

# Introduction to Hypertext Preprocessor (PHP)

# Introduction to PHP

- PHP – A technology for developing web pages that include dynamic content
- “Equivalent” of Microsoft’s Active Server Pages technology
- Pages have a file extension of .PHP
- PHPs enable web developers to enhance HTML code by adding special PHP elements that are processed by the server
- PHP is a free software released under the PHP License
- PHP Scripts are executed on the server



# What is a PHP file

- PHP stands for PHP: Hypertext Preprocessor
- PHP supports many databases ( MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC, etc.)
- PHP is an open source software (OSS)
- This means its free to use and isn't being controlled by a single entity.
- It is being developed by a group of developers
- PHP syntax resembles that of JavaScript and ActionScript in different ways.

# What can PHP do

- PHP can generate dynamic page content
- PHP can create, open, read, write, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can restrict users to access some pages on your website
- PHP can encrypt data



# What can PHP do

- PHP files are returned to the browser as plain HTML.
- PHP can be written in any text editor.
- PHP script will be located inside special tags, much like JavaScript  
e.g. `<?php //php script here ?>`
- PHP code can be located any where in the page.
- PHP is case sensitive.
- Every variable in PHP will have the \$ symbol as a prefix e.g. `$myName = "John";`
- Every line of code MUST be terminated with a ; symbol

# What is a MySQL

- MySQL is a small database server
- MySQL is ideal for small and medium applications
- MySQL supports standard SQL
- MySQL compiles on a number of platforms
- • MySQL is free to download and use



# PHP + MySQL

- PHP combined with MySQL is cross-platform (means that you can develop in Windows and serve on a Unix platform).

# Basic PHP Syntax

- You cannot view the PHP source code by selecting "View source" in the browser – you will only see the output from the PHP file, which is plain HTML.
- A PHP scripting block always starts with `<?php` and ends with `?>`.
- A PHP scripting block can be placed anywhere in the document.

```
<?php
```

```
?>
```



# Basic PHP Syntax

- A PHP file normally contains HTML tags, just like an HTML file, and some PHP scripting code.
- Below, we have an example of a simple PHP script which sends the text
- “Hello World” to the browser:

```
<html>
<body>

<?php
echo "Hello World";
?>

</body>
</html>
```

- Each code line in PHP must end with a semicolon. The semicolon is a separator and is used to distinguish one set of instructions from another.
- There are two basic statements to output text with PHP: echo and print. In the example above we have used the echo statement to output the text “Hello World”.

# Comments in PHP

- In PHP, we use `//` to make a single-line comment or `/*` and `*/` to make a large comment block.

```
<html>
<body>

<?php

//This is a comment

/*
This is
a comment
block
*/

?>

</body>
</html>
```



# PHP Variables

- Variables are used for storing a values, like text strings, numbers or arrays.
- When a variable is set it can be used over and over again in your script
- All variables in PHP start with a \$ sign symbol.
- The correct way of setting a variable in PHP:

```
$var_name = value;
```

# PHP Variables

- New PHP programmers often forget the \$ sign at the beginning of the variable. In that case it will not work.
- Let's try creating a variable with a string, and a variable with a number:

```
<?php  
  
$txt = "Hello World!";  
$number = 16;  
  
?>
```



# PHP is a Loosely Typed Language

- In PHP a variable does not need to be declared before being set.
- In the previous example, you see that you do not have to tell PHP which data type the variable is. PHP automatically converts the variable to the correct data type, depending on how they are set.
- In a strongly typed programming language, you have to declare (define) the type and name of the variable before using it.
- In PHP the variable is declared automatically when you use it.

# Variable Naming Rules

- A variable name must start with a letter or an underscore “\_”
- A variable name can only contain alpha-numeric characters and underscores (a-Z, 0-9, and \_)
- A variable name should not contain spaces. If a variable name is more than one word, it should be separated with underscore (\$my\_string), or with capitalization (\$myString)



# Strings In PHP

- String variables are used for values that contains character strings.
- We are going to look at some of the most common functions and operators used to manipulate strings in PHP.
- After we create a string we can manipulate it. A string can be used directly in a function or it can be stored in a variable.
- Below, the PHP script assigns the string "Hello World" to a string variable called \$txt:

```
<?php  
  
$txt="Hello World";  
echo $txt;  
  
?>
```

# The Concatenation Operator

- There is only one string operator in PHP.
- The concatenation operator (.) is used to put two string values together.

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

- To concatenate two variables together, use the dot (.) operator:
- If we look at the code above you see that we used the concatenation operator two times. This is because we had to insert a third string. Between the two string variables we added a string with a single character, an empty space, to separate the two variables.



# Strings In PHP

➤The output of the code will be:

```
Hello World 1234
```

# Using the strlen() function

- The strlen() function is used to find the length of a string.
- Let's find the length of our string "Hello world!":

```
<?php  
  
echo strlen("Hello world!");  
  
?>
```

- The output of the code above will be:

```
12
```

- The length of a string is often used in loops or other functions, when it is important to know when the string ends. (i.e. in a loop, we would want to stop the loop after the last character in the string)



# Using the strpos() function

- The strpos() function is used to search for a string or character within a string.
- If a match is found in the string, this function will return the position of the first match.
- If no match is found, it will return FALSE.
- Let's see if we can find the string "world" in our string:

```
<?php  
echo strpos("Hello world!", "world");  
?>
```

- The output of the code above will be:

```
6
```

- As you see the position of the string "world" in our string is position 6. The reason that it is 6, and not 7, is that the first position in the string is 0, and not 1.

# Running PHP pages

- One way to develop and run PHP pages, need:
- If you wish to install full XAMPP (155mb) in your own computer, this is the link  
<https://www.apachefriends.org/index.html>
- If you wish to run PHP and MySQL in your window, you can watch Youtube, this is the link  
<https://www.youtube.com/watch?v=ERPytWypYzA>



# Install XAMPP



The screenshot shows a web browser window with the title "XAMPP Installers and Downloads". The address bar shows a secure connection to <https://www.apachefriends.org/index.html>. The page content includes a section titled "What is XAMPP?" which states that XAMPP is the most popular PHP development environment and is a free, easy-to-install Apache distribution containing MariaDB, PHP, and Perl. To the right of this text is a large orange square logo with a white play button icon and the word "XAMPP" below it. At the bottom of the page, there is a green "Download" button with the text "Click here for other versions" and three buttons for different operating systems: "XAMPP for Windows v5.6.30 (PHP 5.6.30)", "XAMPP for Linux v5.6.30 (PHP 5.6.30)", and "XAMPP for OS X v5.6.30 (PHP 5.6.30)".

**What is XAMPP?**

XAMPP is the most popular PHP development environment

XAMPP is a completely free, easy to install Apache distribution containing MariaDB, PHP, and Perl. The XAMPP open source package has been set up to be incredibly easy to install and to use.

**Download**  
Click here for other versions

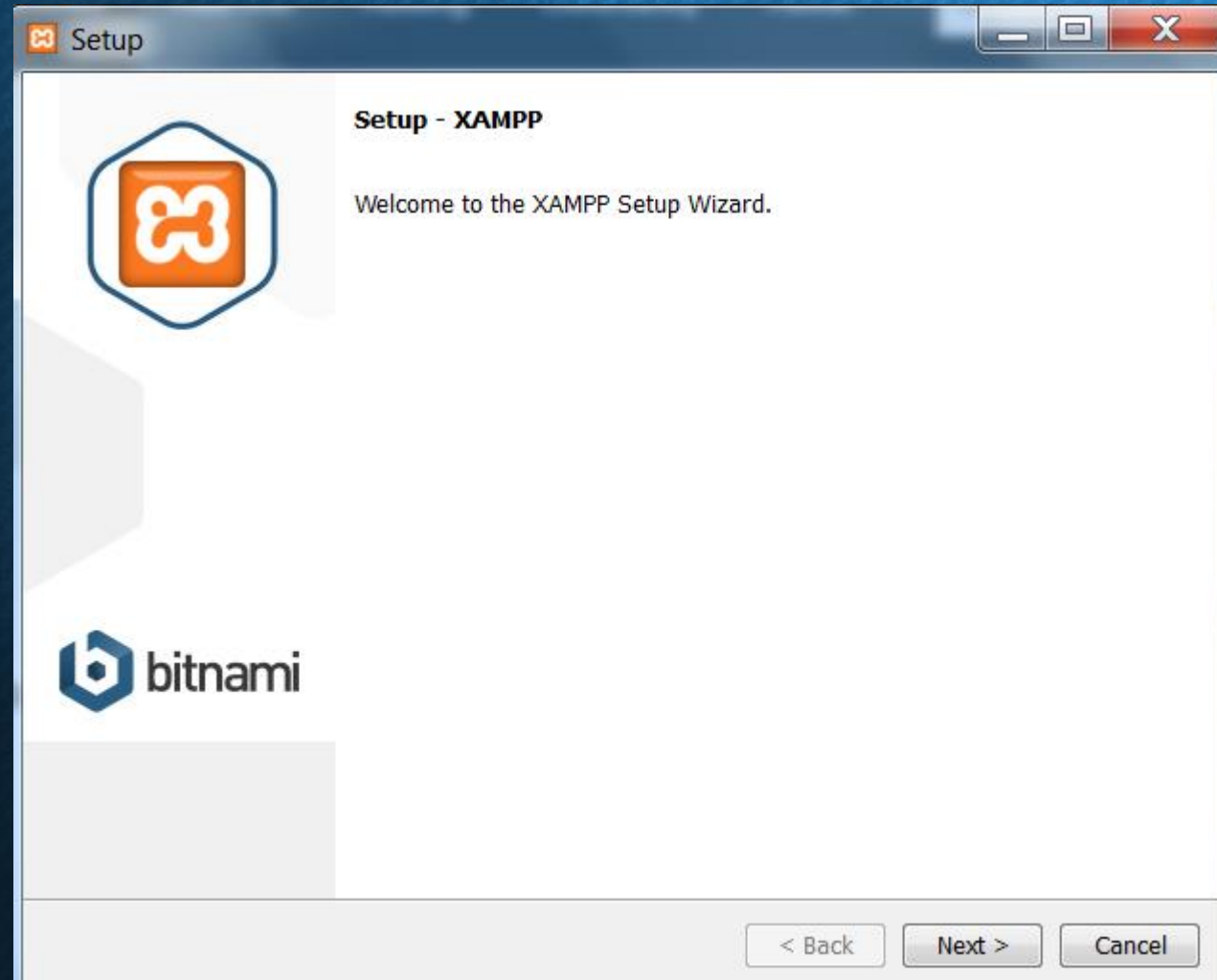
 **XAMPP for Windows**  
v5.6.30 (PHP 5.6.30)

 **XAMPP for Linux**  
v5.6.30 (PHP 5.6.30)

 **XAMPP for OS X**  
v5.6.30 (PHP 5.6.30)

# Install XAMPP

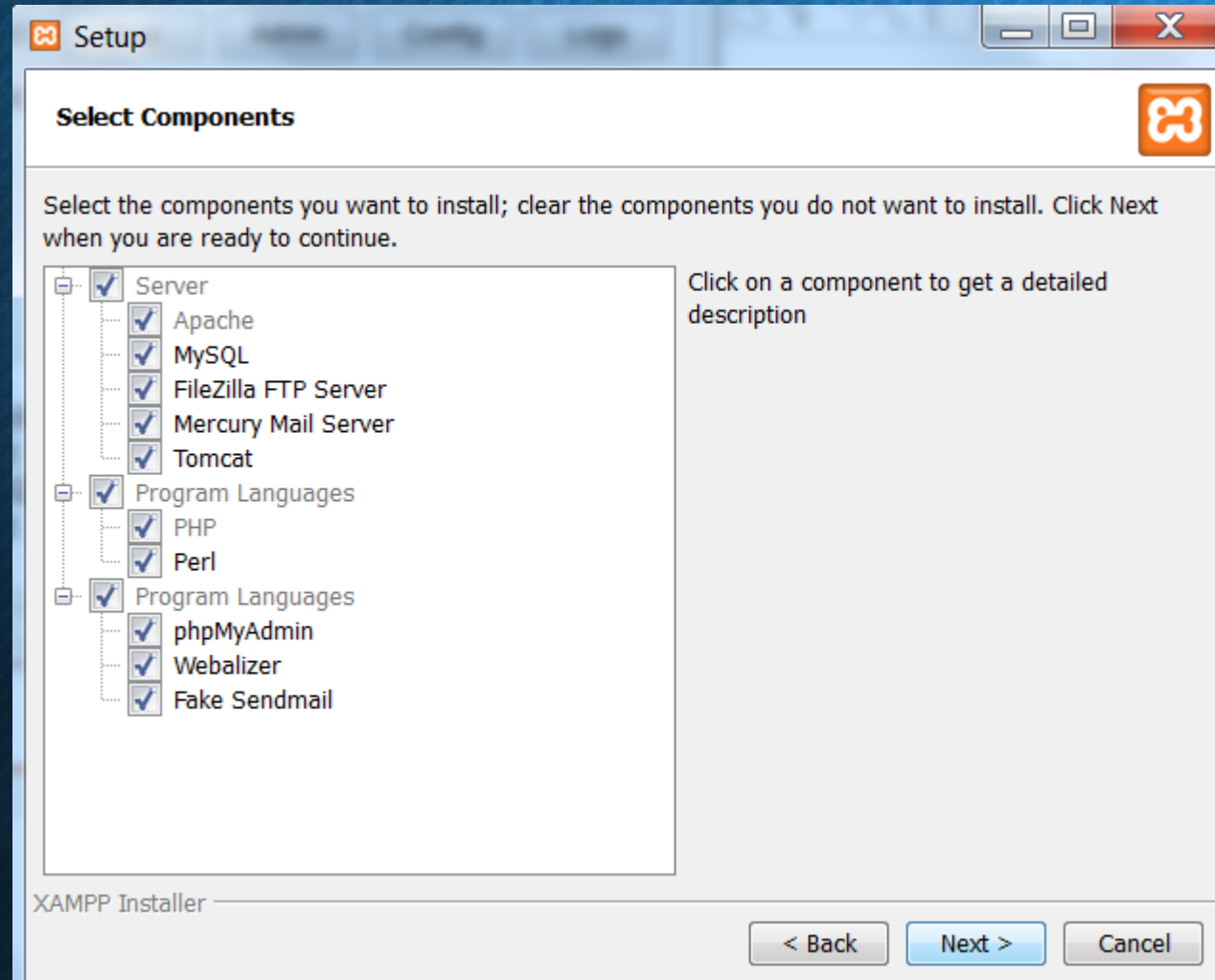
Select Next:





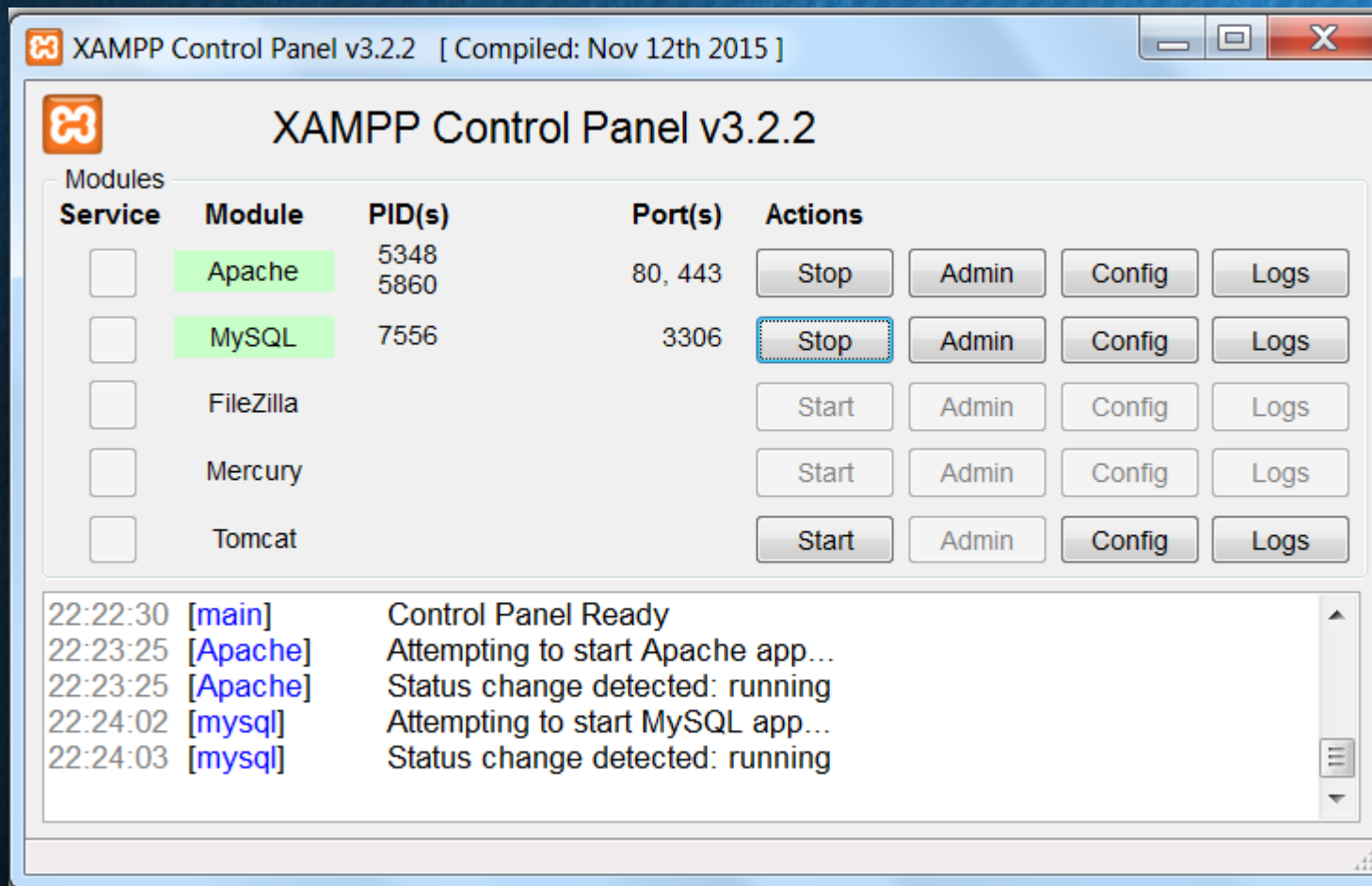
# Install XAMPP

Select Next:



# Start, stop, test XAMPP servers

The universal control center is the "XAMPP Control Panel". It is started with: `\xampp\xampp-control.exe`





# Start, stop, test XAMPP servers

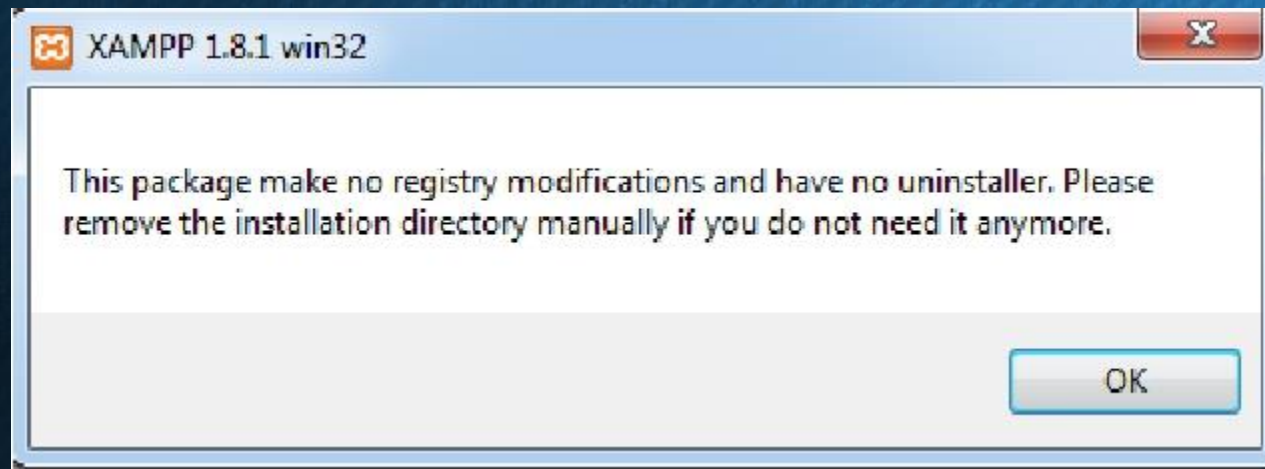
- After starting of Apache (and MySQL), go to the address `http://localhost/` or `http://127.0.0.1/` in your browser , you should see the home page



- Write your first “helloworld.php” and place it inside `xampp/htdocs`, or it even better to create your own WebD folder inside `htdocs`.
- Go to the address, You should see your first PHP page!

## Small tips

- A very useful information for “uninstall” XAMPP



- Once the XAMPP Portable is installed, the Apache, MySQL and PHP are ready to use.