

# Getting Set Up with PHP

Setup tutorial and all materials at:

[https://github.com/sketchings/  
beginning-php-training](https://github.com/sketchings/beginning-php-training)

# About PHP

## Why PHP?

- ▷ It's the easiest and cheapest to deploy
- ▷ Low barrier to entry
- ▷ Duck typing (If it walks/talks/quacks like a duck...)
- ▷ Built for the web - current versions are *fast*
  - ▷ Built-in HTML templates
  - ▷ Easy access to server environment, cookies, sessions, form input
- ▷ Familiar C-like syntax (Perl, Java, C++, JavaScript)

## Why Not PHP?

- ▷ Evolved, not designed
  - ▷ Like English, lots of exceptions to rules and inconsistencies
  - ▷ No benevolent dictator
- ▷ PHP projects, over time, often show same tendency
  - ▷ Leave cleanup / technical debt for next iteration
  - ▷ Scrum backlog substitutes for design/architecture

## Why Not PHP?

- ▷ Built for the web, not for anything else
- ▷ Low barrier to entry
- ▷ Conveniences like duck typing create opportunities for insecure code and unexpected behavior that trips up the inexperienced

## Don't Worry...

- ▷ That's why we're here.

## Life of a PHP Script

1. Browser requests something from web server
2. Web server hands request to PHP
3. PHP interpreter does stuff (next slide)
4. Web server sends response to browser

## PHP Interpreter Handles Web Request

1. Web server fires up PHP interpreter
  1. Parse PHP script(s)
  2. Execute code
  3. Hand response to web server
  4. Shut completely down
2. Web server sends response to browser

## Life in the PHP Interpreter

- ▷ Web server passes request to PHP when it encounters a file with a PHP file extension (almost always `.php`).
- ▷ PHP treats everything as HTML until it encounters a PHP open tag:
  - ▷ `<?php` “open tag” starts it processing as PHP
  - ▷ `?>` “close tag” means stop parsing as PHP
  - ▷ Only use close tag if you have more HTML

# Setup Starting Point

## What elements do we need, to start?

Think about building a fire.  
Without these three elements  
there's no fire!

- Oxygen
- Heat
- Fuel



## Three Things Power a PHP Website

- ▷ PHP
- ▷ Database (MySQL)
- ▷ Web server (Apache/nginx)

Like the three elements needed to start a fire - you need all three or there's no fire!

- Oxygen
- Heat
- Fuel



## Why Do We Care About Elements?

- ▷ Many possible setups
- ▷ Many possible environments
- ▷ But you always need the three elements

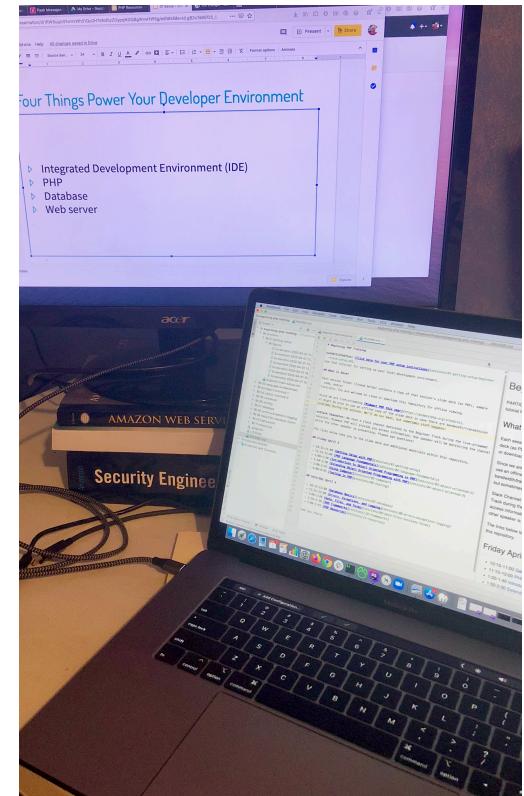
Your mission, should you choose to accept it, is to get **(PHP, Database, Server)** working in whatever environment you have available



# Four Things Power Your Developer Environment

- ▷ Integrated Development Environment (IDE)
- ▷ PHP
- ▷ Database
- ▷ Web server

No matter your setup, you need these four things



# Install

## Easy Start - Have You Done These Steps?

<https://github.com/sketchings/beginning-php-training>

- ▷ IDE - Visual Studio (free) or PhpStorm (30-day trial)
- ▷ MAMP (free) for integrated PHP, MySQL, Apache, nginx

Above options available Windows or Mac

# Visual Studio

Microsoft | Visual Studio Products Downloads Buy Support Subscriber Access [Free Visual Studio](#)

## Visual Studio

Best-in-class tools for any developer

### Visual Studio

Full-featured IDE to code, debug, test, and deploy to any platform

[Download Visual Studio](#)

[Learn more >](#)

### Visual Studio Code

Editing and debugging on any OS  
(By using Visual Studio Code you agree to its [license](#) and [privacy statement](#))

[Download Visual Studio Code](#)

[Learn more >](#)

### Visual Studio for Mac

Develop apps and games for iOS, Android, and web using .NET

[Read more about activating your license](#)

[Download Visual Studio for Mac](#)

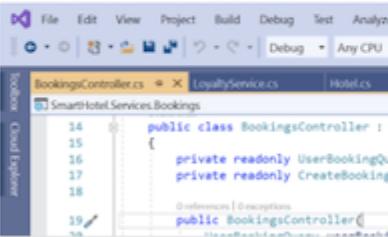
[Learn more >](#)

<https://visualstudio.microsoft.com/free-developer-offers/>

Microsoft | Visual Studio Products Downloads Buy Support Subscriber Access Free Visual Studio All Microsoft

# Everything you need to create great apps. Free.

### Visual Studio Community

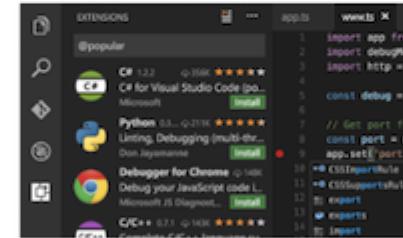


Full-featured IDE to code, debug, test, and deploy to any platform

[Free download](#)

[Learn more >](#)

### Visual Studio Code

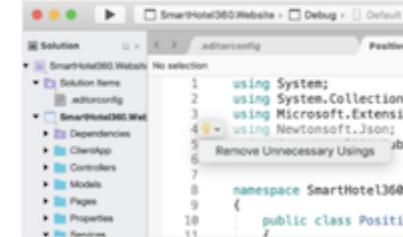


Editing and debugging on any OS  
 (By using Visual Studio Code you agree to its [license and privacy statement](#))

[Free download](#)

[Learn more >](#)

### Visual Studio Community for Mac



Develop apps and games for iOS, Android, and web using .NET  
[Read more about activating your license](#)

[Free download](#)

[Learn more >](#)

<https://laracasts.com/series/visual-studio-code-for-php-developers>

19

# Visual Studio Code for PHP Developers



Out of the blue, Microsoft jumps into the editor wars with an incredible offering that gives Sublime Text an overwhelming run for its money. In fact, it just might surpass it! So come along, as I demonstrate the ins, the outs, the tips, the techniques. Say hello to your new best friend: **Visual Studio Code**.

BEGIN    WATCH LATER    SUBSCRIBE

LATEST EPISODE IN THIS SERIES  
Added 2 years ago

Type Checking JavaScript Files

What if we could add a single commented line to the top of any vanilla JavaScript file...  
[WATCH](#)

Intermediate    18 episodes    1:20:20 hrs    Tooling

[F](#) [P](#) [T](#)

YOUR TEACHER | JEFFREY WAY



Hi, I'm Jeffrey. I'm the creator of Laracasts and spend most of my days building the site and thinking of new ways to teach confusing concepts. I live in Orlando, Florida with my wife and two kids.



VISIT WEBSITE

## You May Also Enjoy...

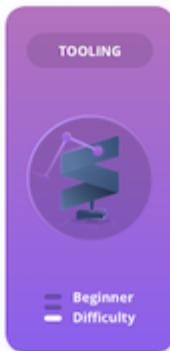
Here are some additional series that may be up your alley.



### Be Awesome in PHPStorm

Think about how many hours each week you spend within your editor. Doesn't it make sense to unlock every inch of its capabilities? I certainly subscri...

27 Lessons



### Sublime Text Mastery

If you intend to dedicate a career to web development, shouldn't you, in the process, harness every ounce of your code editor? In this series, I'll sh...

11 Lessons



### Vim Mastery

There's a reason why, after decades, countless developers still prefer Vim as their code editor of choice. That many people can't be wrong, right? In...

22 Lessons

<https://symfonycasts.com/screencast/phpstorm>

The screenshot shows a web browser displaying the SymfonyCasts website at <https://symfonycasts.com/screencast/phpstorm>. The page title is "Lean and Mean Dev with PhpStorm (for Symfony)". The main content area features a large image of the PhpStorm logo (a stylized "PS" inside a purple hexagon) on a grey background. To the left of the image is a teal button with the number "48:53". To the right, there is a call-to-action button that says "Login or register to track your progress!". Below this, a list of video lessons is shown:

Lesson Title	Duration
01. Setup	5:12
02. Annotations	2:47
03. Twig	5:01
04. Namespaces	4:05
05. Doctrine	5:00

**What you'll be learning**

Tip: This course is built on Symfony 2, but the important concepts of

<https://www.jetbrains.com/phpstorm/>

PhpStorm

Coming in 2020.1 What's New Features Learn Blog & Social Buy [Download](#)

## Getting Started



**Step Into Debugging with PhpStorm**

Gary Hockin  
Developer Advocate, JetBrains

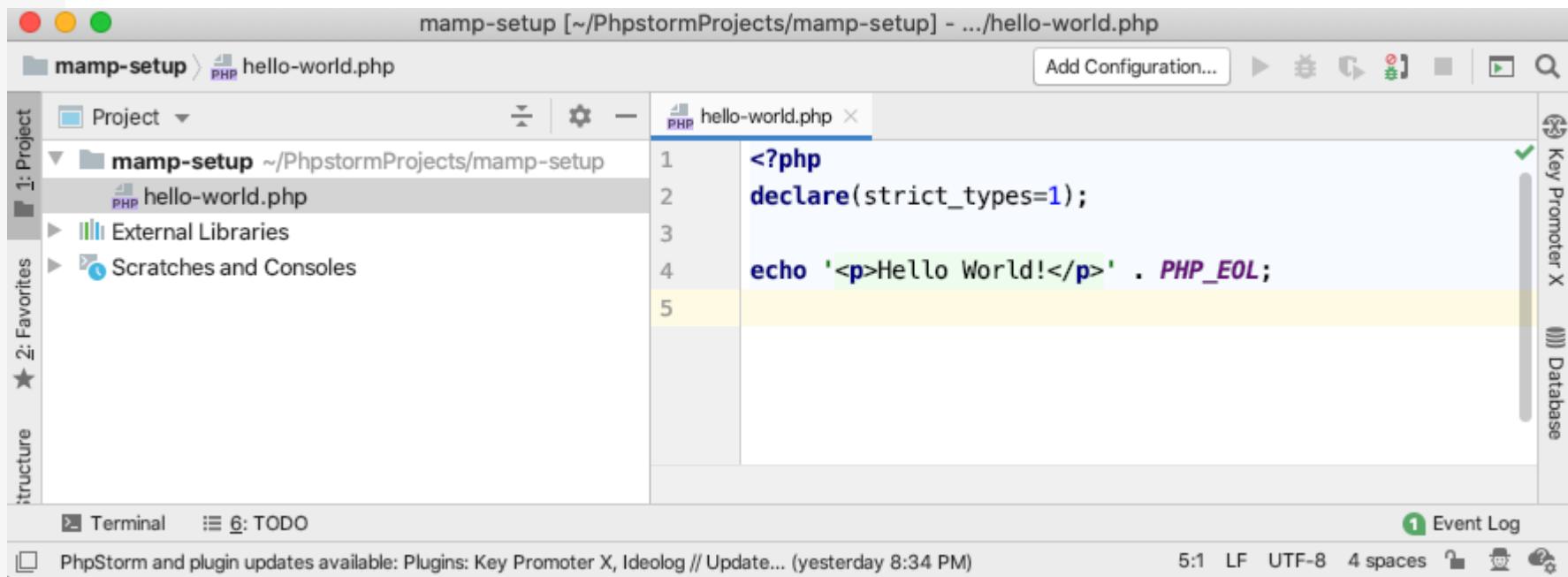
Category	Video Title	Duration
Debugging	Step Into Debugging with PhpStorm	58:32
Development Environment	Deployment and Remote Hosts in PhpStorm - PhpStorm Video Tutorial	5:26
Getting Started	Introduction to PhpStorm	6:18
	Basic PhpStorm Editor Actions and Navigation - PhpStorm	2:28

Resources

- [PhpStorm Tutorials](#)
- [PhpStorm Quick Start Guide](#)
- [PhpStorm Plugin Repository](#)
- [PhpStorm Online Help](#)
- [PhpStorm Leaflet \(1.9 Mb\)](#)
- [Keyboard Reference \(277 Kb\)](#)
- [Video Tutorials](#)
- [Webinars](#)
- [Ultimate Debugging Guide](#)

23

# Create a Project in Your Editor / IDE

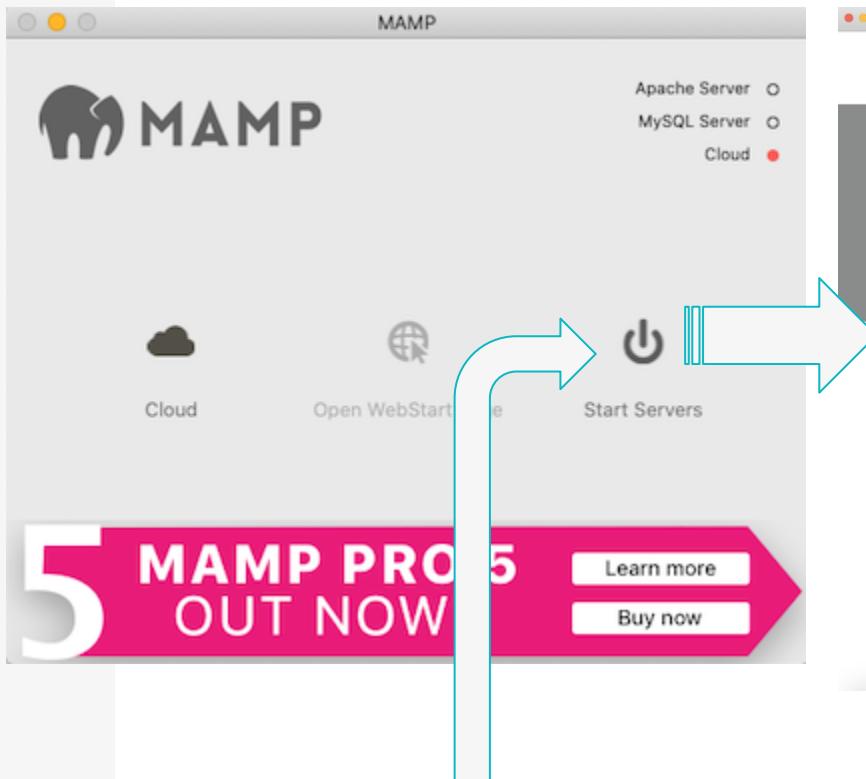


The screenshot shows the PhpStorm IDE interface. The title bar reads "mamp-setup [~/PhpstormProjects/mamp-setup] - .../hello-world.php". The left sidebar shows a project structure with a "Project" view expanded, displaying a folder "mamp-setup" containing a file "hello-world.php". Below this are sections for "External Libraries" and "Scratches and Consoles". The main editor area contains the following PHP code:

```
<?php
declare(strict_types=1);
echo '<p>Hello World!</p>' . PHP_EOL;
```

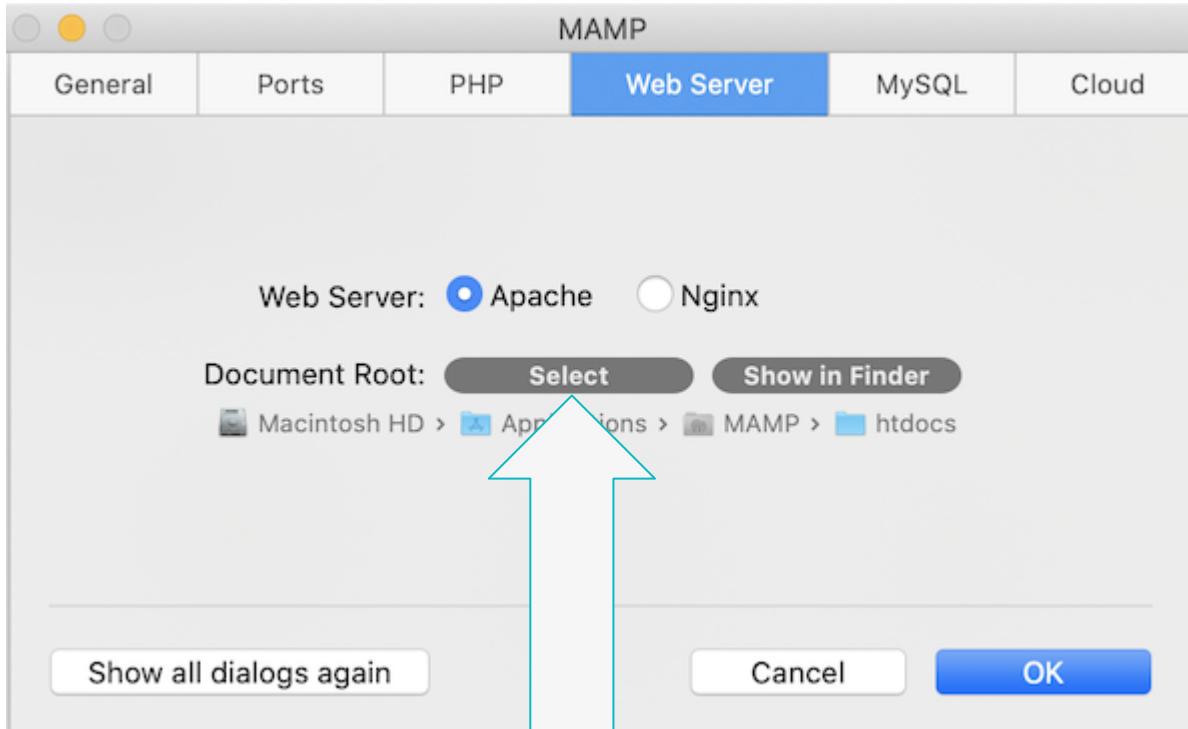
The code editor has syntax highlighting for PHP keywords and punctuation. The status bar at the bottom shows "Terminal", "6: TODO", "Event Log", "PhpStorm and plugin updates available: Plugins: Key Promoter X, Ideolog // Update... (yesterday 8:34 PM)", and file encoding information "5:1 LF UTF-8 4 spaces".

# Install MAMP - Start Servers - localhost:8888



The screenshot shows the "Welcome to MAMP" page. It indicates that the version is 5.7 and up to date. On the left, there are sections for "BUY MAMP PRO", "GET MAMP CLOUD", and "MAMP NEWS". The "BUY MAMP PRO" section features a "MAMP PRO" button. The "GET MAMP CLOUD" section features a "MAMP Cloud" button. The "MAMP NEWS" section includes a "Tweets by @mamp\_en" link and a "MAMP" button. On the right, there are sections for "PHP" and "MySQL". The "PHP" section says "phpinfo shows the current configuration of PHP.". The "MySQL" section says "MySQL can be administered with phpMyAdmin." A small note at the bottom right mentions a new font called "Mono".

## Preferences - Document Root



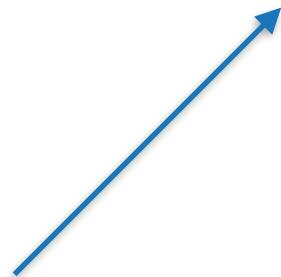
## Turn On Error Display

### PHP

[phpinfo](#) shows the current configuration of PHP.

### PHP-Caches

APC (not loaded)  
eAccelerator (not loaded)  
XCache (not loaded)  
OPcache (not loaded)



### phpMyAdmin

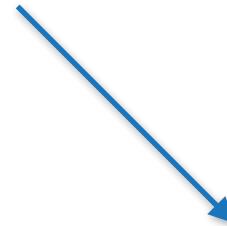
Configure your MySQL databases with [phpMyAdmin](#).

# Locate “ini” File

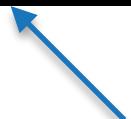


System	Darwin Edwards-MacBook-Pro.local 19.3.0 Darwin Kernel Version 19.3.0: Thu Jan 9 20:58:23 PST 2020; root:xn
Build Date	Feb 17 2020 12:49:52
Configure Command	'./configure' '--with-apxs2=/Applications/MAMP/Library/bin/apxs' '--with-zlib' '--with-zlib-dir=/Applications/MAMP/Library' '--prefix=/Applications/MAMP/bin/php/php7.4.2' '--sysconfdir=/Applications/MAMP/bin/php/php7.4.2/conf' '--with-bz2=/Applications/MAMP/Library' '--with-mysqli=mysqlnd' '--enable-mbstring=all' '--with-curl=/Applications/MAMP/Library' '--with-pgsql=shared,/Applications/MAMP/Library/pg' '--enable-exif' '--with-gettext=shared,/Applications/MAMP/Library' '--with-pgsql=shared,/Applications/MAMP/Library/pg' '--with-openssl=/Applications/MAMP/Library' '--with-iconv=/Applications/MAMP/Library' '--with-tidy=shared,/Applications/MAMP/Library' '--with-readline' '--with-mhash' '--with-iconv-dir=/Applications/MAMP/Library' '--with-argon2=/Applications/MAMP/Library' '--disable-cgi' '--with-zip' '--with-kerberos' '--with-pdo-sqlite' '--with-sqlite3=/Applications/MAMP/Library' '--disable-phpdbg' '--enable-gd' '--with-jpeg' '--with-freetype' '--with-pear' '--with-sqlite3-cflags=' 'SQLITE_CFLAGS=' 'SQLITE_LIBS=-lsqlite3' 'JPEG_CFLAGS=' 'JPEG_LIBS=-ljpeg' 'SASL_CFLAGS=-l/usr/include/sasl'
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/Applications/MAMP/bin/php/php7.4.2/conf
Loaded Configuration File	/Applications/MAMP/bin/php/php7.4.2/conf/php.ini

## Edit the File



```
Edwards-MacBook-Pro:design-pattern edwardbarnard$ pushd /Applications/MAMP/bin/php/php7.4.2/conf/  
/Applications/MAMP/bin/php/php7.4.2/conf ~/PhpstormProjects/DesignPattern/design-pattern  
Edwards-MacBook-Pro:conf edwardbarnard$ ls -al  
total 296  
drwxrwxr-x  5 edwardbarnard  admin    160 Apr  3 08:51 .  
drwxrwxr-x  7 edwardbarnard  admin    224 Feb 17 06:52 ..  
-rw-rw-r--  1 edwardbarnard  admin   1151 Feb 17 06:12 pear.conf  
-rw-rw-r--@ 1 edwardbarnard  admin  70515 Apr  3 08:51 php.ini  
-rw-rw-r--  1 edwardbarnard  admin  71147 Feb 17 06:12 php.ini.temp  
Edwards-MacBook-Pro:conf edwardbarnard$ vi php.ini  
Edwards-MacBook-Pro:conf edwardbarnard$ █
```



## Edit the File

```
; Common Values:  
; E_ALL (Show all errors, warnings and notices including coding standards.)  
; E_ALL & ~E_NOTICE (Show all errors, except for notices)  
; E_ALL & ~E_NOTICE & ~E_STRICT (Show all errors, except for notices and coding  
; E_COMPILE_ERROR|E_RECOVERABLE_ERROR|E_ERROR|E_CORE_ERROR (Show only errors)  
; Default Value: E_ALL & ~E_NOTICE & ~E_STRICT & ~E_DEPRECATED  
; Development Value: E_ALL  
; Production Value: E_ALL & ~E_DEPRECATED & ~E_STRICT  
; http://php.net/error-reporting  
error_reporting = E_ALL  
  
; This directive controls whether or not and where PHP will output errors,  
; notices and warnings too. Error output is very useful during development, but  
; it could be very dangerous in production environments. Depending on the code  
; which is triggering the error, sensitive information could potentially leak  
; out of your application such as database usernames and passwords or worse.  
; For production environments, we recommend logging errors rather than  
; sending them to STDOUT.  
; Possible Values:  
; Off = Do not display any errors  
; stderr = Display errors to STDERR (affects only CGI/CLI binaries!)  
; On or stdout = Display errors to STDOUT  
; Default Value: On  
; Development Value: On  
; Production Value: Off  
; http://php.net/display-errors  
display_errors = Off  
  
; The display of errors which occur during PHP's startup sequence are handled  
; separately from display_errors. PHP's default behavior is to suppress those  
; errors from clients. Turning the display of startup errors on can be useful in  
; debugging configuration problems. We strongly recommend you  
; set this to 'off' for production servers.  
; Default Value: Off  
; Development Value: On  
; Production Value: Off  
; http://php.net/display-startup-errors  
display_startup_errors = Off
```

## Edit the File

```
; http://php.net/error-reporting
error_reporting = E_ALL

; This directive controls whether or not and where PHP will output errors,
; notices and warnings too. Error output is very useful during development, but
; it could be very dangerous in production environments. Depending on the code
; which is triggering the error, sensitive information could potentially leak
; out of your application such as database usernames and passwords or worse.
; For production environments, we recommend logging errors rather than
; sending them to STDOUT.

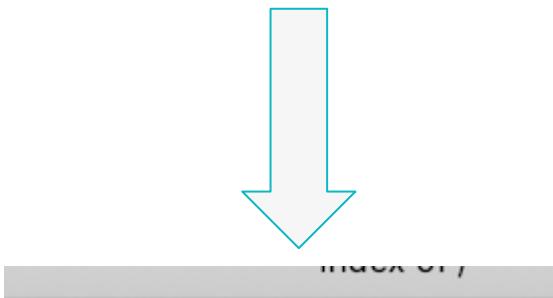
; Possible Values:
; Off = Do not display any errors
; stderr = Display errors to STDERR (affects only CGI/CLI binaries!)
; On or stdout = Display errors to STDOUT
; Default Value: On
; Development Value: On
; Production Value: Off
; http://php.net/display-errors
display_errors = On

; The display of errors which occur during PHP's startup sequence are handled
; separately from display_errors. PHP's default behavior is to suppress those
; errors from clients. Turning the display of startup errors on can be useful in
; debugging configuration problems. We strongly recommend you
; set this to 'off' for production servers.
; Default Value: Off
; Development Value: On
; Production Value: Off
; http://php.net/display-startup-errors
display_startup_errors = On

; Besides displaying errors, PHP can also log errors to locations such as a
; file or a database table. To do this, you must first enable error logging by
; setting error_log to a valid file or database connection identifier.
```

31

## “My Website”



A screenshot of a web browser showing the "Welcome to MAMP" page. The URL in the address bar is "localhost:8888/MAMP?language=English". The page features a large blue upward-pointing arrow on the left. At the top, there are navigation links: "START", "MY WEBSITE", "TOOLS", "MAMP WEBSITE", and a red "BUY MAMP PRO" button. The main content area says "Welcome to MAMP" and "Your version is 5.7 and up to date." Below this are three sections: "BUY MAMP PRO", "GET MAMP CLOUD", and "MAMP NEWS". Each section has a small icon, a title, and a brief description.

## Index of /

- [idea/](#)
- [hello-world.php](#)



A screenshot of a web browser showing the result of running "hello-world.php". The browser interface includes a top bar with red, yellow, and green dots, and navigation buttons (&lt;, &gt;, and a refresh icon). The main content area displays the URL "localhost:8888/hello-world.php" and the text "Hello World!".

Hello World!

## Easiest Exercise Ever

1. Create a file called `hello.php`
2. Inside it, type `Hello World!`
3. Save the file to your project directory for your local web server.
4. Navigate to the page on your local website.
5. Easiest. Hello. World. Ever.

## Hello World with PHP Code

- ▶ In hello.php, instead of Hello World! , write this and then reload the web page on your localhost:

```
<?php  
Echo 'Hello World!';
```

# MySQL

The screenshot shows the MAMP welcome page with a large downward arrow pointing to the MySQL section. The MySQL connection parameters listed are:

- Host**: localhost
- Port**: 8889
- User**: root
- Password**: root
- Socket**: /Applications/MAMP/tmp/mysql/mysql.sock

**Examples:**

PHP >= 5.6.x    PHP <= 5.5.x    Python    Perl

```
$user = 'root';
$password = 'root';
$db = 'inventory';
$host = 'localhost';
$port = 8889;

$link = mysqli_init();
$success = mysqli_real_connect(
    $link,
    $host,
    $user,
```

The screenshot shows the phpMyAdmin interface version 4.9.3. At the top, the URL bar displays "localhost:8888/phpMyAdmin/?lang=en". The main menu bar includes links for Databases, SQL, Status, User accounts, Export, Import, Settings, Replication, Variables, and More.

The left sidebar lists recent databases: information\_schema, mysql, performance\_schema, and sys. A large blue arrow points from the bottom of the sidebar down towards the "Console" button at the bottom center.

The central content area is divided into several sections:

- General settings:** Shows the server connection collation as "utf8mb4\_unicode\_ci".
- Appearance settings:** Includes language set to "English", theme set to "Original", and font size set to "82%".
- Database server:** Lists the following details:
  - Server: Localhost via UNIX socket
  - Server type: MySQL
  - Server connection: SSL is not being used
  - Server version: 5.7.26 - MySQL Community Server (GPL)
  - Protocol version: 10
  - User: root@localhost
  - Server charset: UTF-8 Unicode (utf8)
- Web server:** Lists the following details:
  - Apache/2.2.34 (Unix) mod\_wsgi/3.5 Python/2.7.13 PHP/7.4.2 mod\_ssl/2.2.34 OpenSSL/1.0.2o DAV/2 mod\_fastcgi/mod\_fastcgi-SNAP-0910052141 mod\_perl/2.0.11 Perl/v5.24.0
  - Database client version: libmysql - mysqlnd 7.4.2
  - PHP extension: mysqli curl mbstring
  - PHP version: 7.4.2
- phpMyAdmin:** Version information: 4.9.3

At the bottom center is a "Console" button.

# Success!

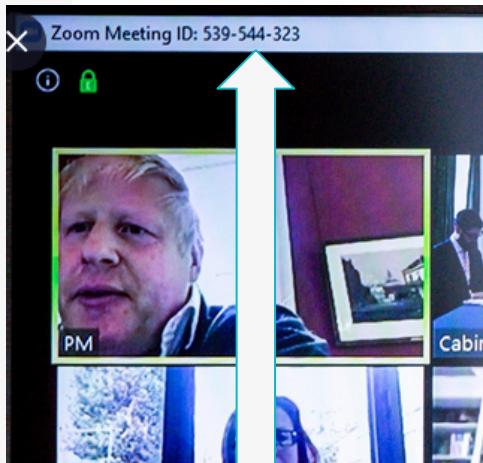
- ▷ Editor / IDE
- ▷ PHP
- ▷ Database
- ▷ Web server



# Development Environment

- ▷ Remember your laptop could be stolen
  - ▶ What constitutes a reportable data breach?
  - ▶ Can you keep sensitive passwords/keys OFF of your laptop?
  - ▶ Don't allow sensitive production data onto your laptop copy of the database (PII, HIPAA, GDPR)

# Don't Be Like the Prime Minister

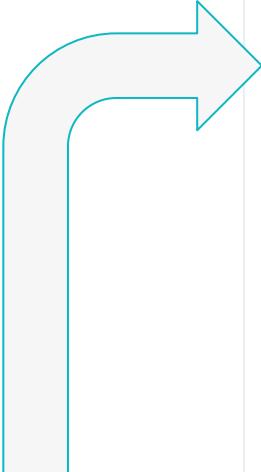


Zoom Meeting ID: 539-544-323

Boris Johnson #StayHomeSaveLives   
@BorisJohnson

This morning I chaired the first ever digital Cabinet.

Our message to the public is: stay at home, protect the NHS, save lives. #StayHomeSaveLives



## Remember - Web Images Contain Meta Data

- ▷ Meta data can reveal more than you intended
- ▷ Photos that include computer screens or sticky notes can prove embarrassing
- ▷ Well-intentioned screen shots... well... don't be like the Prime Minister
- ▷ Not a PHP problem? Still is YOUR problem!



## Ways to Build that “Fire” (dev environment)

- ▷ Integrated environments MAMP, WAMP, XAMPP:
  - ▶ Install and configure PHP, database, web server
  - ▶ Might not duplicate production env closely enough



## Containerized - Vagrant, Docker

- ▷ Rapid setup for new members of a team
- ▷ Specific environment(s) pre-built
- ▷ “Spin up” multiple experimental environments

## Virtual Machine

- ▷ Virtual Box (free) or VMWare
- ▷ Complete operating system and filesystem
- ▷ Can run
  - ▶ Windows on Mac
  - ▶ Linux on Windows
  - ▶ Et cetera

## Build Your Own

- ▷ Many ways to install PHP, including building from source
- ▷ Can be necessary for specific language extensions, specific versions, database drivers, etc.
- ▷ The XDebug extension is important - and is included in most packaged environments such as MAMP
- ▷ HomeBrew for the Mac

## Case-Sensitive Filesystem

- ▷ Linux servers tend to have case-sensitive file system
- ▷ Mac tends to NOT have case-sensitive file system
- ▷ Windows... reports vary

This disparity will become a problem when...

## PHP Does Not Care... Or Does It?

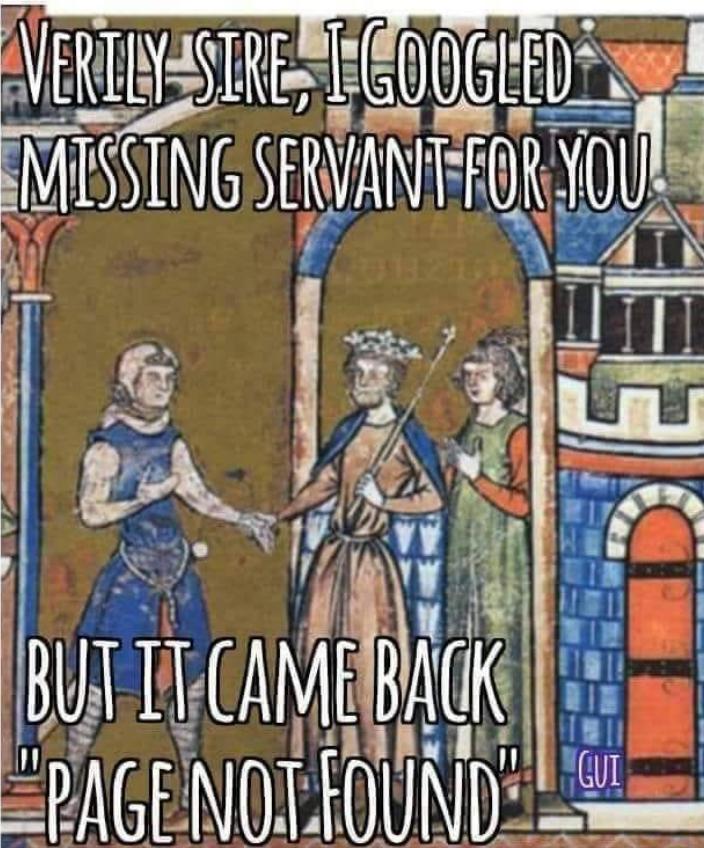
- ▷ PHP class names are NOT case sensitive
  - ▶ Class **FireBuilder** and **class firebuilder** are the same
- ▷ The linux filesystem DOES care
  - ▶ **FireBuilder.php** and **firebuilder.php** are NOT the same file
- ▷ The Mac filesystem DOES NOT care
  - ▶ **FireBuilder.php** and **firebuilder.php** ARE the same file

## What Happens?

1. Suppose real **class FireBuilder** declared in **FireBuilder.php**
2. Other code invokes **\$fire = new Firebuilder();**
3. Works fine in dev environment
4. Push to production... **class Firebuilder** not found
  - a. Mac autoloader finds **Firebuilder.php** when looking for **FireBuilder.php**
  - b. Linux autoloader cannot find **Firebuilder.php** because only **FireBuilder.php** exists
5. “Works for me”

## Blank Page

- ▷ “500” error in dev environment
- ▷ Often blank white page in production
- ▷ Find and check error logs
  - ▶ Coding error
  - ▶ Failed database connection
  - ▶ Unsuccessful deployment
- ▷ Recent code change most likely



## Remember Your Mission

- ▷ Get the four elements installed and playing-well with each other:
  - ▶ Code editor / IDE
  - ▶ PHP
  - ▶ Web server
  - ▶ Database
- ▷ Web pages are plain text (html and css) - you can edit them with your code editor / IDE

## What's Next

Build that fire!

- ▷ Code Editor / IDE
- ▷ PHP (7.2 or newer)
- ▷ Database (MySQL)
- ▷ Web server (Apache or nginx)

Easy route:

- ▷ Visual Studio / PhpStorm
- ▷ MAMP

Bonus: Install Composer  
<https://getcomposer.org/>

