

LECTURE 1

INT 220(SERVER SIDE SCRIPTING)

Introduction

- PHP stands for Hypertext Preprocessor.
- It is a very popular and widely-used open source server-side scripting language to write dynamically generated web pages.
- It was originally created by Rasmus Lerdorf in 1994.
- It was initially known as Personal Home Page.
- PHP is a server-side scripting language that is embedded in HTML.



Why Learn PHP?

PHP one of the most preferred languages for creating interactive websites and web applications. PHP scripts can be easily embedded into HTML.

With PHP, you can build

- Web Pages and Web-Based Applications
- Content Management Systems, and
- Ecommerce Applications etc.

A number of PHP based web frameworks have been developed to speed-up the web application development. The examples are WordPress, Laravel, Symfony etc.

Introduction(contd.)

Versions:

- PHP Version 1.0
- PHP Version 2.0
- PHP Version 3.0
- PHP Version 4.0
- PHP Version 5.0
- PHP Version 6.0
- **PHP Version 7.0**
- PHP Version 8.0 -> 8.3

For more information, check :

<https://php.watch/versions>

Is PHP compiled or interpreted?

Introduction(contd.)

- A PHP interpreter, which can be implemented as a module, is commonly used to process PHP code on a web server.
- The outcome of the interpreted and executed PHP code – which is an HTML data, CSS, JavaScript, images – would make up the entirety or portion of an HTTP response on a web server.
- PHP scripts are executed on the server and the result is sent to the web browser as plain HTML.
- PHP can be integrated with the number of popular databases, including MySQL, Oracle, Microsoft SQL Server, Sybase, and so on.
- PHP files have extension ".php"

Introduction(contd.)

- ele"PHP"ant



What You Can Do with PHP

- You can generate pages and files dynamically.
- You can create, open, read, write and close files on the server.
- You can collect data from a web form such as user information, email, phone no, etc.
- You can send emails to the users of your website.
- You can send and receive cookies to track the visitor of your website.
- You can store, delete, and modify information in your database.
- You can restrict unauthorized access to your website.
- You can encrypt data for safe transmission over internet.

Advantages of PHP over Other Languages

- **Easy to learn:** PHP is easy to learn and use. For beginner programmers who just started out in web development, PHP is often considered as the preferable choice of language to learn.
- **Open source:** PHP is an open-source project. It is developed and maintained by a worldwide community of developers who make its source code freely available to download and use.
- **Portability:** PHP runs on various platforms such as Microsoft Windows, Linux, Mac OS, etc. and it is compatible with almost all servers used today such as Apache.

Advantages of PHP over Other Languages(contd.)

- **Fast Performance:** Scripts written in PHP usually execute or runs faster than those written in other scripting languages like ASP, Ruby, Python, Java, etc.
- **Vast Community:** Since PHP is supported by the worldwide community, finding help or documentation related to PHP online is extremely easy.

Setting Up a Local Web Server

- PHP script execute on a web server running PHP. So before you start writing any PHP program you need the following program installed on your computer.
 - The Apache Web server
 - The PHP engine
 - The MySQL database server
- You can either install them individually or choose a pre-configured package for your operating system like Linux and Windows. Popular pre-configured package is XAMPP

Setting Up a Local Web Server(contd.)

- XAMPP is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for developing locally php application for testing purposes.
- Everything that you need for developing an application like a web server - server application (Apache), database (MySQL), and scripting language (PHP)

Setting Up a Local Web Server(contd.)

- **XAMPP** is a software distribution which provides the Apache web server, PHP and MySQL database
- This software is mainly used to test the websites locally.
- It provides the MySQL administrative tool PhpMyAdmin to easily manage your databases using a web browser.

<https://www.apachefriends.org/download.html>

Advantages of XAMPP

- It is **free and easy** to use and easily available for Windows, Linux and Mac OS .
- It is a **beginners friendly solution package** for full stack web development.
- It is a **open source** software package which gives a easy installation experience.
- It is very **simple and lightweight** to create set up for development, testing and deployment.
- It is a **time-saver** and provides several ways for managing configuration changes.

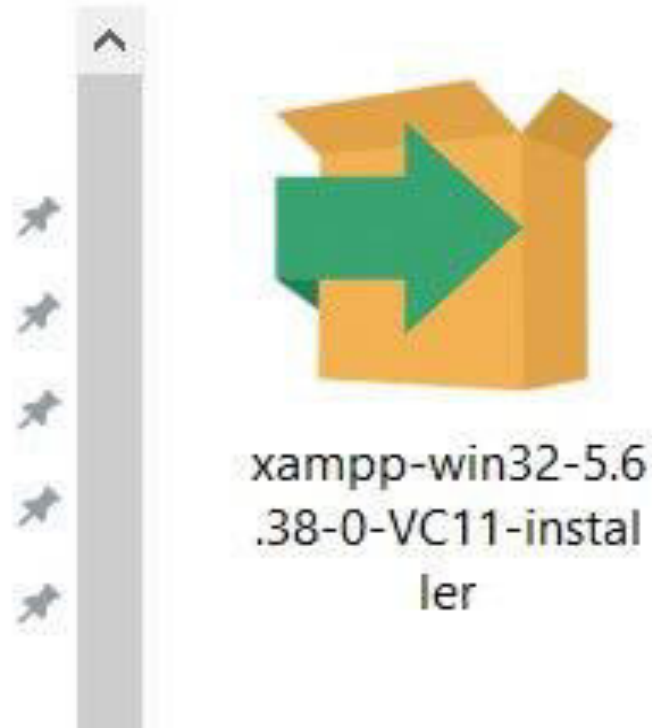
Setting Up a Local Web Server(contd.)



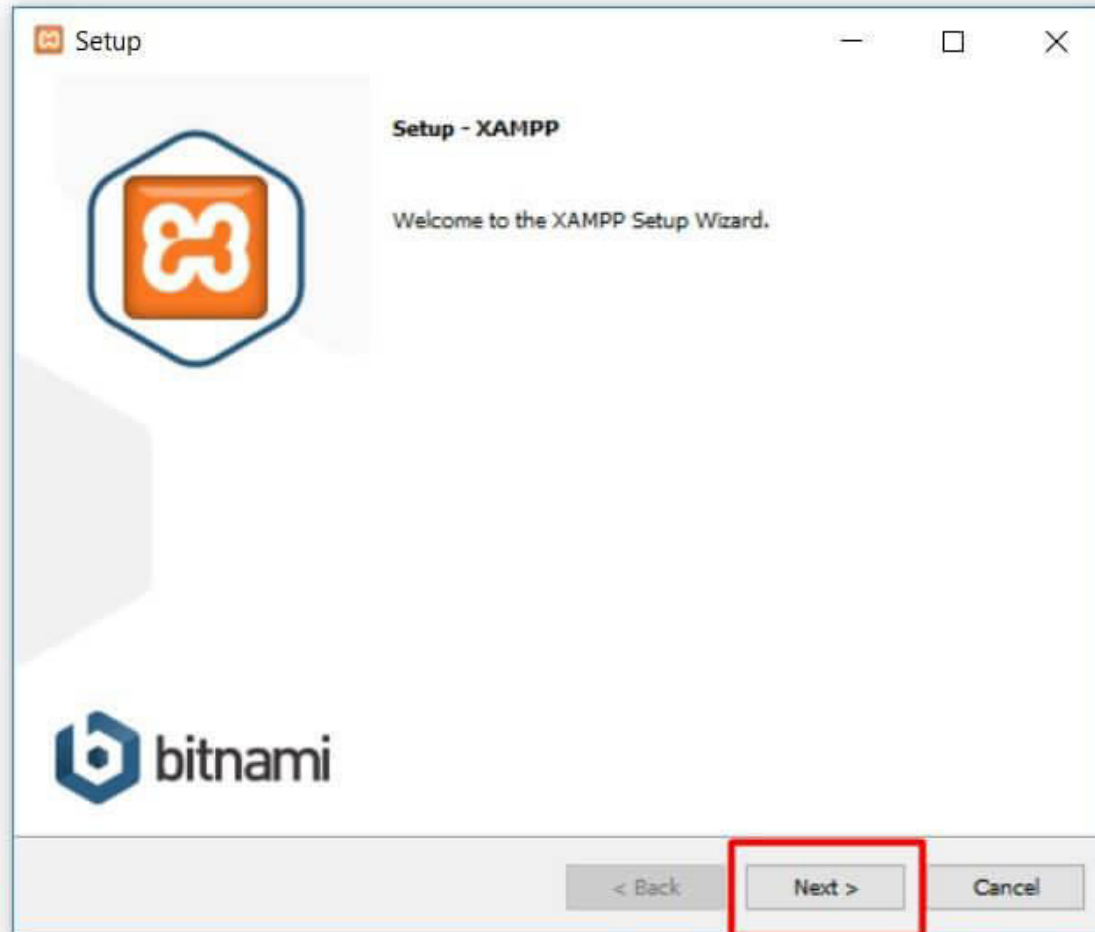
XAMPP for **Windows** 7.4.27, 8.0.14 & 8.1.1

Version		Checksum			Size
7.4.27 / PHP 7.4.27	What's Included?	md5	sha1	Download (64 bit)	160 Mb
8.0.14 / PHP 8.0.14	What's Included?	md5	sha1	Download (64 bit)	161 Mb
8.1.1 / PHP 8.1.1	What's Included?	md5	sha1	Download (64 bit)	164 Mb

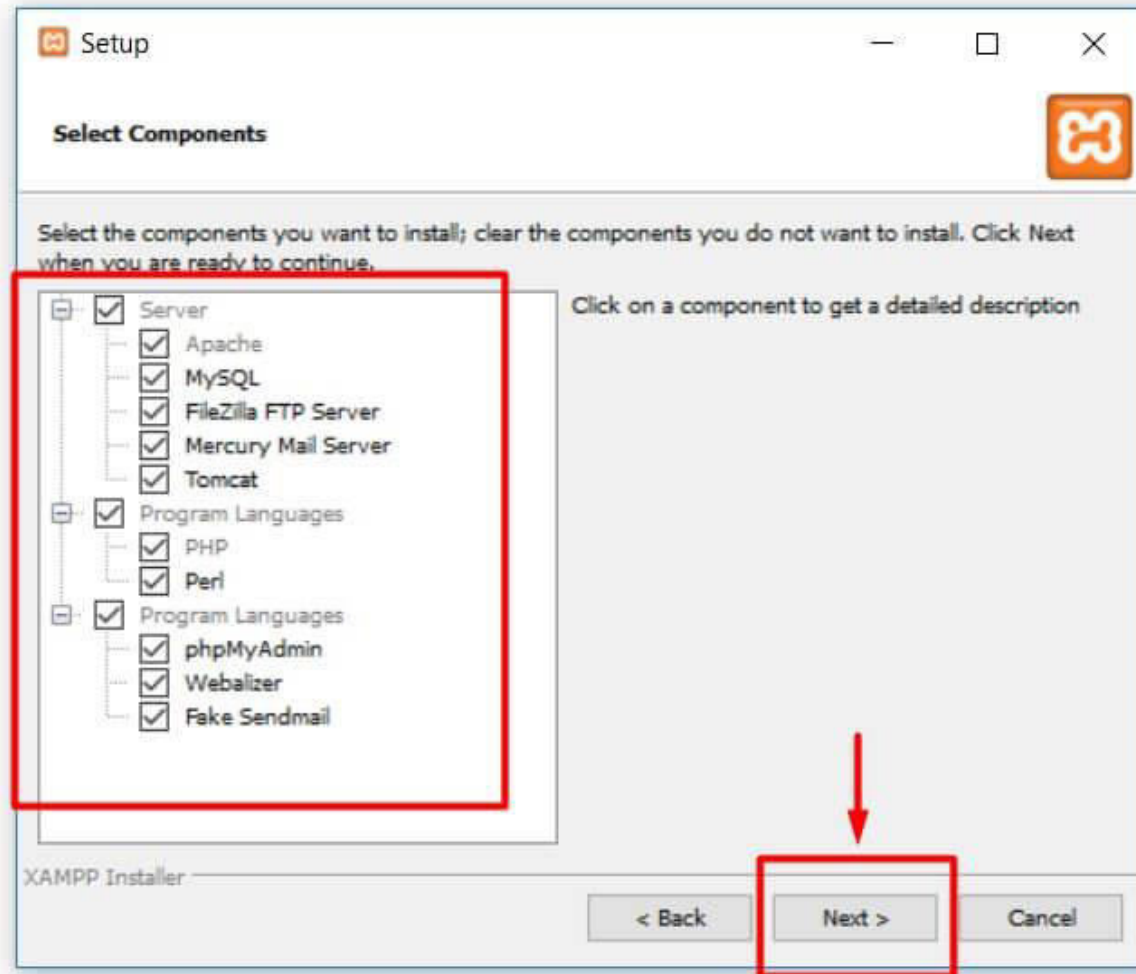
Setting Up a Local Web Server(contd.)



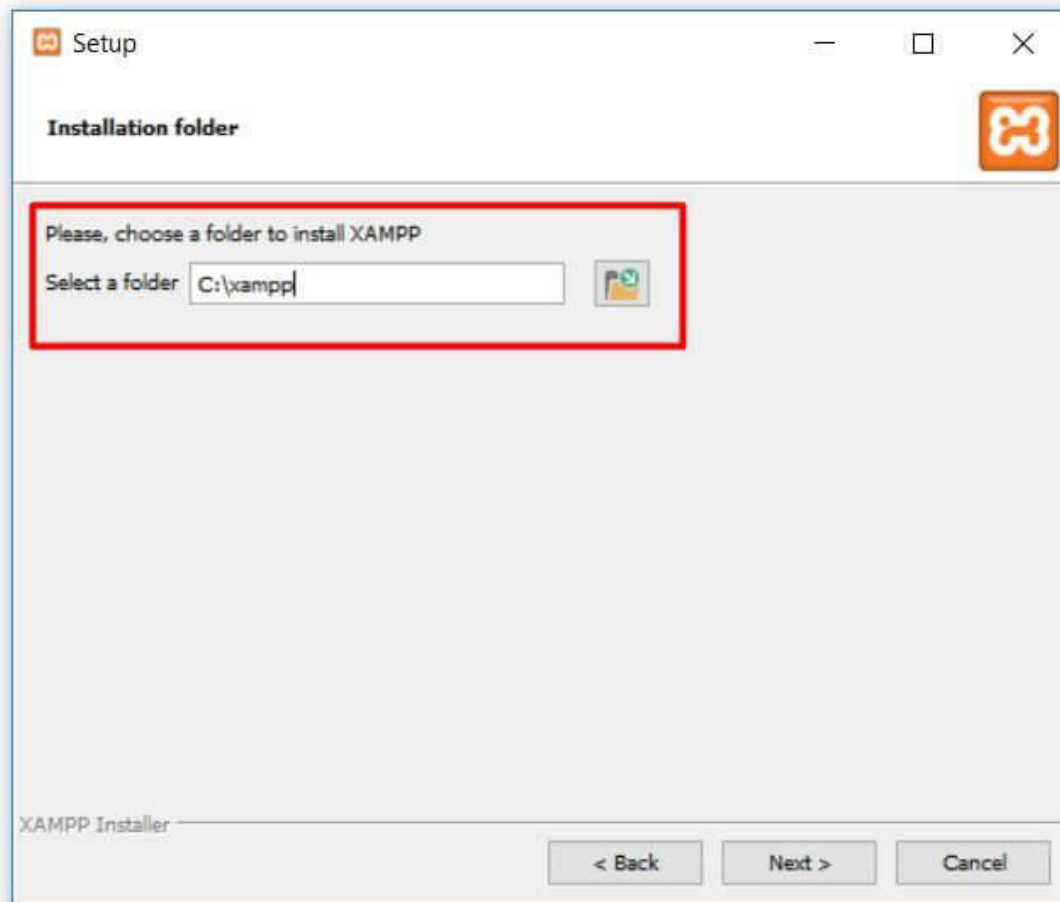
Setting Up a Local Web Server(contd.)



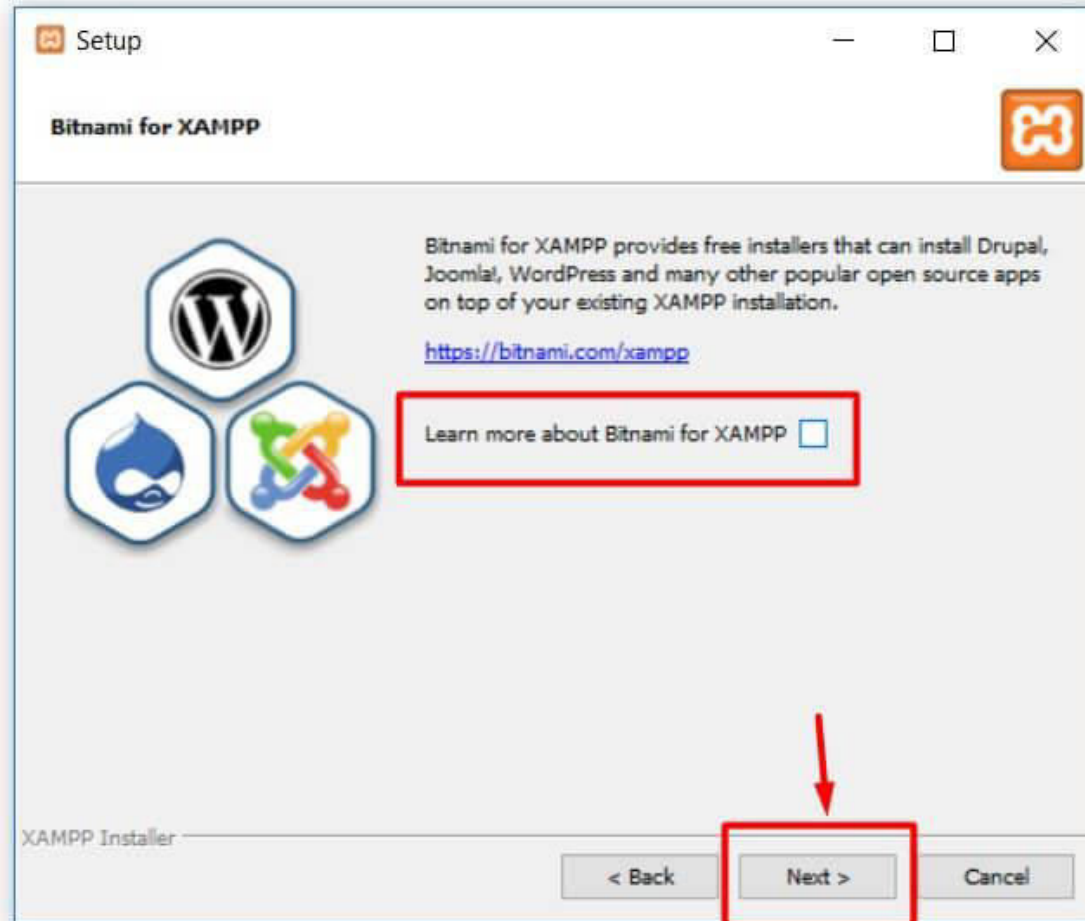
Setting Up a Local Web Server(contd.)



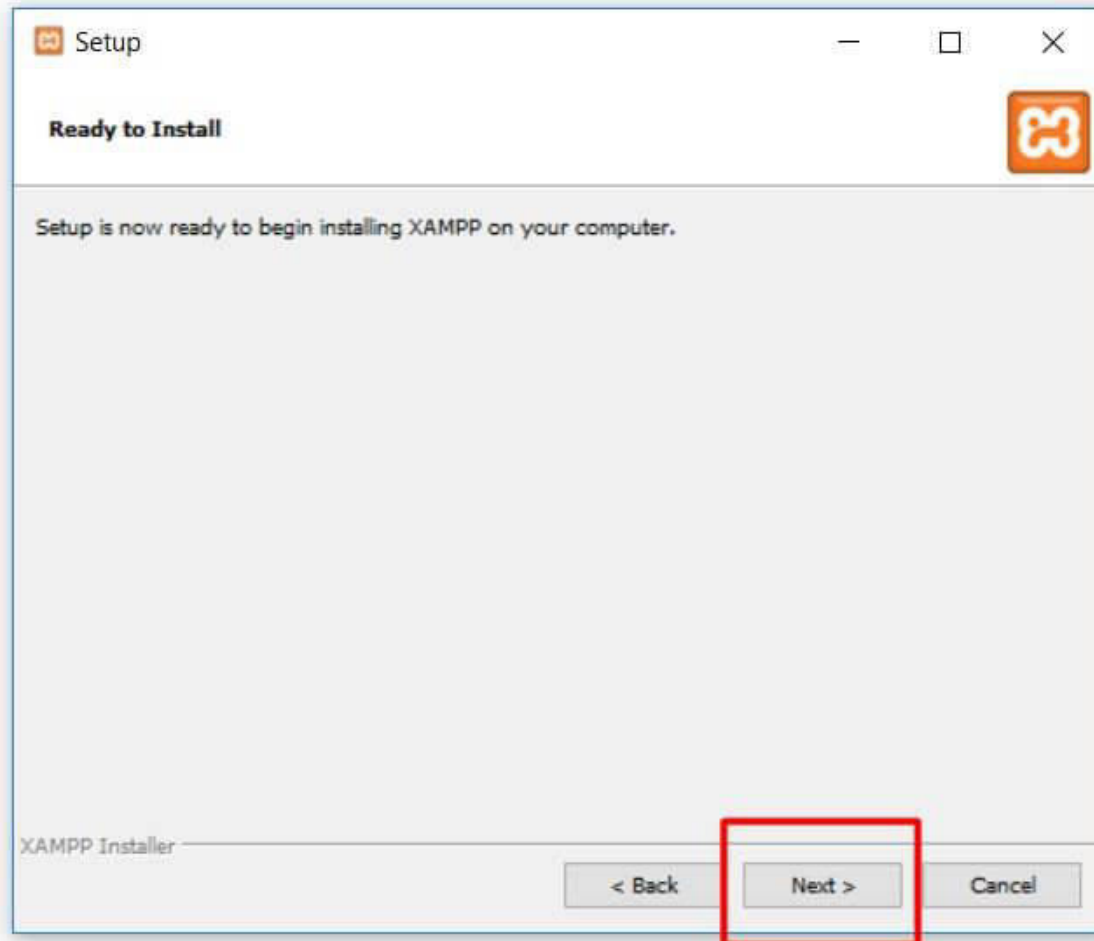
Setting Up a Local Web Server(contd.)



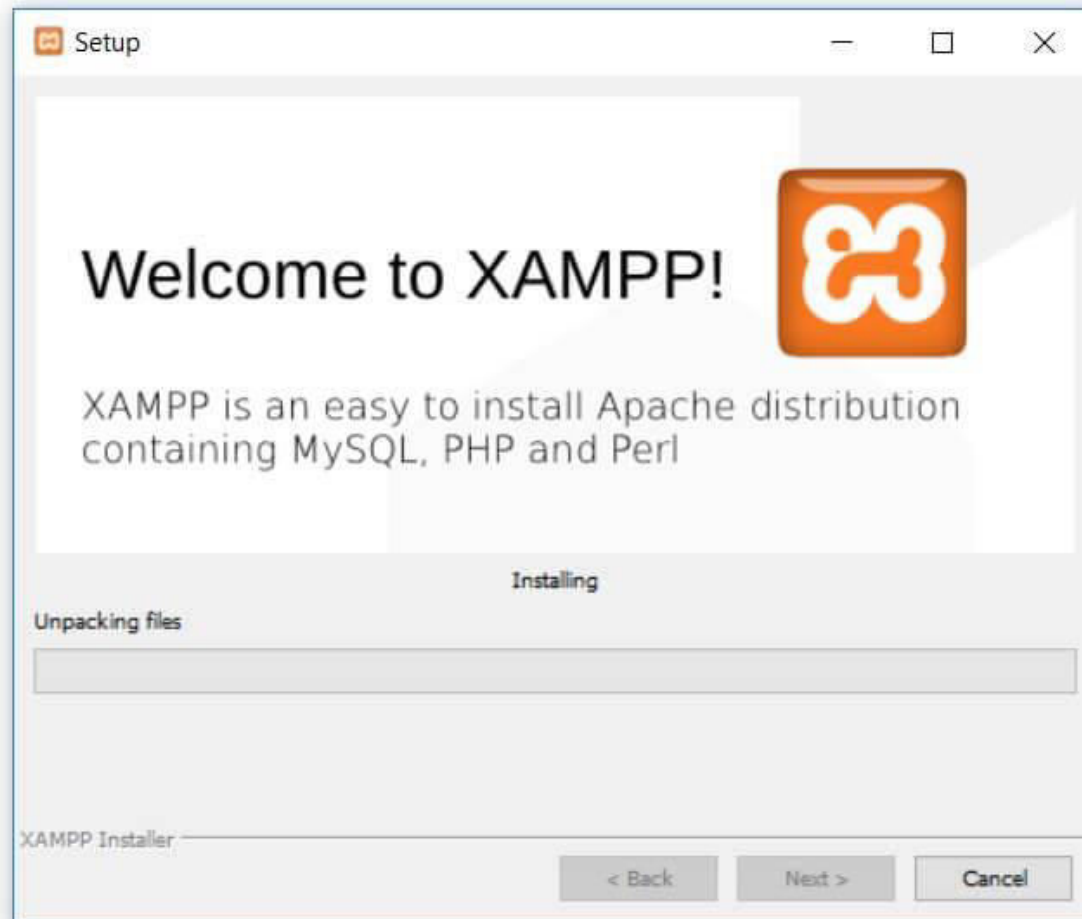
Setting Up a Local Web Server(contd.)



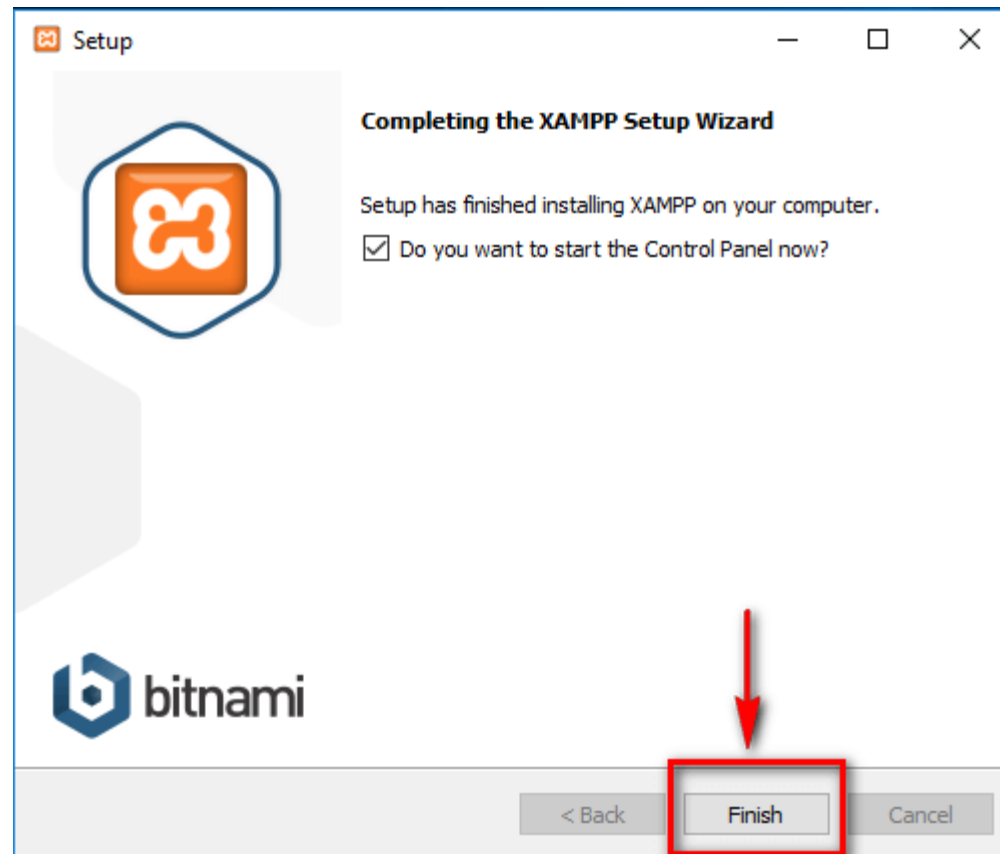
Setting Up a Local Web Server(contd.)



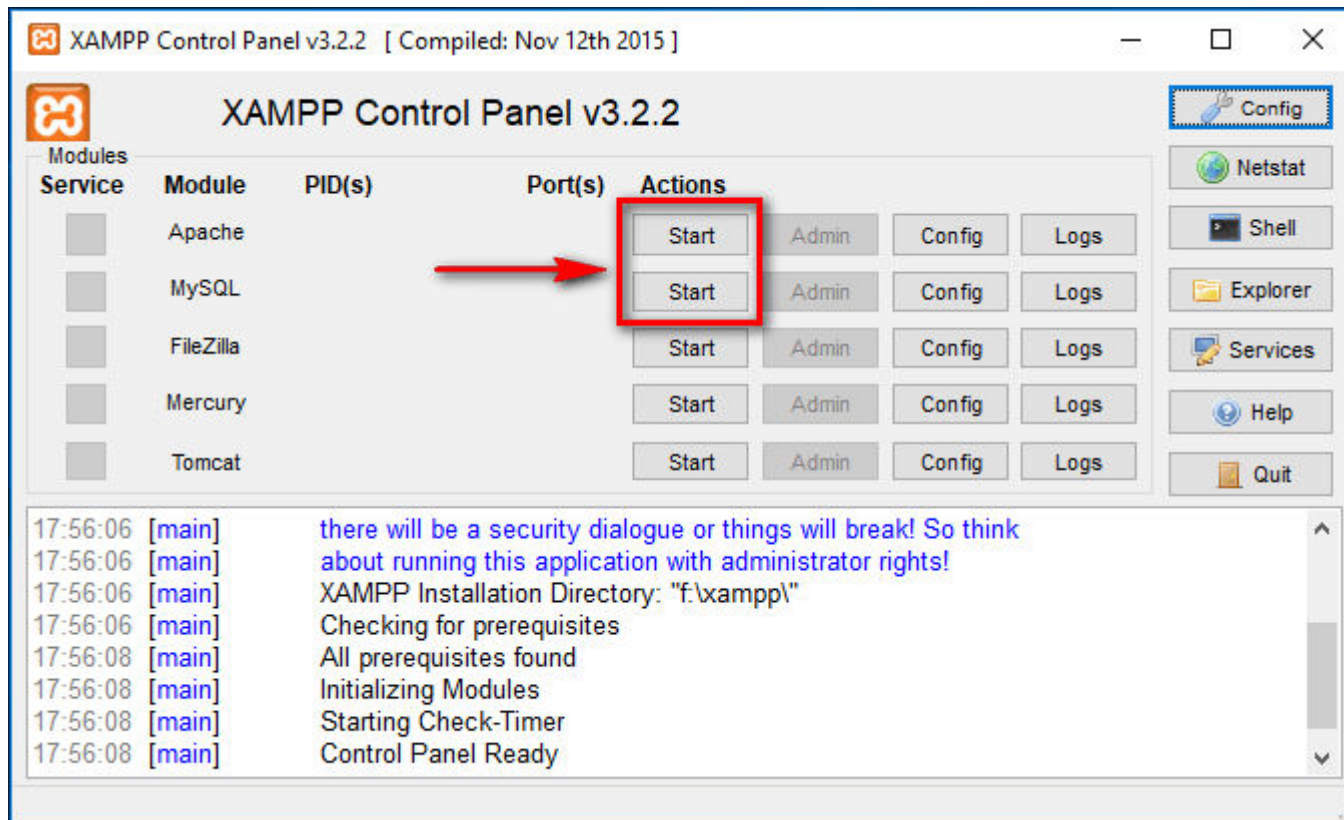
Setting Up a Local Web Server(contd.)



Setting Up a Local Web Server(contd.)



Setting Up a Local Web Server(contd.)



Setting Up a Local Web Server(contd.)

XAMPP Control Panel v3.2.2 [Compiled: Nov 12th 2015]

XAMPP Control Panel v3.2.2

Service	Module	PID(s)	Port(s)	Actions
<input type="checkbox"/>	Apache	12260 8824	80, 443	Stop Admin Config Logs
<input type="checkbox"/>	MySQL	11336	3306	Stop Admin Config Logs
<input type="checkbox"/>	FileZilla			Start Admin Config Logs
<input type="checkbox"/>	Mercury			Start Admin Config Logs
<input type="checkbox"/>	Tomcat			Start Admin Config Logs

Config Netstat Shell Explorer Services Help Quit

17:56:08 [main] Initializing Modules
17:56:08 [main] Starting Check-Timer
17:56:08 [main] Control Panel Ready
17:57:26 [Apache] Attempting to start Apache app...
17:57:26 [Apache] Status change detected: running
17:57:30 [mysql] Attempting to start MySQL app...
17:57:32 [mysql] Status change detected: running

Setting Up a Local Web Server(contd.)

The screenshot shows the XAMPP Control Panel v3.2.1 interface. The title bar indicates it was compiled on May 7th, 2013. The main area contains a table of modules and their status, along with a log of actions at the bottom. Red arrows point to various buttons and the log, with text labels explaining their functions.

Service	Module	PID(s)	Port(s)	Actions
<input type="checkbox"/>	Apache	7108 5728	80, 443	Stop Admin Config Logs
<input type="checkbox"/>	MySQL			Start Admin Config Logs
<input type="checkbox"/>	FileZilla			Start Admin Config Logs
<input type="checkbox"/>	Mercury			Start Admin Config Logs
<input type="checkbox"/>	Tomcat			Start Admin Config Logs

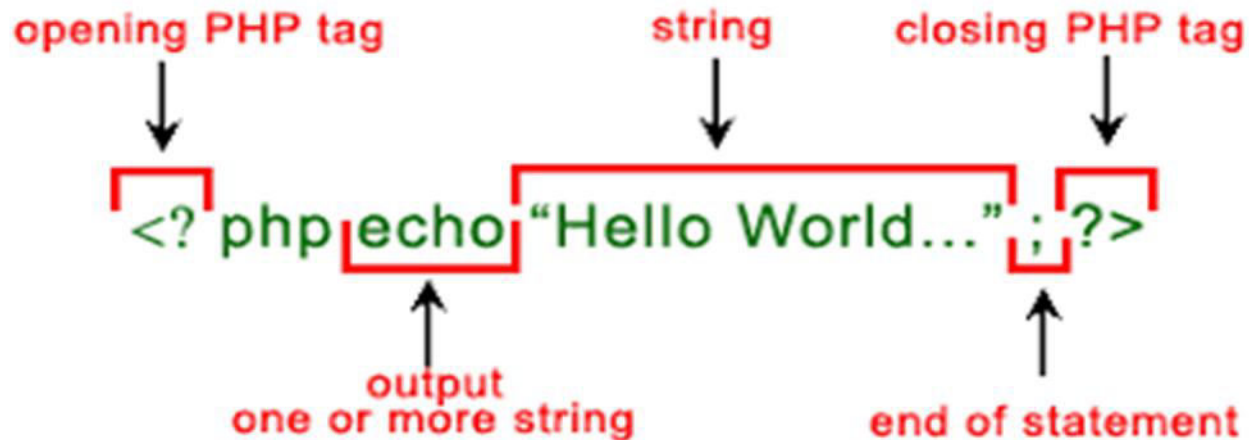
Annotations:

- Opens configuration panel:** Points to the **Config** button in the top right.
- Shows all processes running on server (local computer):** Points to the **Netstat** button in the top right.
- Opens UNIX shell:** Points to the **Shell** button in the top right.
- Opens Windows Explorer:** Points to the **Explorer** button in the top right.
- Shows all current services running in the background:** Points to the **Services** button in the top right.
- Used to start, stop and configure different XAMPP modules:** Points to the **Config** button in the Actions column of the Apache row.
- Logs all current actions:** Points to the log window at the bottom.

Log Window Content:

```
10:11:19 PM [main] Checking for prerequisites
10:11:52 PM [main] All prerequisites found
10:11:52 PM [main] Initializing Modules
10:11:52 PM [main] Starting Check-Timer
10:11:52 PM [main] Control Panel Ready
10:12:54 PM [Apache] Attempting to start Apache app...
10:12:55 PM [Apache] Status change detected: running
10:16:39 PM [main] Executing "c:\xampp\"
10:16:45 PM [main] Executing "services.msc"
```

Basic Syntax



PHP tags

- When PHP parses a file, it looks for opening and closing tags, which are `<?php` and `?>` which tell PHP to start and stop interpreting the code between them.
- Parsing in this manner allows PHP to be embedded in all sorts of different documents, as everything outside of a pair of opening and closing tags is ignored by the PHP parser.
- PHP includes a short echo tag `<?=' which is a short-hand to the more verbose <?php echo.`

PHP tags(contd.)

- Using short tags should be avoided when developing applications or libraries that are meant for redistribution, or deployment on PHP servers which are not under your control, because short tags may not be supported on the target server.
- For portable, redistributable code, be sure not to use short tags.

PHP tags(contd.)

Recommended tags

Standard tag:

```
<?php echo 'hello'; ?>
```

Short echo tag:

```
<?= 'hello'; ?>
```

NOTE: If file contains only php code, you can exclude the closing tag

PHP tags(contd.)

```
<?php  
    echo 'This is a test';  
?>
```

```
<?php echo 'This is a test' ?>
```

```
<?php echo 'We omitted the last closing tag';
```

Escaping from HTML

- Everything outside of a pair of opening and closing tags is ignored by the PHP parser which allows PHP files to have mixed content.
- This allows PHP to be embedded in HTML documents, for example to create templates.

<p>This is going to be ignored by PHP and displayed by the browser.</p>

<?php echo 'While this is going to be parsed.'; ?>

<p>This will also be ignored by PHP and displayed by the browser.</p>

PHP Comments

- A comment is simply text that is ignored by the PHP engine.
- The purpose of comments is to make the code more readable.
- It may help other developer (or you in the future when you edit the source code) to understand what you were trying to do with the PHP.
- PHP support single-line as well as multi-line comments. To write a single-line comment either start the line with either two slashes (//) or a hash symbol (#).

PHP Comments(contd.)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>PHP Comments</title>
</head>
<body>

<?php
// This is a single line comment
# This is also a single line comment
    echo 'Hello World!';
?>

</body>
</html>
```

PHP Comments(contd.)

- However to write multi-line comments, start the comment with a slash followed by an asterisk (/*) and end the comment with an asterisk followed by a slash (*/), like this:

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
    <title>PHP Comments</title>
```

```
</head>
```

```
<body>
```

```
<?php
```

```
/*
```

```
This is a multiple line comment block  
that spans across more than  
one line
```

```
*/
```

```
echo "Hello, world!";
```

```
?>
```

```
</body>
```

```
</html>
```

Case Sensitivity in PHP

Variable names in PHP are case-sensitive.

PHP echo vs print

- echo has no return value while print has a return value of 1
- echo can take multiple parameters (although such usage is rare) while print can take one argument.
- echo is marginally faster than print.