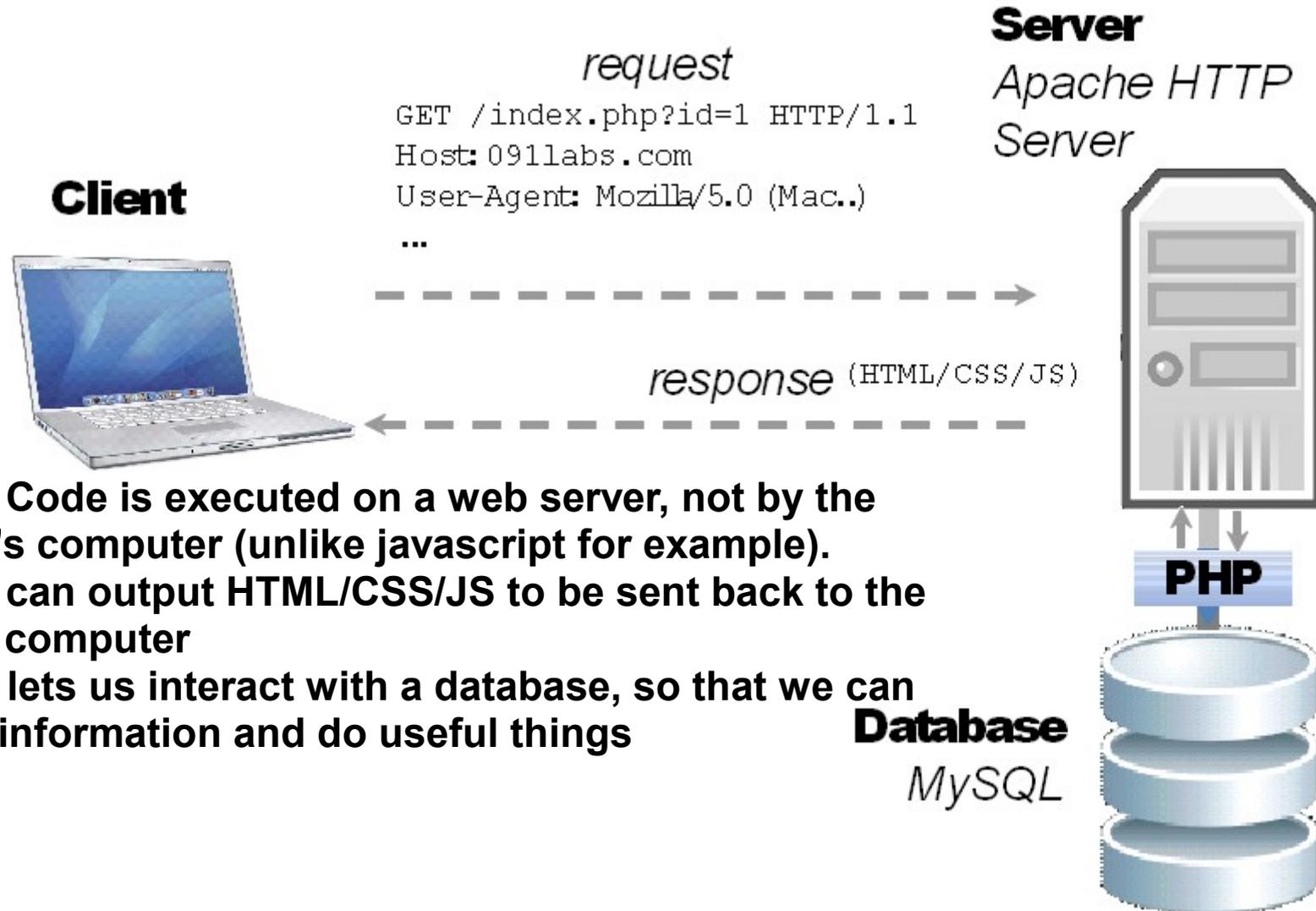
091 Labs

22rd October 2010

Outline: Week 1

- What is PHP?
- Setting up a development environment:
 - > XAMPP
 - > XDebug
 - > Eclipse
- “Hello world!”
- Variables and arrays
- Taking user input: GET/POST requests
- Creating a database via phpMyAdmin
- Connecting to the database server in PHP
- Getting our data into the database
 - > SQL injection
- Listing the database contents

PHP is a backend language



- PHP Code is executed on a web server, not by the client's computer (unlike javascript for example).
- PHP can output HTML/CSS/JS to be sent back to the client computer
- This lets us interact with a database, so that we can store information and do useful things

PHP mixes with HTML/CSS/JS

```
<html>
<head>
<title> Hello World </title>

<h1>Hello <?php echo $_GET[ 'name' ] ?>!</h1>

</head>
</html>
```



Hello 091 Labs!

Setting up a development environment:

- **XAMPP**

An all-in-one web stack. We need a local web server to test our PHP code. Instead of setting up Apache (web server), PHP and MySQL separately, we can use XAMPP to make it easy.

The address of our local web server will be **http://localhost**

- **Xdebug**

A debugger lets us step through code line by line, and look at information about our code. Indispensable tool for PHP programmers.

- **Eclipse IDE**

Setting up XAMPP:

- Run the EXE installer
- Accept all default settings (do not install XAMPP in C:\program files, this will cause problems on Windows Vista and 7)... selecting C:\ as an install path will install to C:\XAMPP
- A shortcut to “XAMPP Control Panel” will be created on your desktop. You can use this to start/stop Apache/MySQL

Setting up Eclipse:

Eclipse is standalone software, and requires a Java Runtime Environment. You probably have one installed, if not you can get it here:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

- Extract the Eclipse PHP zip file to C:\ (so that Eclipse is in C:\eclipse)
- Create a shortcut to C:\eclipse\eclipse.exe on your desktop or quick launch bar

Enabling Xdebug in XAMPP:

Uncomment and change the following lines in **C:\xampp\php\php.ini**

```
xdebug.remote_enable=1  
xdebug.remote_host="localhost"  
xdebug.remote_port=9000  
xdebug.remote_handler="dbgp"  
zend_extension="C:\xampp\php\ext\php_xdebug"
```

****You must restart Apache after making these changes****

Enabling XDebug in Eclipse:

Window->Preferences->PHP->Debug->PHP debugger->XDebug

Confirm that Xdebug is working:

- Open the XAMPP control Panel
- Click 'Start' beside Apache to start the web server
- Go to <http://localhost/xampp/phpinfo.php>
- Search for "xdebug". You should see something like this:

xdebug

xdebug support	enabled
Version	2.0.6-dev
Supported protocols	Revision

“Hello World”

Create the folder in the Document Root

- **C:\xampp\htdocs** is our “Document Root”. Create a folder in htdocs called “MyApp”.

Import the project into Eclipse

--> File->New->PHP Project

--> Create project from existing location (enter C:\xampp\htdocs\MyApp)

--> Click Finish

Create an index.php file

- Right-click the project in PHP Explorer, select New->PHP file, type **index.php**

- Open this file and type “<?php echo “Hello World”; ?>

- It can then be accessed at the URL <http://localhost/MyApp/>

***“index.php” is the default filename if no filename is specified, so**

<http://localhost/MyApp/> defaults to <http://localhost/MyApp/index.php>

****Note: Windows is not case-sensitive but most web servers are UNIX based and don't consider “myapp” and “MyApp” the same****

Variables in PHP

- PHP is a “**loosely typed**” language. This means you do not have to specify what kind of data a variable holds, e.g:

Java: String myStr = new String(“Hello World”);
PHP: \$myStr = “Hello world”;

PHP figures out what the data type is. We can even do this:

```
$myVar = 0; // $myVar is an integer with value 0  
$myVar .= "91 Labs"; // Add '91 Labs' to the end of $myVar  
echo $myVar;
```

This prints out “091 Labs”, PHP automatically converted \$myVar from an integer to a string.

- This makes PHP a great beginners language for learning the concepts of programming without worrying about memory allocation etc.
- The main scalar data types are:
 - > Integers (e.g. 0,1,2,3)
 - > Float (e.g. 7.24, 3.1234)
 - > String (e.g. “Hello World”, “091 Labs”)
 - > Boolean (True or False)

Arrays in PHP

There are two types of array in PHP:

Numerical Array:

```
$employee[0] = "Bob";  
$employee[1] = "John";  
$employee[2] = "Jack";  
$employee[3] = "Jill";
```

No need to declare the array or the size of the array unlike most other languages

Associative Array:

```
$a = array();  
$a['color'] = 'red';  
$a['taste'] = 'sweet';  
$a['shape'] = 'round';  
$a['name'] = 'apple';
```

Maps a key to a value

No need to worry about memory allocation, but to remove elements for efficiency, use **`unset($a['color']);`**

Taking user input: GET/POST

There are a few ways to pass data from the client to the server.

You have probably seen a URL like `http://.../myscript.php?id=1&a=2&message=hello`

This is a **GET** request.

PHP will give you an **associative array** of the user data called `$_GET`. So:

```
$_GET['id'] = 1  
$_GET['a'] = 2  
$_GET['message'] = "hello";
```

However there are some times when you don't want all of this data to show up in the URL, e.g. When taking in passwords or uploading files.

For this case, HTML lets us submit "POST" requests from forms. These are accessed in the same way, but using the `$_POST` variable.

Resources

- **php.net** – Contains the PHP manual, user-contributed examples and other documentation related to PHP.
- **phpfreaks.com** (tutorials)
- **tizag.com** (tutorials)
- **stackoverflow.com** – Best site on the web for asking questions about programming
- **boards.ie/091labs** – I'll try my best to answer your questions