Learn CodeIgniter in 1 Day

By Krishna Rungta

Table Of Content

1. What is MVC?
2. How MVC frameworks work?
3. CodeIgniter Controller
4. CodeIgniter Model

Chapter 5: CodeIgniter Controllers, Views Routing: Learn with Example App

1. How to create a new CodeIgniter project
2. CodeIgniter Routing
3. Create a Route
4. Create a Controller
5. Create a View

Chapter 6: CodeIgniter Routes: Learn with Example

1. What are Routes?
2. Routes Example
3. Creating URL's for the Application
4. Views

Chapter 7: CodeIgniter Form & Form Validation with Example

1. CodeIgniter Form Helper
2. Example Create Form
3. CodeIgniter Form Validation
4. Adding Form Validation Rules
5. Displaying Form Validation Error Messages
6. Populating Submitted Form Data: Sticky Forms
7. Example Form Validation

Chapter 8: Codeigniter Active Record: Insert, Select, Update, Delete

1. How to use Active Record: Example
2. CodeIgniter Database Configuration
3. CodeIgniter Insert Active Record
4. CodeIgniter Select Active Record
5. CodeIgniter Update Active Record
6. CodeIgniter Delete Active Record

Chapter 9: CodeIgniter Database Tutorial: Create, Update, Delete

1. CodeIgniter Working with Database
2. Database Configuration
3. CodeIgniter Database Models
4. Contacts Manager Views

Chapter 10: Pagination in Codeigniter with Step by Step Example

1. Database configuration
2. CodeIgniter Pagination Database Model
3. CodeIgniter Pagination Routes
4. CodeIgniter Pagination Controller

Chapter 11: How to Set Session in Codeigniter With Example

1. CodeIgniter Session Management
2. When to use sessions?
3. Sending Flash Messages to other pages with CI Sessions

4. Storing User Data in CI Sessions
5. CodeIgniter Session Views

Chapter 12: How to Upload Image & File in CodeIgniter (with Example)

1. CodeIgniter File Upload
2. Uploading Images in CodeIgniter
3. Testing the application

Chapter 13: How to Send Email using CodeIgniter

1. CodeIgniter Email Configuration
2. CodeIgniter Email View
3. CodeIgniter Email Controller
4. Email Routes

Chapter 14: Laravel vs CodeIgniter: Which is Better?

1. What is Laravel?
2. What is CodeIgniter?
3. Why use Laravel?
4. Why use CodeIgniter?
5. Features of Laravel
6. Features of CodeIgniter
7. Laravel vs. CodeIgniter: Know the Difference
8. Laravel vs. CodeIgniter which is better?

Chapter 1: What is CodeIgniter? How does it Work?

What is CodeIgniter?

CodeIgniter is a PHP MVC framework for developing applications rapidly. CodeIgniter provides out of the box libraries for connecting to the database and performing various operations. Like sending emails, uploading files, managing sessions, etc.

CodeIgniter Features

Let's see some of the features that make CodeIgniter great. The following list is not exhaustive but gives you an idea of what to expect when working with CodeIgniter.

Small footprint

The entire source code for CodeIgniter framework is close to 2MB. This makes it easy to master CodeIgniter and how it works. It also simplifies deploying and updating it.

Blazing fast

Users tend to favor applications that load very fast. If you have worked with some of the modern frameworks, then you will realize that they take less than one second to load just after installation. CodeIgniter,

you can loads on average around less than 50ms. The extra time spent optimizing like is the case in another framework is freed up when you are working with CodeIgniter.

Loosely coupled

The built-in features are designed to work independently without relying too much on other components. This makes it easy to maintain and make upgrades

MVC Architecture

The framework uses the Model-View-Controller architectural design. It is industry standard practices when working with web applications. MVC separates the data, business logic, and presentation.

Excellent documentation:

The framework is well documented, and there are good books, tutorials and answered forum questions on CodeIgniter. This means whatever challenge that you have, chances are someone has already encountered the problem, solved it and the solution is out there for you.

Application specific built-in components:

CodeIgniter has components for sending email, database management, session management and many more as you will discover as we continue with the tutorials.

Extendable:

CodeIgniter comes with some libraries, and helpers out of the box. If

what you want is not there or you would like to implement an existing feature your way. Then you can do so easily by creating your libraries, helpers, packages, etc.

Short learning curve:

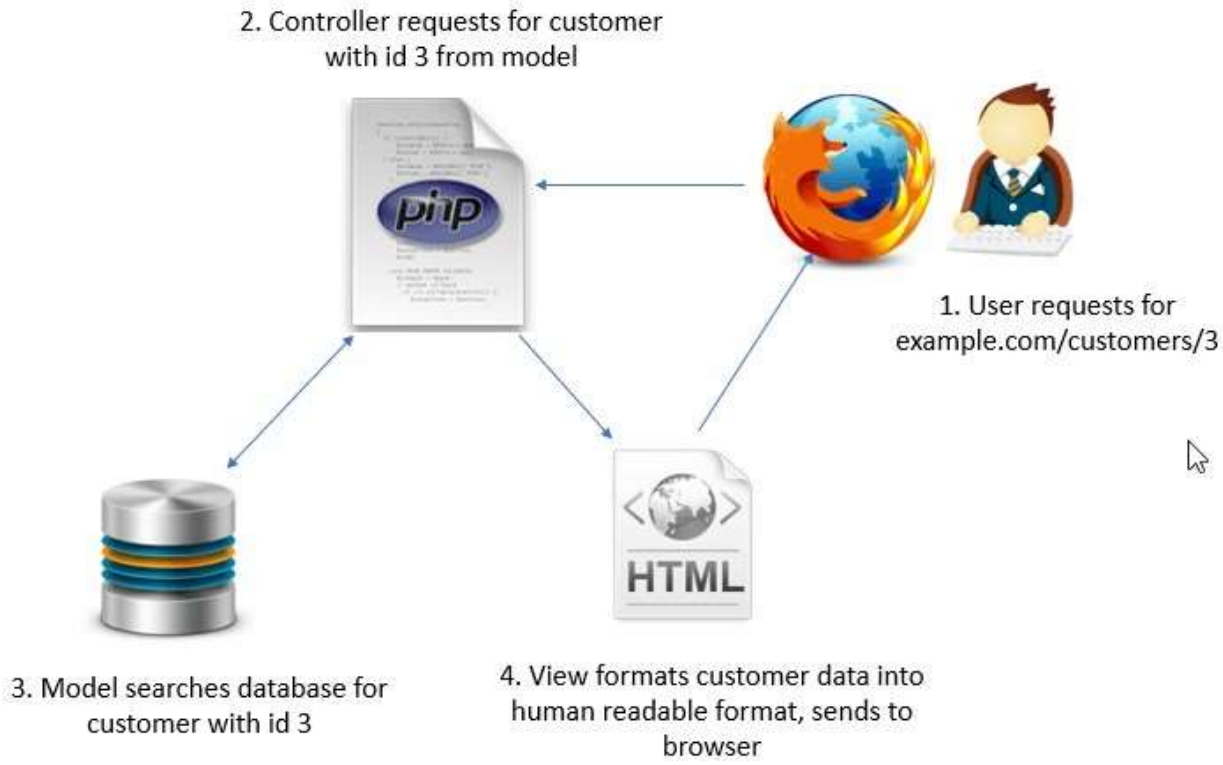
CodeIgniter is easy to master for anyone who is already familiar with PHP. Within a very short time, the student can start developing professional applications using CodeIgniter.

How CodeIgniter Works?

CodeIgniter is an MVC framework. MVC stands for Model View Controller. When a user requests a resource, the controller responds first. The controller understands the user request then request the necessary data if necessary.

For example, if you want to retrieve a customer with the id= 3, the controller will receive your request, then request the model to retrieve the record with the id of 3. The model returns the record to the controller. The controller then forwards the result to the view which formats it into a human-readable format. Then the results are returned to the user in the browser.

The following image shows how CodeIgniter works:



CodeIgniter Release History

2006	First version of CodeIgniter
2009	ExpressionEngine 2.0 launched
2014	British Columbia Institute of Technology took ownership of the project
2019	Stable version 4 expected to launch

Summary

- CodeIgniter is a PHP framework for developing applications rapidly
- The entire source code for CodeIgniter is close to 2MB. This makes it easy to master CodeIgniter and how it works
- The built-in features of CodeIgniter are designed to work independently without relying too much on other components

- The framework uses the Model-View-Controller architectural design
- The framework is well documented, and they are good books, tutorials and answered forum questions on CodeIgniter CodeIgniter comes with
- some libraries, and helpers users out of the box
- CodeIgniter is easy to master for anyone who is already familiar with PHP
- In CodeIgniter user requests a resource, the controller responds first. The controller understands the user request then request the necessary data if it is important

Chapter 2: How to Download & Install CodeIgniter + Composer [Configuration Included]

In this tutorial, we are going to look at how you can install and configure CodeIgniter. They are two ways of installation CodeIgniter. You can download the latest version from the CodeIgniter website, or you can use a tool like a composer to automate the installation

Download and Install Latest CodeIgniter Framework

The source code for the CodeIgniter framework is available on the official CodeIgniter website. If you want to download the latest version of the framework, then you should do it from the official web page.

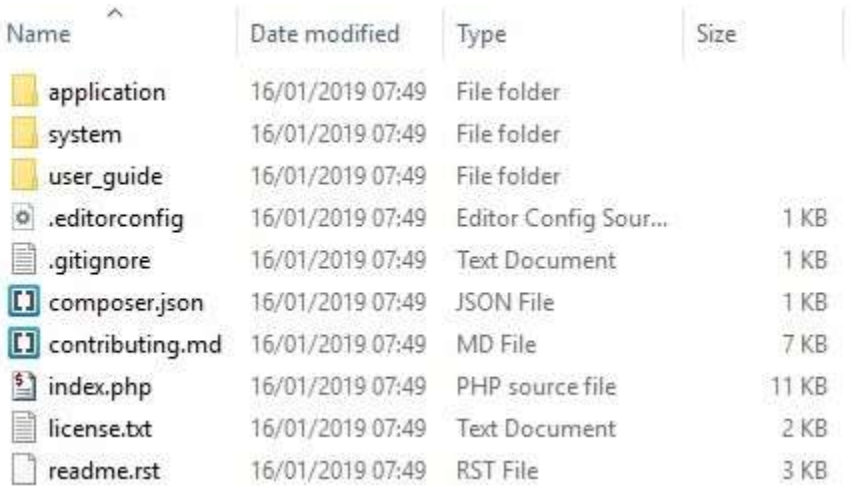
Step 1) Open the following URL in your browser https://codeigniter.com/

The image below shows the download link to the latest version of the framework

Step 2) Clicking the above link will download the framework as a zipped folder. Unzip the contents of CodeIgniter-3.1.10.zip

Step 3) Let's say you want to create a project called the online store. You can follow the following steps to start your project. Create a new directory in on your development drive, e.g, D:\Sites\online-store

Step 4) Open the contents of CodeIgniter-3.1.10, you should be able to see the following files



Name	Date modified	Type	Size
application	16/01/2019 07:49	File folder	
system	16/01/2019 07:49	File folder	
user_guide	16/01/2019 07:49	File folder	
.editorconfig	16/01/2019 07:49	Editor Config Sour...	1 KB
.gitignore	16/01/2019 07:49	Text Document	1 KB
composer.json	16/01/2019 07:49	JSON File	1 KB
contributing.md	16/01/2019 07:49	MD File	7 KB
index.php	16/01/2019 07:49	PHP source file	11 KB
license.txt	16/01/2019 07:49	Text Document	2 KB
readme.rst	16/01/2019 07:49	RST File	3 KB

Copy the above contents to your project directory, e.g., D:\Sites\online-store

Step 5) Just to make sure everything is ok, open the terminal and start the built-in PHP server

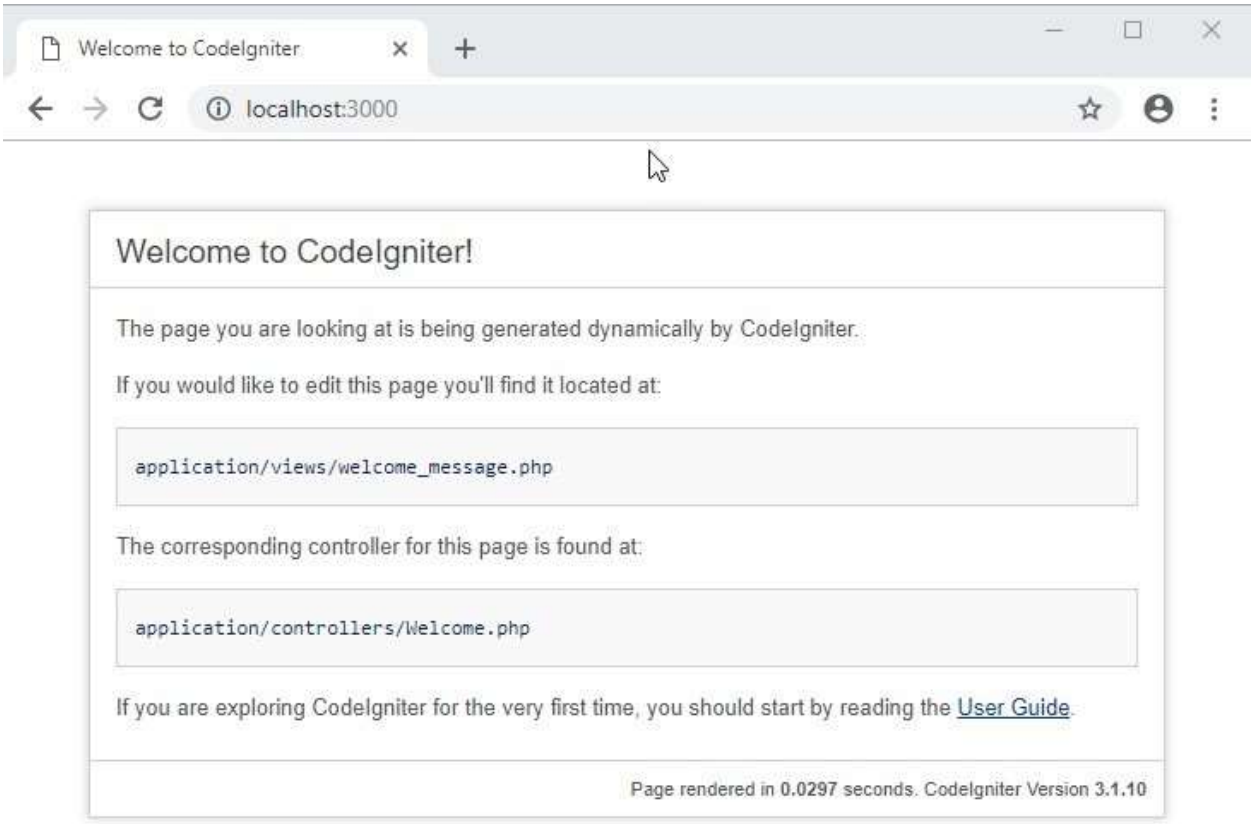
```
cd D:\Sites\ online-store
```

Run the following command

```
php -S localhost:3000
```

load the following URL into your browser

http://localhost:3000/



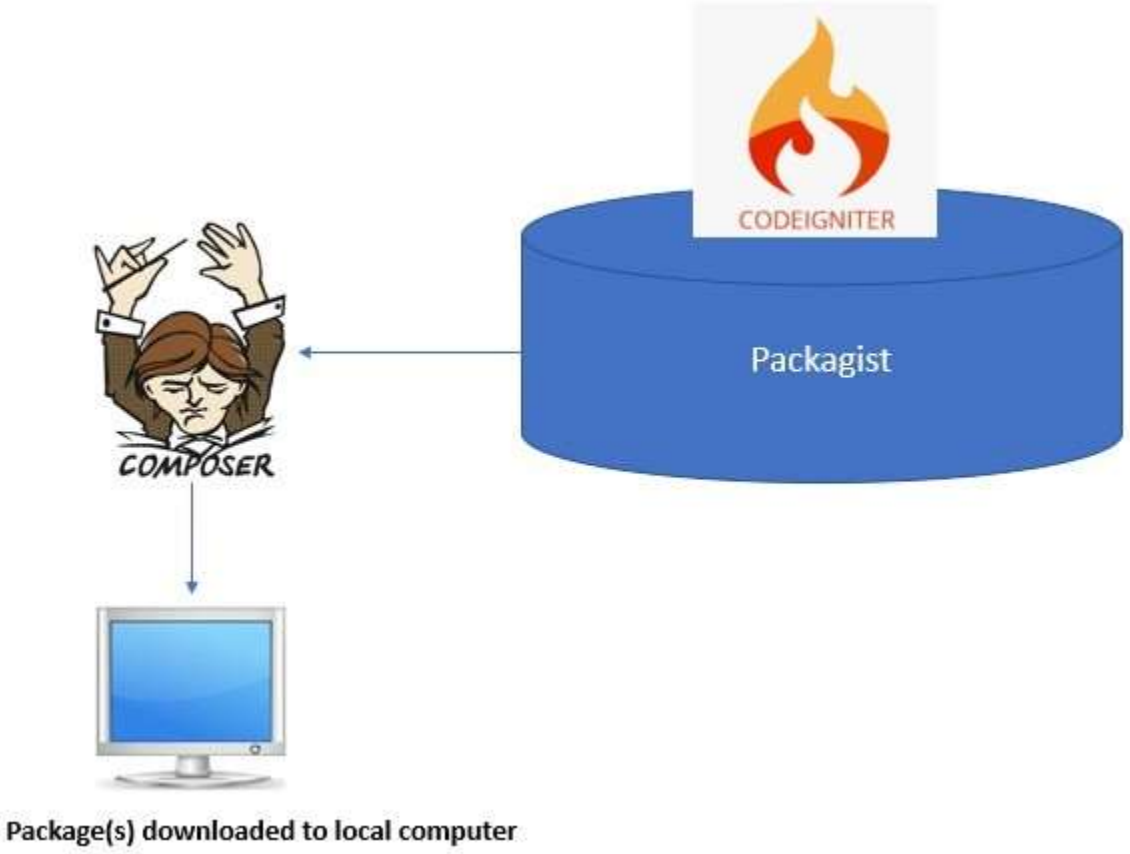
If you see above image, all is working well,

What is Composer?

The composer is a package management system for PHP. A package is simply a collection of PHP scripts that work together towards a single

goal. Based on this definition, CodeIgniter can even though it's a framework, qualifies to be labeled a package in composer terminologies.

The following image shows how the composer works



The author of CodeIgniter hosts the package at Packagist which is a central repository for PHP packages, etc.

When the developer runs the composer command to download CodeIgniter, Composer communicates with Packagist and downloads the latest release of the package. In addition to installing frameworks such as CodeIgniter, Composer can also be used to;

- Install individual packages such as third-party email or database Library

- Update existing packages
- Remove installed packages

How to install Composer

Step 1) Load the following URL in your browser
https://getcomposer.org/download/

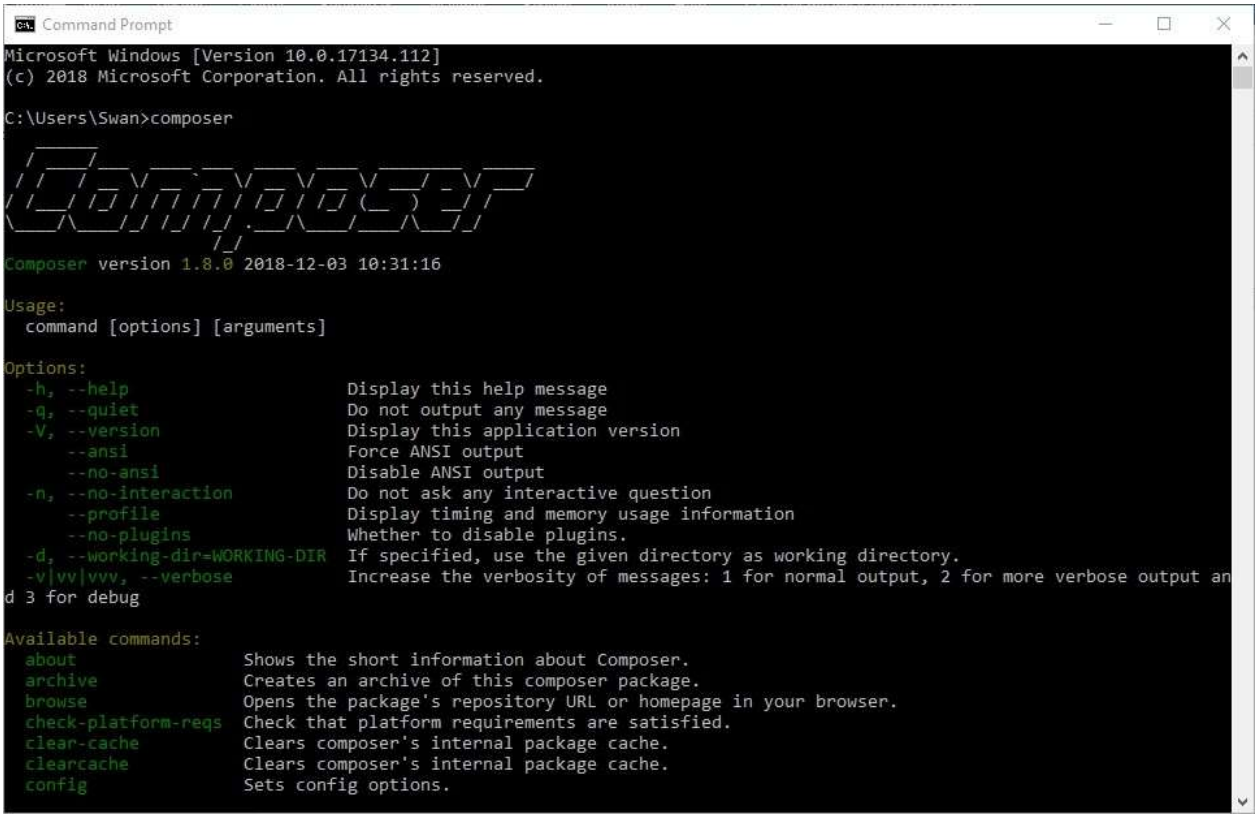
Download the setup and follow the installation instructions.

Step 2) Open the command prompt/terminal Run the

following command

```
composer
```

You will see the following results



If you can see the above results, then congratulations, you have successfully installed the composer.

Let's now create a new project called online-store Run

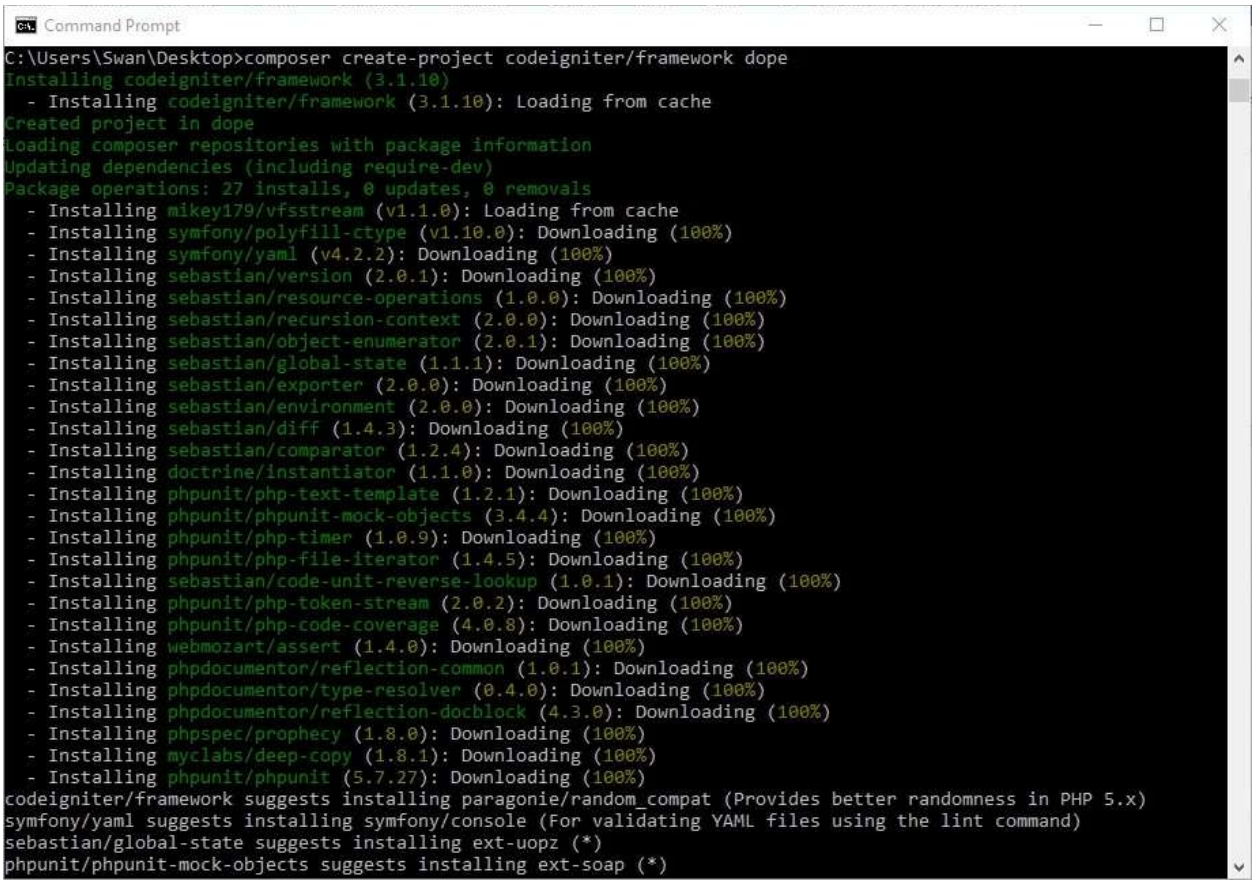
the following command

```
composer create-project CodeIgniter/framework online-store
```

HERE,

- composer create-project CodeIgniter/framework online-store composer invokes the composer program, create-project downloads the specified project framework which is in the namespace CodeIgniter.

You should be able to see results that is similar to the following



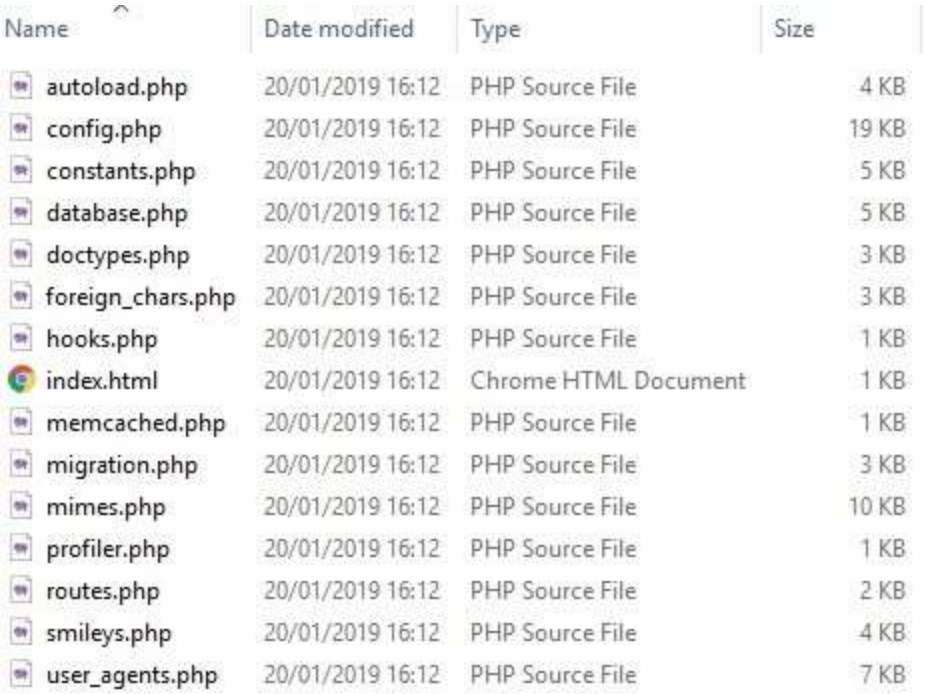
If you are a big fan of commands on the terminal then this is the way to go otherwise you can use the good old fashioned download the zipped file, unzip and happy coding.

CodeIgniter Config Files

Now that we have successfully installed CodeIgniter let's look at the configuration directory

The configuration directory is located in

```
application/config
```



Name	Date modified	Type	Size
autoload.php	20/01/2019 16:12	PHP Source File	4 KB
config.php	20/01/2019 16:12	PHP Source File	19 KB
constants.php	20/01/2019 16:12	PHP Source File	5 KB
database.php	20/01/2019 16:12	PHP Source File	5 KB
doctypes.php	20/01/2019 16:12	PHP Source File	3 KB
foreign_chars.php	20/01/2019 16:12	PHP Source File	3 KB
hooks.php	20/01/2019 16:12	PHP Source File	1 KB
index.html	20/01/2019 16:12	Chrome HTML Document	1 KB
memcached.php	20/01/2019 16:12	PHP Source File	1 KB
migration.php	20/01/2019 16:12	PHP Source File	3 KB
mimes.php	20/01/2019 16:12	PHP Source File	10 KB
profiler.php	20/01/2019 16:12	PHP Source File	1 KB
routes.php	20/01/2019 16:12	PHP Source File	2 KB
smileys.php	20/01/2019 16:12	PHP Source File	4 KB
user_agents.php	20/01/2019 16:12	PHP Source File	7 KB

HERE,

- autoload.php – specifies the helpers, libraries, drivers, packages, etc that should be loaded when the application starts
- config.php – contains application configurations such as base url, language, query strings, etc.
- constants.php – as the name suggets, this file I used to define application constants
- database.php – contains database connection parameters
- doctypes.php – defines document types i.e. html4, html5, sv10 etc
- foreign_chars.php – defines foreign characters that are to say characters that are found in languages such as Russian and others
- hooks.php – allows you to define your own hooks memcached.php – if
- you are using CodeIgniter together with Memcached then you can use this file for configurations. migration.php – if you want to use database
- migrations in CodeIgniter then you can use this file to config the
- settings. mimes.php – contains file mime types

- profile.php – contains settings that ae used by the built-in CodeIgniter compiler
- routes.php – contains the application routes
- smileys.php – contains settings for smileys
- user_agents.php – contains settings for browser user agents, i.e., Chrome, Opera, Firefox, etc.

CodeIgniter Configurations

let's now make some of the most common settings in CodeIgniter

```
Open application/config/config.php
```

Base URL

```
$config['base_url'] = '';
```

Sets the base URL. If its blank then CodeIgniter will set it for you automatically. If you want to be explicit about your base URL, then you can use the something like the following

```
$config['base_url'] = 'http://localhost:3000';
```

HERE,

- $config['base_url'] = 'http://localhost:3000'; sets the base URL to localhost running on port 3000.

Class Prefix

CodeIgniter uses the prefix CI_Classname. As a best practice and to avoid collisions with internal classes, you can prefix your class, i.e., MY_Classname. The following line is used to set your class prefix

```
$config['subclass_prefix'] = 'MY_';
```

Query Strings

These are parameters that are visited in the URL, i.e., example.com/index.php?q=eggs. If you would like to use such URLs, then you will have to set

```
$config['enable_query_strings'] = FALSE;
To
$config['enable_query_strings'] = TRUE;
```

Other settings

They are many settings that you can set in config.php including date formats, cache and view paths, etc. much of what you configure depends on your application needs

How to remove index.php in CodeIgniter

CodeIgniter is an MVC framework. This means it has a single entry point into the application which is index.php. It doesn't matter what URL you access. The all go through index.php. by default, index.php is shown in the URL as shown in the example below

```
example.com/index.php?q=eggs
```

The URL looks longer and weird. The good thing is you can configure CodeIgniter to remove that.

Open application/config/config.php

Locate the following line

```
$config['index_page'] = 'index.php';
Set it to the following
$config['index_page'] = '';
```

HERE,

- We are using mod_rewrite to remove the page so as per requirement, this should be set to blank.

Next, we need to create the .htaccess that rewrites the URLs Add a

new file .htacces in the root directory of the application Add the

following code

```
RewriteEngine on
RewriteCond $1 !^(index\.php|resources|robots\.txt)
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^(.*)$ index.php/$1 [L,QSA]
```

HERE,

- The above code is for configuring web servers that run apache server. The above code basically gets the URI parameters and executes them via index.php even if it's not showing in the browser URL.

Summary

- They are two ways of installation CodeIgniter. You can download the latest version from the CodeIgniter website, or you can use composer to automate the installation
- The composer is a package management system for PHP
- A composer can be used for: Install individual packages, Update existing packages remove installed packages