

Fall, 2020

COMP2121 E-Business

Advanced Topics on PHP

Kevin Kam Fung Yuen
Department of Computing
Hong Kong Polytechnic University
kevin.yuen@polyu.edu.hk

Outline

- Set path for php.exe
- Install X-debugger
- Use of X-debugger
- Object-Oriented Programming (OOP) in PHP
- PHP-OOP and MySQL: Product Display
- Composer
- Laravel

Set path for php.exe

Set the Temporary Path where your

Set Path = D:\eBusinessKevinYuenLabs\php7410

Or

Set Permanent Path in “environment variables” in System properties.

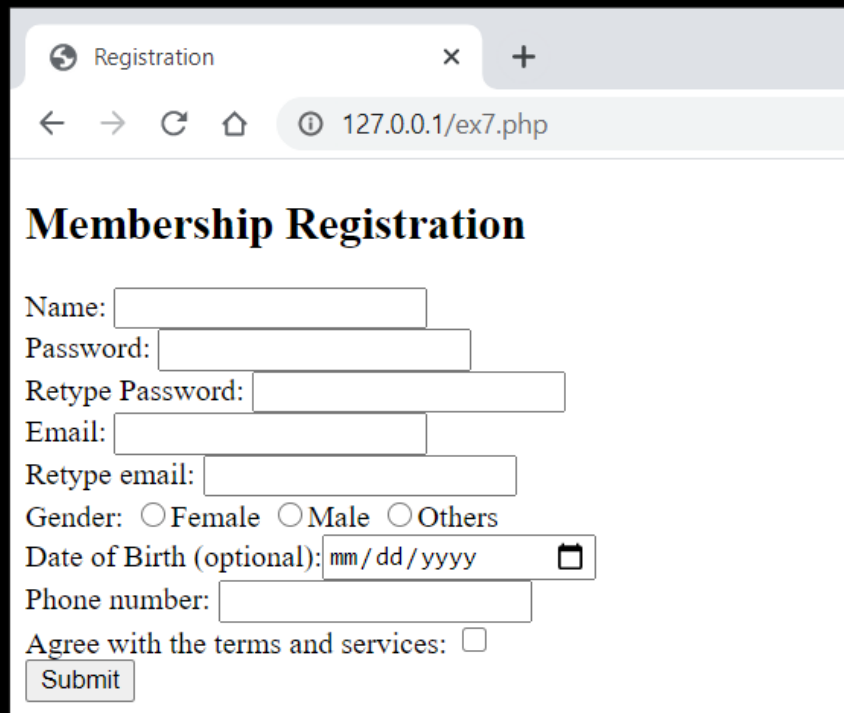
Test php.exe

1. C:\>php -S 127.0.0.1:80 D:\eBusinessKevinYuenLabs\Apache24\htdocs\LAB07PHP\ex7.php


C:\> Select Command Prompt - php -S 127.0.0.1:80 D:\eBusinessKevinYuenLabs\Apache24\htdocs\LAB07PHP\ex7.php

```
C:\>php -S 127.0.0.1:80 D:\eBusinessKevinYuenLabs\Apache24\htdocs\LAB07PHP\ex7.php
[Thu Nov 5 23:57:27 2020] PHP 7.4.10 Development Server (http://127.0.0.1:80) started
[Thu Nov 5 23:57:38 2020] 127.0.0.1:57529 Accepted
[Thu Nov 5 23:57:39 2020] 127.0.0.1:57529 Closing
[Thu Nov 5 23:57:39 2020] 127.0.0.1:57534 Accepted
[Thu Nov 5 23:57:39 2020] 127.0.0.1:57534 Closing
```

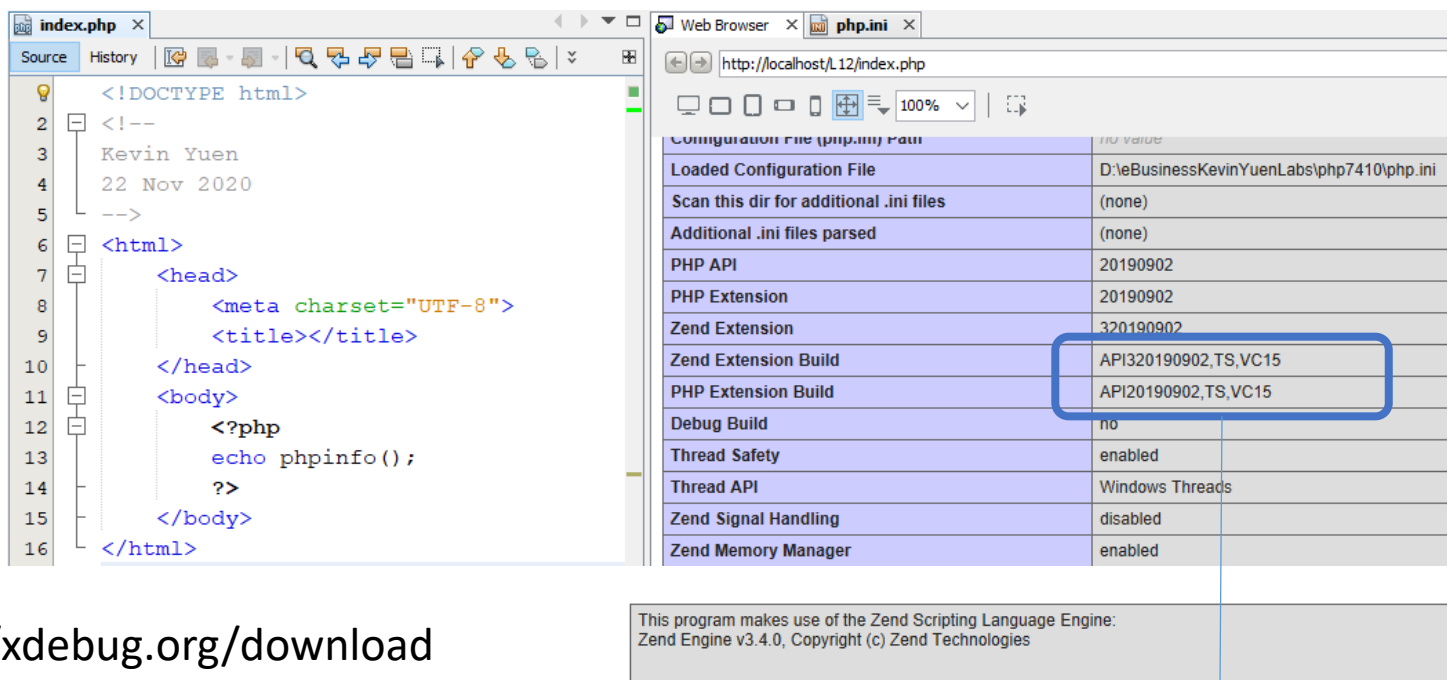
3. "Ctrl + C" to stop



The screenshot shows a web browser window with a single tab titled "Registration". The address bar displays "127.0.0.1/ex7.php". The page content is a "Membership Registration" form with the following fields and controls:

- Name:
- Password:
- Retype Password:
- Email:
- Retype email:
- Gender: ☐ Female ☐ Male ☐ Others
- Date of Birth (optional): 
- Phone number:
- Agree with the terms and services: ☐
- Submit:

2. <http://127.0.0.1/ex7.php>



Install X-debug

X-debug is not installed by default

<https://xdebug.org/download>

- Windows binaries:

Download the right version

PHP 7.2 VC15 (64 bit)	PHP 7.2 VC15 (32 bit)	PHP 7.2 VC15 TS (64 bit)	PHP 7.2 VC15 TS (32 bit)
PHP 7.3 VC15 (64 bit)	PHP 7.3 VC15 (32 bit)	PHP 7.3 VC15 TS (64 bit)	PHP 7.3 VC15 TS (32 bit)
PHP 7.4 VC15 (64 bit)	PHP 7.4 VC15 (32 bit)	PHP 7.4 VC15 TS (64 bit)	PHP 7.4 VC15 TS (32 bit)
PHP 8.0 VS16 (64 bit)	PHP 8.0 VS16 (32 bit)	PHP 8.0 VS16 TS (64 bit)	PHP 8.0 VS16 TS (32 bit)

php.ini : add the code below

```
955 ;full path to your file if you have not set dextension_dir
956 zend_extension= php_xdebug-3.0.0RC1-7.4-vc15-x86_64.dll
957 xdebug.mode=debug
958 xdebug.client_host = "localhost"
959 xdebug.client_port=9003
960
961 ;xdebug.remote_enable=1 ;old one
962 ;xdebug.remote_host="localhost"
963 ;xdebug.remote_port=9000
```

You will find a lot of such codes in the internet,
which is for old version of PHP

Restart apache
Re-run the php with echo phpinfo();

This program makes use of the Zend Scripting Language Engine:
Zend Engine v3.4.0 Copyright (c) Zend Technologies
with Xdebug v3.0.0RC1, Copyright (c) 2002-2020, by Derick Rethans

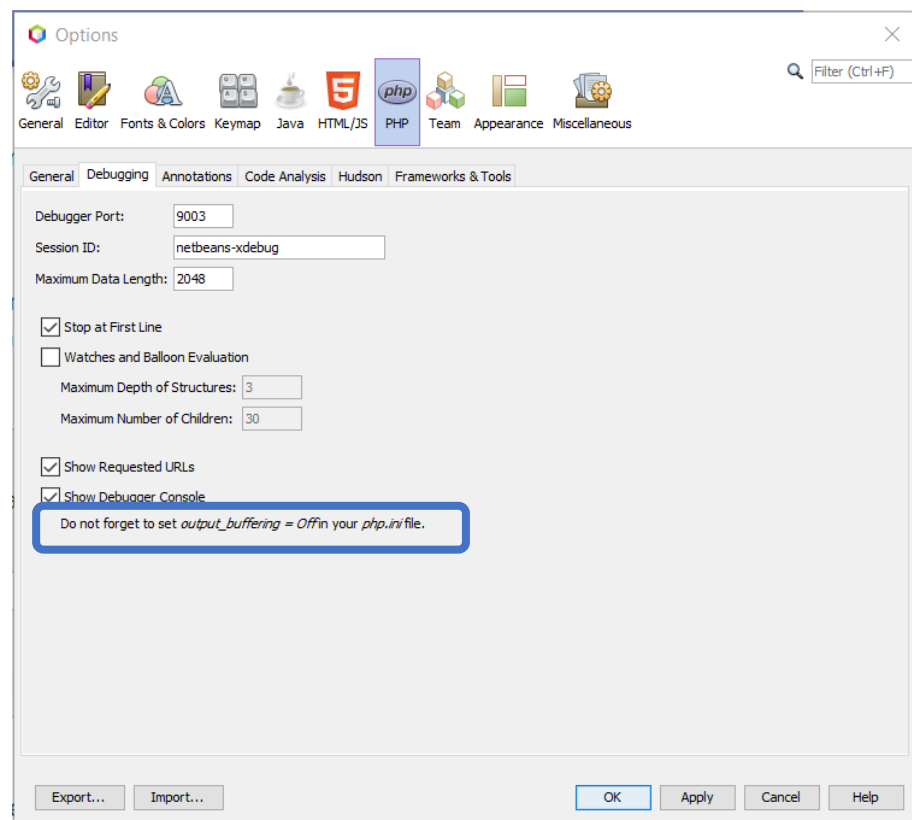


```
C:\Users\Kevin Yuen>php -v
PHP 7.4.10 (cli) (built: Sep  1 2020 16:52:39) ( ZTS Visual C++ 2017 x64 )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies
with Xdebug v3.0.0RC1, Copyright (c) 2002-2020, by Derick Rethans
```

Congratulations!

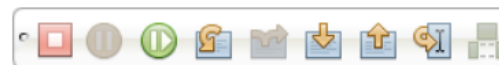
Config Xdebug in Netbeans

Tools > Options



Using the Debugger Toolbar

When you run a debugging session, the debugger toolbar displays above the editor.



The toolbar provides you with the following actions:

- Finish Session** () Finish debugging session
- Pause** () Suspend debugging session
- Resume** () Resume debugging session
- Step Over** () Step over an execution statement
- Step Into** () Step into a function call
- Step Out** () Step out of the current function call
- Run to Cursor** () Run execution to the position of the cursor

We will see more demos using Xdebug in following sessions.

```
215 ;output_buffering = 4096
216 output_buffering = off
```

<https://netbeans.org/kb/docs/php/debugging.html>

Object-Oriented Programming (OOP) in PHP

Procedural programming is about writing procedures or functions that perform operations on the data, while object-oriented programming is about creating objects that contain both data and functions.

Object-oriented programming has several advantages over procedural programming:

- OOP is faster and easier to execute;
- OOP provides a clear structure for the programs;
- OOP helps to keep the PHP code DRY "Don't Repeat Yourself", and makes the code easier to maintain, modify and debug;
- OOP makes it possible to create full reusable applications with less code and shorter development time;

Tip: The "Don't Repeat Yourself" (DRY) principle is about reducing the repetition of code. You should extract out the codes that are common for the application, and place them at a single place and reuse them instead of repeating it.

https://www.w3schools.com/php/php_oop_what_is.asp

Class and Objects/instances

Name
Attributes
Methods

Product
ID name brand Price Description
getProductInfo()

Class and Objects

Phone
ID
Name
Brand
Colour
Capacity
Price
Photo
getProductInfo()

instantiation

phone1: Phone

ID: 1
Name: "iPhone 12 mimi"
Brand : "Apple"
Colour: White
Capacity: 64GB
Price: "HK\$5999"
Photo: file1.jpg
getProductInfo()

phone2: Phone

ID: 2
Name: "Galaxy A42 5G Smartphone"
Brand : "SAMSUNG"
Colour: White
Capacity: 8GB +128GB
Price: "HK\$3198"
Photo: file2.jpg
getProductInfo()

<https://www.fortress.com.hk>

classLib.php

```

7  class Phone {
8      //Properties
9      private $ID;
10     private $Name;
11     private $Brand;
12     private $Colour;
13     private $Capacity;
14     private $Price;
15     private $Photo;
16
17     //Constructor function starts with two underscores (__) before "construct".
18     function __construct($ID,$Name,$Brand,$Colour,$Capacity,$Price,$Photo) {
19         $this->ID = $ID;
20         $this->Name = $Name;
21         $this->Brand = $Brand;
22         $this->Colour = $Colour;
23         $this->Capacity = $Capacity;
24         $this->Price = $Price;
25         $this->Photo = $Photo;
26     }
27
28     function getProductInfo()
29     {
30         // PHP Associative Array
31         return array("ID"=>$this->ID,"Name"=>$this->Name,"Brand"=>$this->Brand,
32                     "Colour"=>$this->Colour,"Capacity"=>$this->Capacity,
33                     "Price"=>$this->Price,"Photo"=>$this->Photo);
34     }
35 }

```

Phone

ID
Name
Brand
Colour
Capacity
Price
Photo

getProductInfo()

phone1: Phone

ID: 1
Name: "iPhone 12 mimi"
Brand : "Apple"
Colour: White
Capacity: 64GB
Price: "HK\$5999"
Photo: file1.jpg

getProductInfo()

phone2: Phone

ID: 2
Name: "Galaxy A42 5G Smartphone"
Brand : "SAMSUNG"
Colour: White
Capacity: 8GB +128GB
Price: "HK\$3198"
Photo: file2.jpg

getProductInfo()

```

13 <?php
14 // import library
15 include 'classLib.php';
16
17 // create Phone objects
18 $phone1 = new Phone("1", "iPhone 12 mimi", "Apple",
19                    "White", "64GB", "HK$5999", "img\p1.webp");
20
21 $phone2 = new Phone("2", "Galaxy A42 5G Smartphone", "SAMSUNG",
22                    "White", "8GB +128GB", "HK$3198", "img\p2.webp");
23
24 print_r($phone1->getProductInfo());
25 echo "<br>-----<br>";
26 print_r($phone2->getProductInfo());
27

```

phoneObjects.php

L12 x

http://localhost/L12/phoneObjects.php

100%

```

Array ( [ID] => 1 [Name] => iPhone 12 mimi [Brand] => Apple [Colour] =>
White [Capacity] => 64GB [Price] => HK$5999 [Photo] => img\p1.webp )
-----
Array ( [ID] => 2 [Name] => Galaxy A42 5G Smartphone [Brand] =>
SAMSUNG [Colour] => White [Capacity] => 8GB +128GB [Price] => HK$3198
[Photo] => img\p2.webp )

```

```
28 // using Array
29 echo "<br><br>Using Array<br>";
30 $productList = array($phone1,$phone2);
31
32 print_r($productList[0]->getProductInfo());
33 echo "<br>-----<br>";
34 print_r($productList[1]->getProductInfo());
35
```

Using Array
Array ([ID] => 1 [Name] => iPhone 12 mimi [Brand] => Apple [Colour] => White [Capacity] => 64GB [Price] => HK\$5999 [Photo] => img\p1.webp)

Array ([ID] => 2 [Name] => Galaxy A42 5G Smartphone [Brand] => SAMSUNG [Colour] => White [Capacity] => 8GB +128GB [Price] => HK\$3198 [Photo] => img\p2.webp)


```
36 //Loop through an Associative Array
37 echo "<br><br>Loop Through an Associative Array<br>";
38 foreach($productList as $product)
39 {
40     $prodInf = $product->getProductInfo();
41     foreach($prodInf as $k => $v) {
42         echo "Key=" . $k . ", Value=" . $v;
43         echo "<br>";
44     }
45     echo "<br>";
46 }
47
```

Loop Through an Associative Array

Key=ID, Value=1

Key=Name, Value=iPhone 12 mimi

Key=Brand, Value=Apple

Key=Colour, Value=White

Key=Capacity, Value=64GB

Key=Price, Value=HK\$5999

Key=Photo, Value=img\p1.webp

Key=ID, Value=2

Key=Name, Value=Galaxy A42 5G Smartphone

Key=Brand, Value=SAMSUNG

Key=Colour, Value=White

Key=Capacity, Value=8GB +128GB

Key=Price, Value=HK\$3198


Key=Photo, Value=img\p2.webp

```

48 //Print product list
49 $html = "<table border = '1'>";
50 $html .= "<thead>";
51 $html .= "<tr>";
52 $html .= "<th>Description</th>";
53 $html .= "<th>Photo </th>";
54 $html .= "</tr> </thead>";
55 $html .= "<tbody>";
56 foreach($productList as $product)
57 {
58     $phoneInfo = $product->getProductInfo();
59     $html .= "<tr><td><ul>";
60     $html .= "<li>Name: " . $phoneInfo['Name'] . "</li>";
61     $html .= "<li>Brand: " . $phoneInfo['Brand'] . "</li>";
62     $html .= "<li>Colour: " . $phoneInfo['Colour'] . "</li>";
63     $html .= "<li>Capacity: " . $phoneInfo['Capacity'] . "</li>";
64     $html .= "<li>Price: " . $phoneInfo['Price'] . "</li>";
65     $html .= "</ul></td>";
66     $html .= "<td><img src='". $phoneInfo['Photo']";
67     $html .= "' width='300' height='200' /> </td></tr>";
68 }
69 $html .= "</tbody></table>";
70 echo $html;

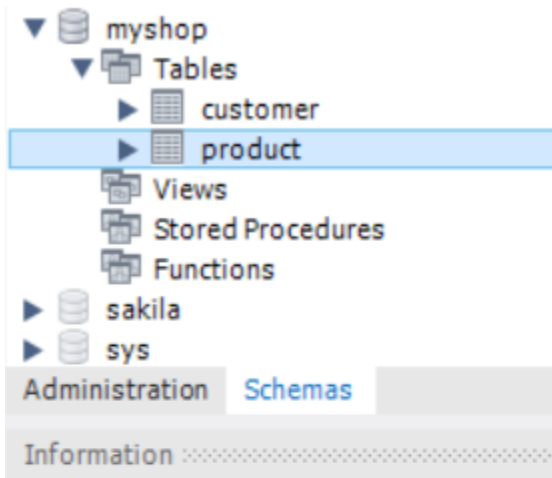
```

Chrome browser supports webp files

Description	Photo
<ul style="list-style-type: none"> Name: iPhone 12 mimi Brand: Apple Colour: White Capacity: 64GB Price: HK\$5999 	<p>包裝隨附內容</p> 
<ul style="list-style-type: none"> Name: Galaxy A42 5G Smartphone Brand: SAMSUNG Colour: White Capacity: 8GB +128GB Price: HK\$3198 	

PHP-OOP and MySQL

1. Create database schema in MySQL



2. Create a table called product

DDL for myshop.product

```
1 CREATE TABLE `product` (  
2   `id` int NOT NULL AUTO_INCREMENT,  
3   `Name` varchar(45) DEFAULT NULL,  
4   `Brand` varchar(45) DEFAULT NULL,  
5   `Colour` varchar(45) DEFAULT NULL,  
6   `Capacity` varchar(45) DEFAULT NULL,  
7   `Price` decimal(10,0) DEFAULT NULL,  
8   `Photo` varchar(45) DEFAULT NULL,  
9   PRIMARY KEY (`id`)  
10 ) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci
```

3. Insert product Data

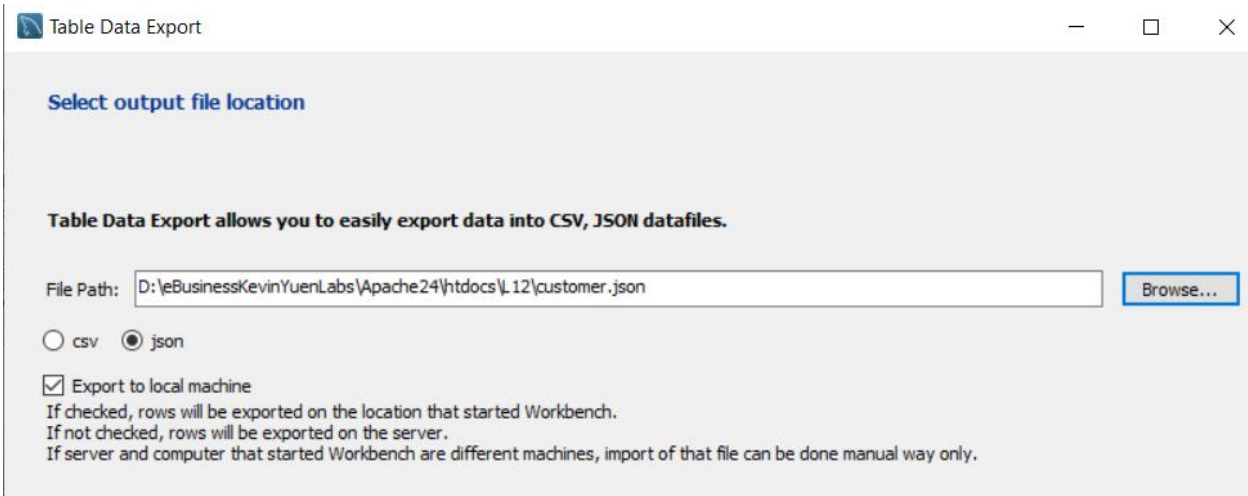
Table: product

Columns:

id int AI PK
Name varchar(45)
Brand varchar(45)
Colour varchar(45)
Capacity varchar(45)
Price decimal(10,0)
Photo varchar(45)

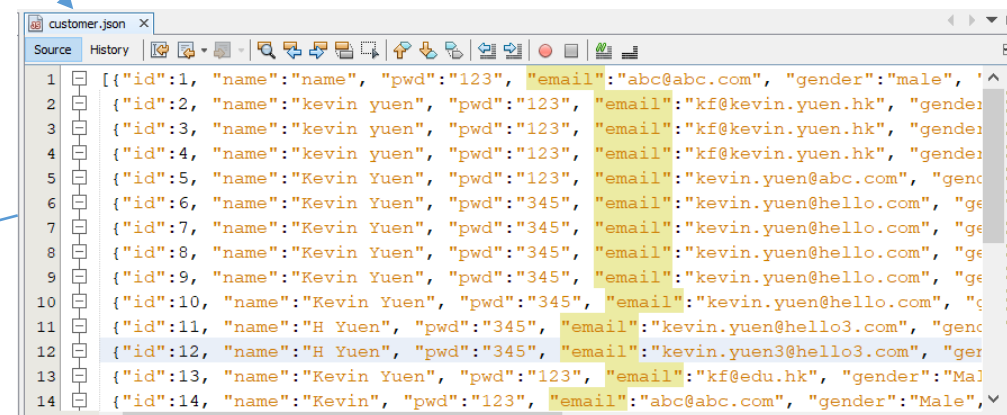
	id	Name	Brand	Colour	Capacity	Price	Photo
▶	1	iPhone 12 mimi	Apple	White	64GB	5999	p1.webp
	2	alaxy A42 5G Smartphone	SAMSUNG	White	8GB +128GB	3198	p2.webp
	3	VIVO 3 China version Smartphone	VIVO	Black	128GB	4198	p3.webp
	4	Mi 10T Lite 5G Smartphone	Mi	Blue	128GB + 6GB	2099	p4.webp
	5	Redmi Note 9 Pro Smartphone	Redmi	White	128GB + 6GB	1899	p5.webp
●	NULL	NULL	NULL	NULL	NULL	NULL	NULL

4. Merge the customer table from previous labs



From the database schema with customer table, right click the customer table, and select Table Data Export Wizard .

Open customer.json to take a look



From the database schema with customer table, right click Tables, and select Table Data Import Wizard .

classLib.php

1. Copy connection.php from previous lab
2. Revise the code

Create a browser log in
javascript for debug purpose

```
//create console.log
function consoleLog($obj)
{
    if (is_array($obj) || is_object($obj))
    {
        echo("<script>console.log('".json_encode($obj)."'</script>");
    }
    else
    {
        echo("<script>console.log('".$obj."'</script>");
    }
}

$serverName = "localhost";
$username = "root";
$password = $yuenLocalPWD; //you may write your password directly
$dbname = "myshop";

$connection = new mysqli($serverName, $username, $password, $dbname);

// Test connection
if (!$connection->connect_error) {

    // Display in Browser Log.
    consoleLog("Connected to MySQL successfully");
    //echo "Connected successfully";

} else {
    //die("Connection failed: " . $connection->connect_error);
    consoleLog(die("Connection failed: " . $connection->connect_error));
}
```


phoneMySQL.php

```

13 <?php
14 // import library
15 include 'classLib.php';
16 include 'connection.php';
17
18 $sqlStatement = "SELECT * FROM product";
19 $result = $connection->query($sqlStatement);
20
21 $productList = array();
22 if ($result->num_rows > 0) {
23     // output data of each row
24     while($row = $result->fetch_assoc()) {
25         //print_r($row);
26         // echo "<br>-----<br>";
27         $phone = new Phone($row['id'], $row['Name'], $row['Brand'],
28             $row['Colour'], $row['Capacity'], $row['Price'], $row['Photo']);
29         array_push($productList, $phone);
30     }
31 } else {
32     consoleLog("No result for product table!");
33 }
34 $connection ->close();
35
36 //echo print_r($productList); // debug
37 echo displayProductList($productList);
38 ?>

```

classLib.php

```

53 function displayProductList($productList)
54 {
55     $html = "<table border = '1'>";
56     $html .= "<thead>";
57     $html .= "<tr>";
58     $html .= "<th>Description</th>";
59     $html .= "<th>Photo </th>";
60     $html .= "</tr> </thead>";
61     $html .= "<tbody>";
62     foreach($productList as $product)
63     {
64         $phoneInfo = $product->getProductInfo();
65         $html .= "<tr><td><ul>";
66         $html .= "<li>Name: " . $phoneInfo['Name'] . "</li>";
67         $html .= "<li>Brand: " . $phoneInfo['Brand'] . "</li>";
68         $html .= "<li>Colour: " . $phoneInfo['Colour'] . "</li>";
69         $html .= "<li>Capacity: " . $phoneInfo['Capacity'] . "</li>";
70         $html .= "<li>Price: HKD$ " . $phoneInfo['Price'] . "</li>";
71         $html .= "</ul></td>";
72         $html .= "<td><img src='img\\\". $phoneInfo['Photo']";
73         $html .= "' width='300' height='200'/> </td></tr>";
74     }
75     $html .= "</tbody></table>";
76     return $html;

```

Product Display	
localhost/L12/phoneMySQL.php	
Description	Photo
<ul style="list-style-type: none"> Name: iPhone 12 mimi Brand: Apple Colour: White Capacity: 64GB Price: HKD\$5999 	
<ul style="list-style-type: none"> Name: alaxy A42 5G Smartphone Brand: SAMSUNG Colour: White Capacity: 8GB +128GB Price: HKD\$3198 	
<ul style="list-style-type: none"> Name: VIVO 3 China version Smartphone Brand: VIVO Colour: Black Capacity: 128GB Price: HKD\$4198 	
<ul style="list-style-type: none"> Name: Mi 10T Lite 5G Smartphone Brand: Mi Colour: Blue Capacity: 128GB + 6GB Price: HKD\$2099 	
<ul style="list-style-type: none"> Name: Redmi Note 9 Pro Smartphone Brand: Redmi Colour: White Capacity: 128GB + 6GB Price: HKD\$1899 	

Congratulations!

Composer



A Dependency Manager for PHP

Latest: 2.0.4 ([changelog](#))

Composer 2.0 is now available! [Read our announcement!](#)

[Getting Started](#)

[Download](#)

[Documentation](#)

[Browse Packages](#)

[Issues](#)

[GitHub](#)

<https://getcomposer.org/>

Composer is a tool for dependency management in PHP. It allows you to declare the libraries your project depends on and it will manage (install/update) them for you.

Suppose:

1. You have a project that depends on a number of libraries.
2. Some of those libraries depend on other libraries.

Composer:

1. Enables you to declare the libraries you depend on.
2. Finds out which versions of which packages can and need to be installed, and installs them (meaning it downloads them into your project).
3. You can update all your dependencies in one command.

Install composer

Download the installer to the current directory

```
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
```

Verify the installer SHA-384

- ```
php -r "if (hash_file('sha384', 'composer-setup.php') ===
'c31c1e292ad7be5f49291169c0ac8f683499edddcfd4e42232982d0fd193004208a58ff6f353fde0012d35fdd72bc394')
{ echo 'Installer verified'; } else { echo 'Installer corrupt'; unlink('composer-setup.php'); } echo PHP_EOL;"
```

Run the installer

- ```
php composer-setup.php
```

Remove the installer

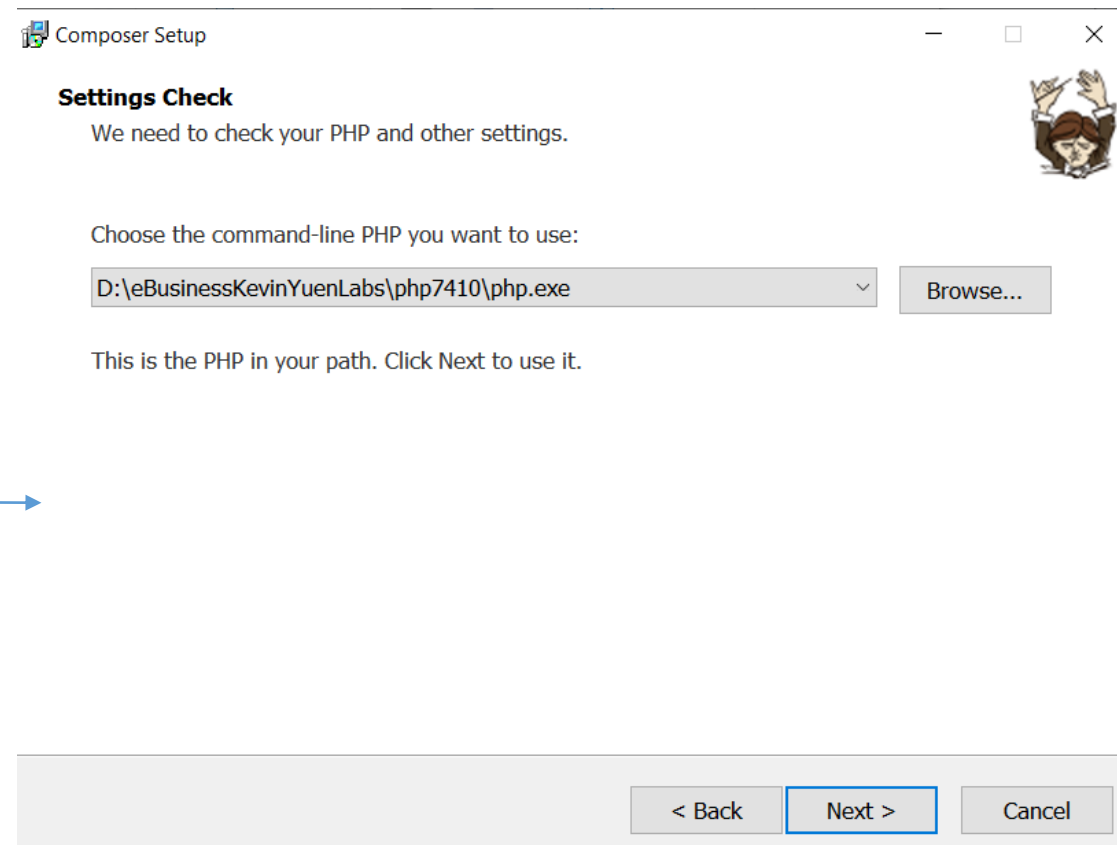
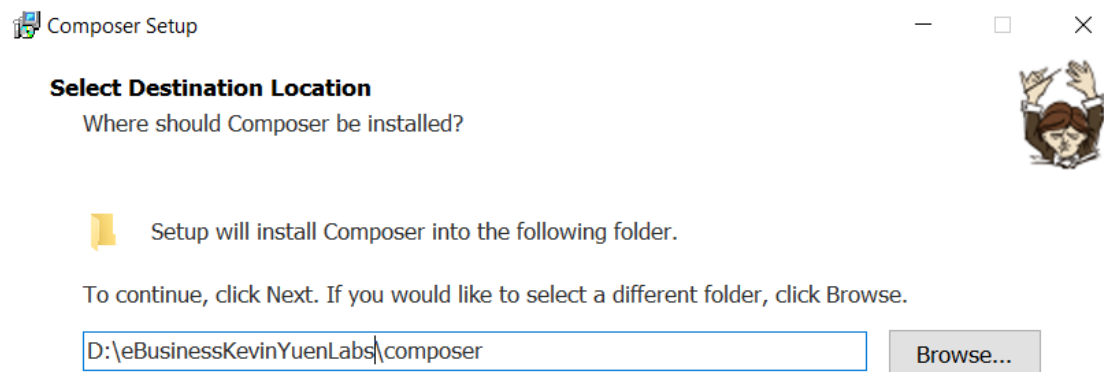
- ```
php -r "unlink('composer-setup.php');"
```

For window,

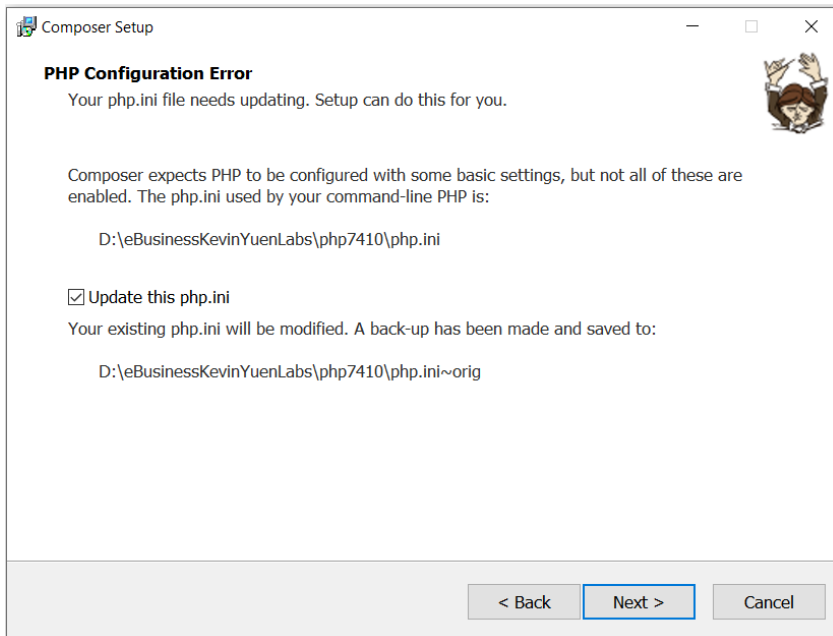
<https://getcomposer.org/Composer-Setup.exe>

<https://getcomposer.org/download/>

## Choose developer's mode → next







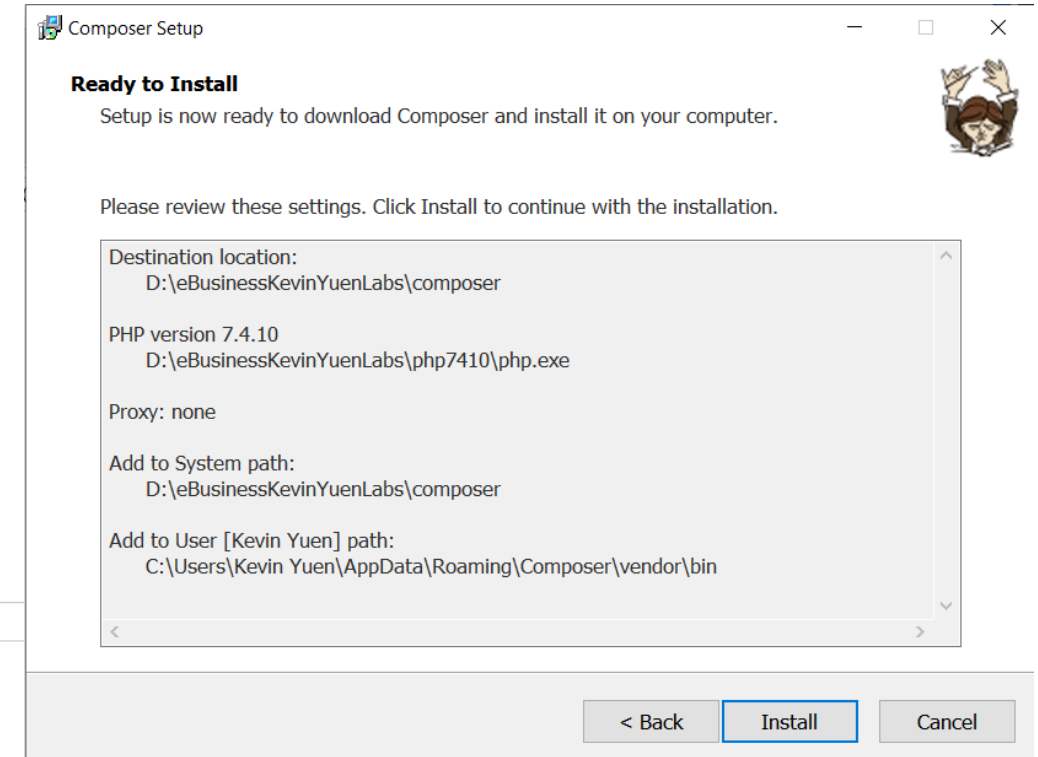
No proxy server

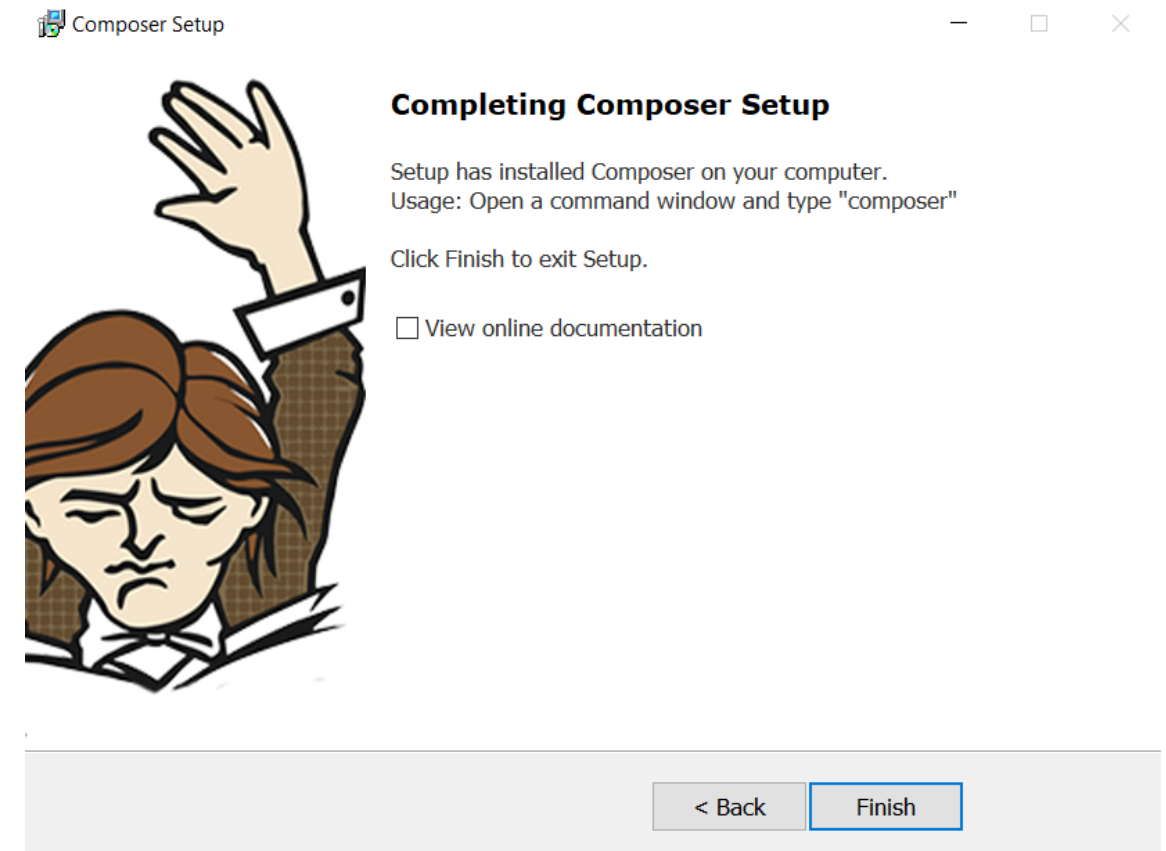
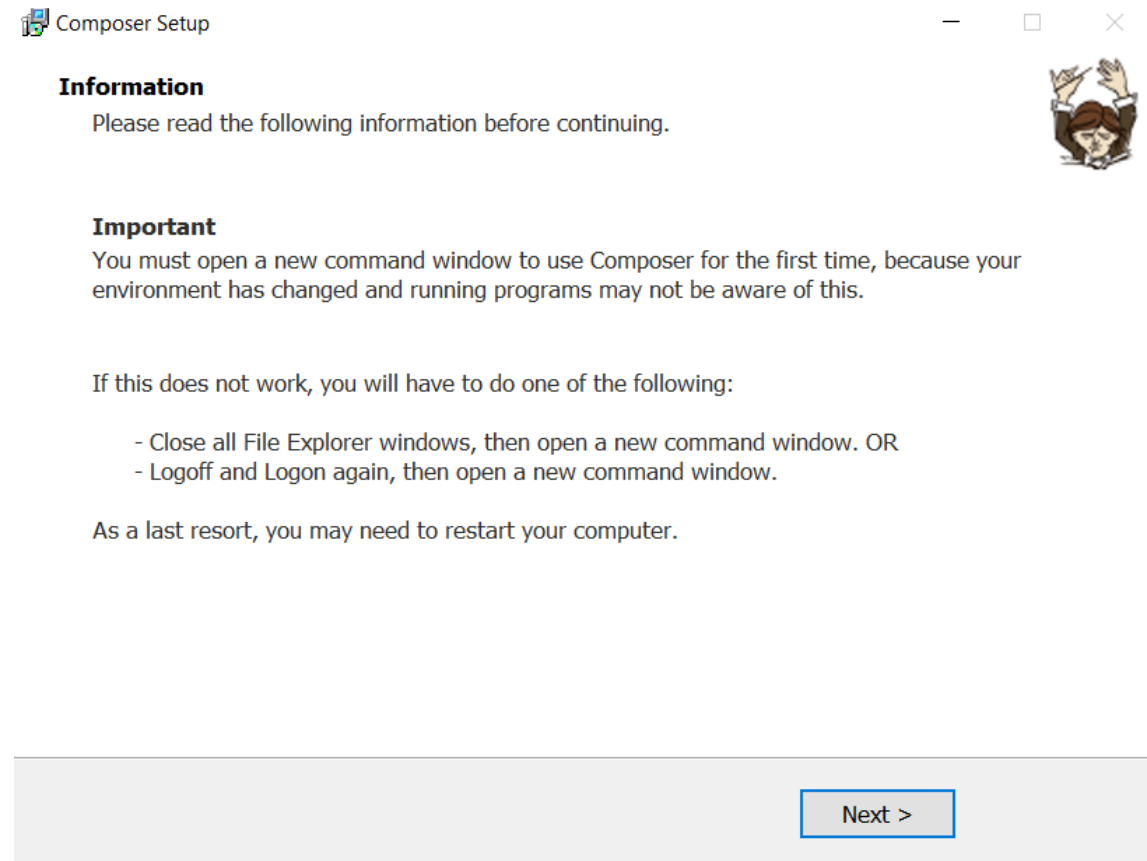
**Proxy Settings**


Choose if you need to use a proxy.

☐ Use a proxy server to connect to internet

Enter proxy url







Command Prompt

```
C:\>ipconfig /flushdns

Windows IP Configuration
: Successful flushing of the DNS resolver cache.
```

```
C:\>composer -v
```

```
Composer version 2.0.4 2020-10-30 22:39:11
```

### Usage:

```
command [options] [arguments]
```

Options:

```
-h, --help Display this help message
-q, --quiet Do not output any message
-V, --version Display this application version
 --ansi Force ANSI output
 --no-ansi Disable ANSI output
-n, --no-interaction Do not ask any interactive question
```

# Congratulations!

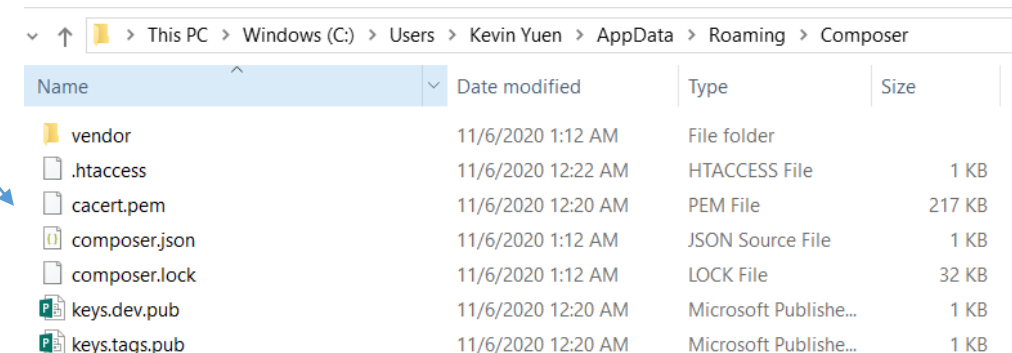
## Method 1: Install laravel

```
C:\>composer global require laravel/installer
Changed current directory to C:/Users/Kevin Yuen/AppData/Roaming/Composer
Using version ^4.1 for laravel/installer
./composer.json has been created
Running composer update laravel/installer
Loading composer repositories with package information
Updating dependencies
Lock file operations: 12 installs, 0 updates, 0 removals
- Locking laravel/installer (v4.1.0)
- Locking psr/container (1.0.0)
- Locking symfony/console (v5.1.8)
- Locking symfony/polyfill-ctype (v1.20.0)
- Locking symfony/polyfill-intl-grapheme (v1.20.0)
- Locking symfony/polyfill-intl-normalizer (v1.20.0)
- Locking symfony/polyfill-mbstring (v1.20.0)
- Locking symfony/polyfill-php73 (v1.20.0)
- Locking symfony/polyfill-php80 (v1.20.0)
```

```
6 package suggestions were added by new dependencies, use `composer suggest` to see details.
Generating autoload files
10 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
```

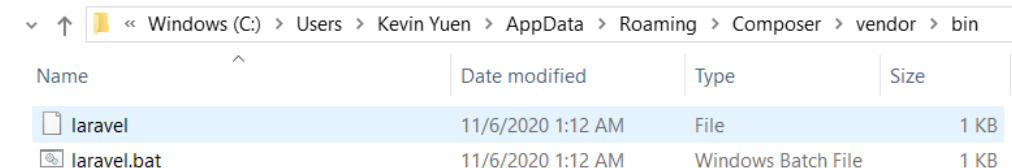
```
C:\>
```

download the Laravel installer using Composer:  
composer global require laravel/installer



| This PC > Windows (C:) > Users > Kevin Yuen > AppData > Roaming > Composer |                    |                       |        |  |
|----------------------------------------------------------------------------|--------------------|-----------------------|--------|--|
| Name                                                                       | Date modified      | Type                  | Size   |  |
| vendor                                                                     | 11/6/2020 1:12 AM  | File folder           |        |  |
| .htaccess                                                                  | 11/6/2020 12:22 AM | HTACCESS File         | 1 KB   |  |
| cacert.pem                                                                 | 11/6/2020 12:20 AM | PEM File              | 217 KB |  |
| composer.json                                                              | 11/6/2020 1:12 AM  | JSON Source File      | 1 KB   |  |
| composer.lock                                                              | 11/6/2020 1:12 AM  | LOCK File             | 32 KB  |  |
| keys.dev.pub                                                               | 11/6/2020 12:20 AM | Microsoft Publishe... | 1 KB   |  |
| keys.tags.pub                                                              | 11/6/2020 12:20 AM | Microsoft Publishe... | 1 KB   |  |

C:\Users\Kevin Yuen\AppData\Roaming\Composer\vendor\bin



| << Windows (C:) > Users > Kevin Yuen > AppData > Roaming > Composer > vendor > bin |                   |                    |      |  |
|------------------------------------------------------------------------------------|-------------------|--------------------|------|--|
| Name                                                                               | Date modified     | Type               | Size |  |
| laravel                                                                            | 11/6/2020 1:12 AM | File               | 1 KB |  |
| laravel.bat                                                                        | 11/6/2020 1:12 AM | Windows Batch File | 1 KB |  |

# Test laravel Installation

```
C:\>laravel -v
Laravel Installer 4.1.0

Usage:
 command [options] [arguments]

Options:
 -h, --help Display this help message
 -q, --quiet Do not output any message
 -V, --version Display this application version
 --ansi Force ANSI output
 --no-ansi Disable ANSI output
 -n, --no-interaction Do not ask any interactive question
 -v|vv|vvv, --verbose Increase the verbosity of messages:
 1 for normal output, 2 for more verbose output and 3 for
 debug

Available commands:
 help Displays help for a command
 list Lists commands
 new Create a new Laravel application

C:\>
```

laravel new myBlog in project folder

```
D:\eBusinessKevinYuenLabs\laravel>laravel new myBlog
```



Creating a "laravel/laravel" project at "./myBlog"



### Problem 1

- laravel/framework[v8.12.0, ..., 8.x-dev] require league/flysystem ^1.1 -> satisfiable by league/flysystem[1.1.0, ..., 1.x-dev].
- league/flysystem[1.1.0, ..., 1.x-dev] require ext-fileinfo \* -> it is missing from your system. Install or enable PHP's fileinfo extension.

- Root composer.json requires laravel/framework ^8.12 -> satisfiable by laravel/framework[v8.12.0, ..., 8.x-dev]

To enable extensions, verify that they are enabled in your .ini files:

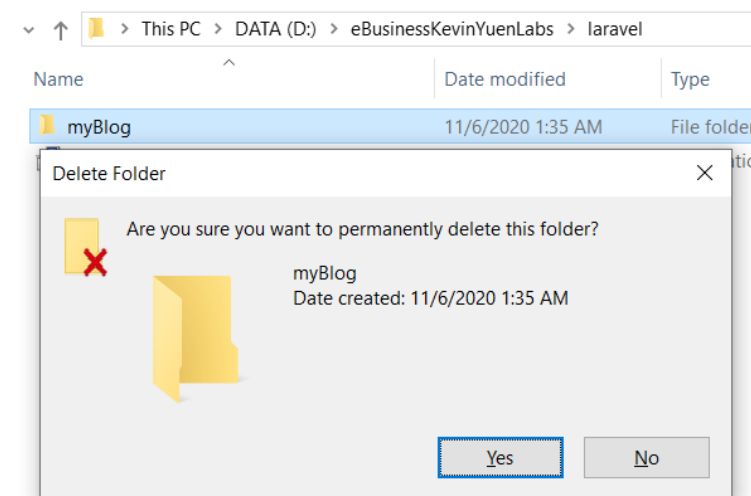
- D:\eBusinessKevinYuenLabs\php7410\php.ini

You can also run `php --ini` inside terminal to see which files are used by PHP in CLI mode.

D:\eBusinessKevinYuenLabs\laravel>

→ php.ini      Extension=php\_fileinfo.dll

```
896 ; extension=/path/to/extension/mysql.so
897 extension=php_mysql.dll
898 extension=php_fileinfo.dll
899 extension_dir = "D:\eBusinessKevinYuenLabs\php7410\ext"
900 ;
```



Try again

laravel new myBlog

```
D:\eBusinessKevinYuenLabs\laravel>laravel new myBlog

Laravel

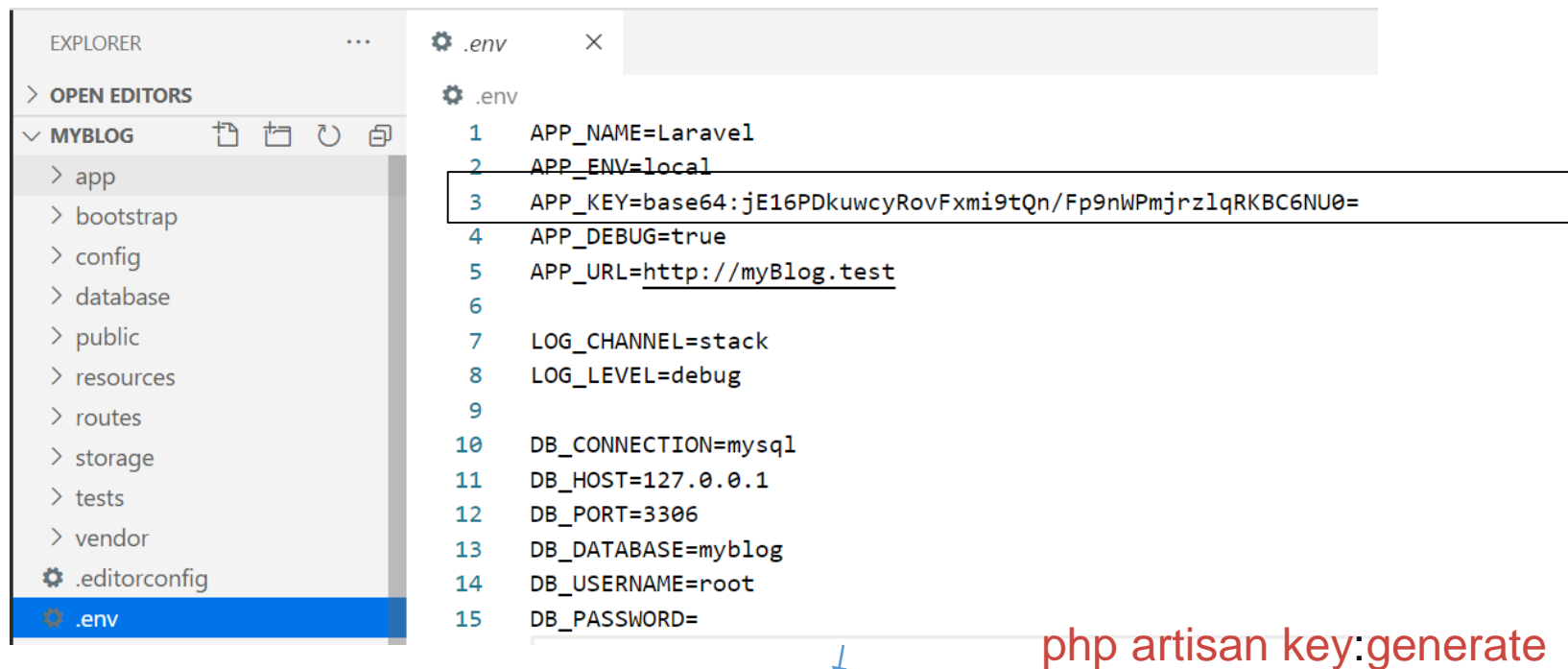
Creating a "laravel/laravel" project at "./myBlog"
Installing laravel/laravel (v8.4.0)
- Installing laravel/laravel (v8.4.0): Extracting archive
Created project in D:\eBusinessKevinYuenLabs\laravel\myBlog
```

...Long messages..., and finally

```
Package manifest generated successfully.
71 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
> @php artisan key:generate --ansi
Application key set successfully.

Application ready! Build something amazing.

D:\eBusinessKevinYuenLabs\laravel>
```



The screenshot shows the VS Code interface. On the left, the Explorer sidebar displays the project structure for 'MYBLOG', with the '.env' file selected at the bottom. The main editor area shows the contents of the '.env' file, which includes configuration for Laravel. Line 3, 'APP\_KEY=base64:jE16PDkuwcyRovFxmI9tQn/Fp9nWPmjrz1qRKBC6NU0=', is highlighted with a black box. A red arrow points from this line down to the terminal command below.

```
.env
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:jE16PDkuwcyRovFxmI9tQn/Fp9nWPmjrz1qRKBC6NU0=
4 APP_DEBUG=true
5 APP_URL=http://myBlog.test
6
7 LOG_CHANNEL=stack
8 LOG_LEVEL=debug
9
10 DB_CONNECTION=mysql
11 DB_HOST=127.0.0.1
12 DB_PORT=3306
13 DB_DATABASE=myblog
14 DB_USERNAME=root
15 DB_PASSWORD=
```

php artisan key:generate

```
D:\eBusinessKevinYuenLabs\laravel\myBlog>php artisan key:generate
Application key set successfully.
```

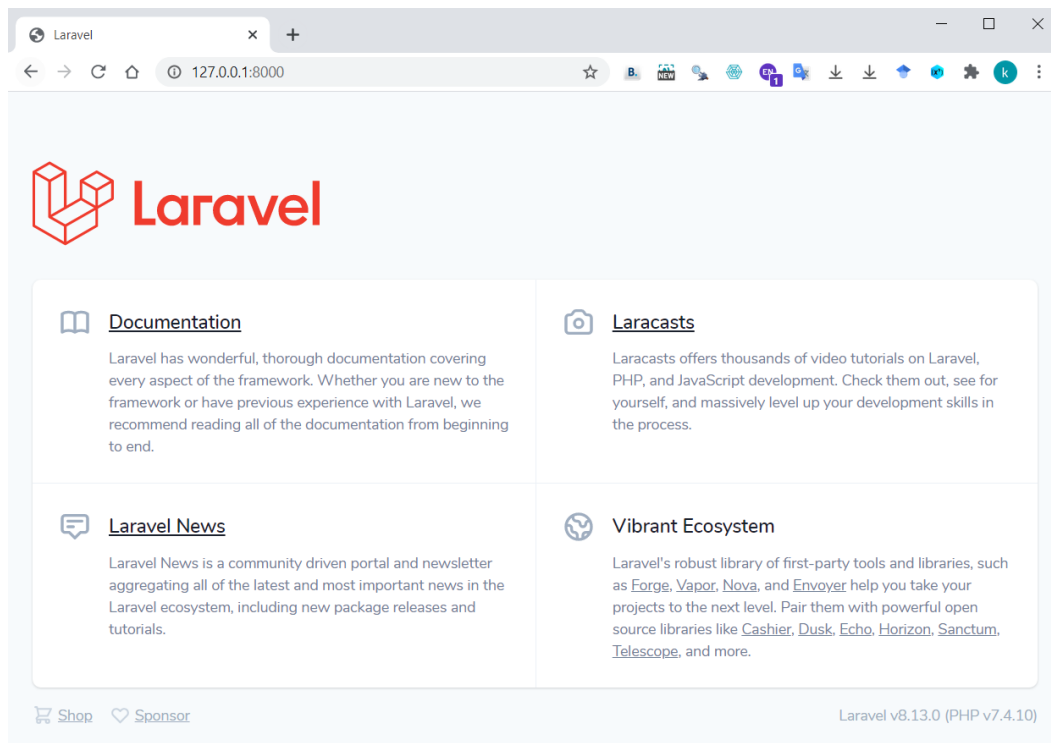
Change key

```
3 APP_KEY=base64:Mcb5ToXTQ3XzTjseeLh6vO2hzWOWABM1XGV2vfjU/8Y=
```

php artisan serve under the new folder, myBlog.

```
D:\eBusinessKevinYuenLabs\laravel>cd myBlog

D:\eBusinessKevinYuenLabs\laravel\myBlog>php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
[Fri Nov 6 01:57:52 2020] PHP 7.4.10 Development Server (http://127.0.0.1:8000)
```



```
D:\eBusinessKevinYuenLabs\laravel\myBlog>php artisan -v
Laravel Framework 8.13.0

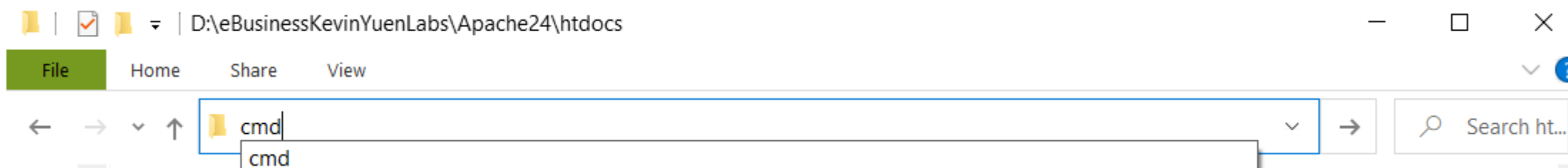
Usage:
 command [options] [arguments]

Options:
 -h, --help Display this help message
 -q, --quiet Do not output any message
 -V, --version Display this application version
 --ansi Force ANSI output
 --no-ansi Disable ANSI output
```

You may use apache http server by deploying to the directory  
D:\eBusinessKevinYuenLabs\Apache24\htdocs

## Method 2: composer create-project laravel/laravel ...

1. Move to the path D:\eBusinessKevinYuenLabs\Apache24\htdocs
2. Type cmd



3. Command prompt with current location is display

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.18363.1198]
(c) 2019 Microsoft Corporation. All rights reserved.

D:\eBusinessKevinYuenLabs\Apache24\htdocs>
```



## 4. Type

composer create-project laravel/laravel --prefer-dist **MyLaravelProject** dev-develop

```
Microsoft Windows [Version 10.0.18363.1198]
(c) 2019 Microsoft Corporation. All rights reserved.

D:\eBusinessKevinYuenLabs\Apache24\htdocs>composer create-project laravel/laravel --prefer-dist MyLaravelProject dev-develop
Creating a "laravel/laravel" project at "./MyLaravelProject"
Installing laravel/laravel (dev-develop e253d4ebb9dd147bbcd633efd0b3018a1955f9f)
 - Downloading laravel/laravel (dev-develop e253d4e)
 - Installing laravel/laravel (dev-develop e253d4e): Extracting archive
Created project in D:\eBusinessKevinYuenLabs\Apache24\htdocs\MyLaravelProject
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
```

....

```
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: fideloper/proxy
Discovered Package: fruitcake/laravel-cors
Discovered Package: laravel/tinker
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Package manifest generated successfully.
71 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
> @php artisan key:generate --ansi
Application key set successfully.
```

....Successfully!

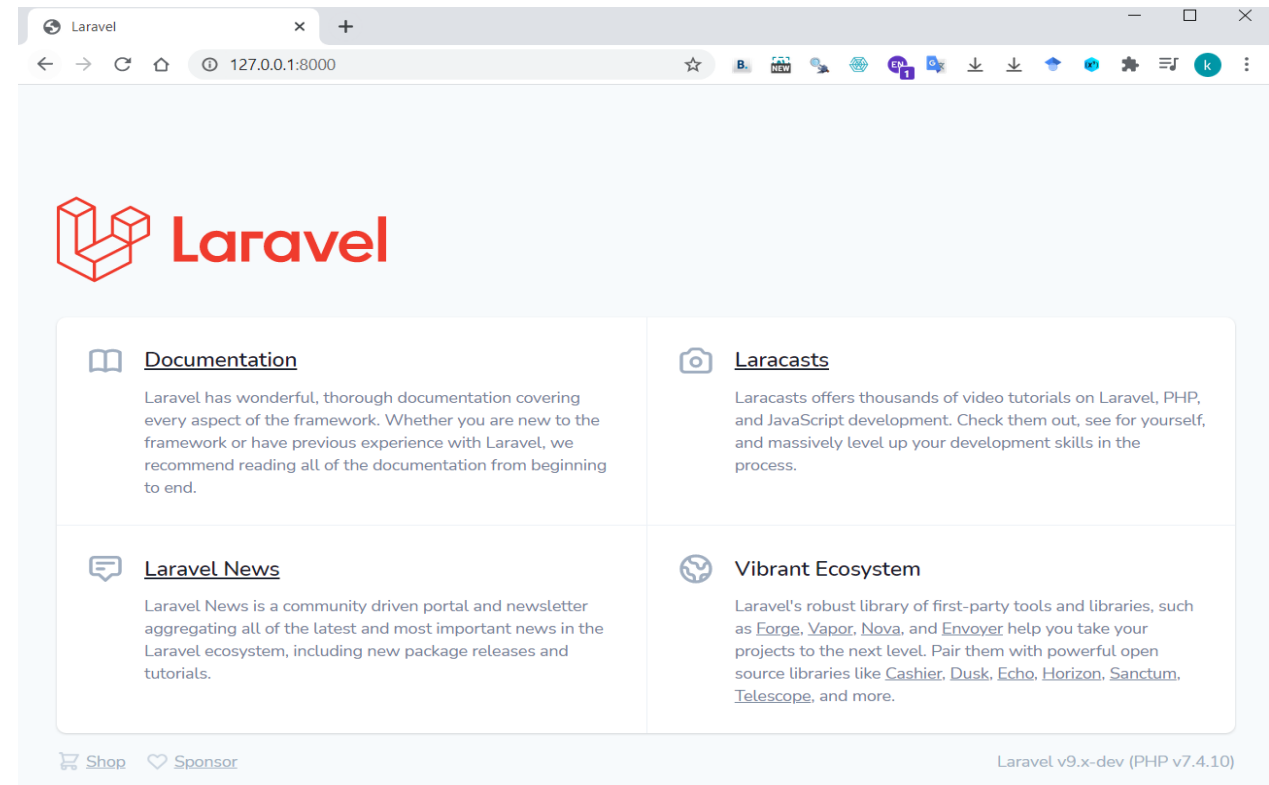
5. Go the project just created, e.g. `cd MyLaravelProject`

6. Run `php artisan serve`

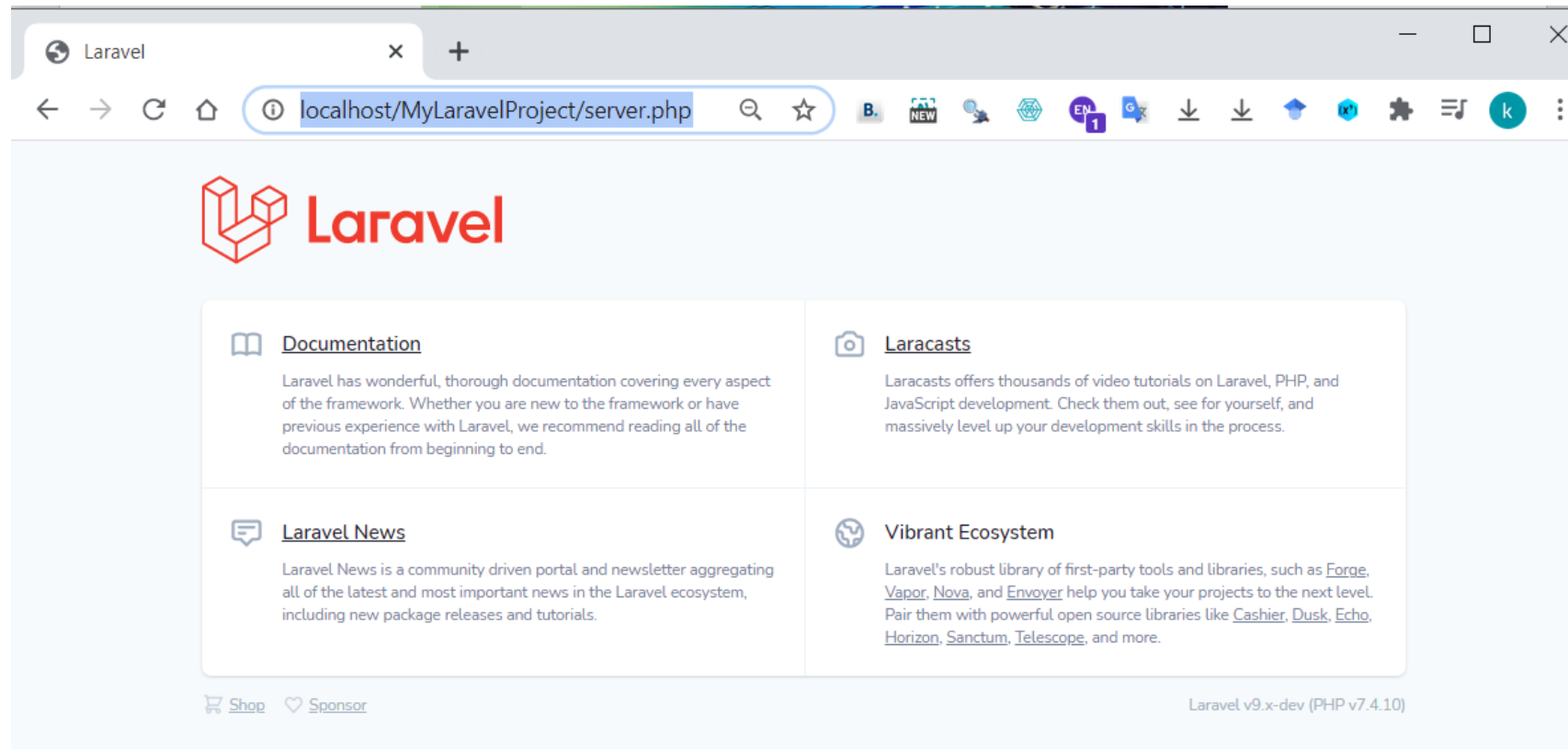
```
D:\eBusinessKevinYuenLabs\Apache24\htdocs>cd MyLaravelProject

D:\eBusinessKevinYuenLabs\Apache24\htdocs\MyLaravelProject>php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
[Tue Nov 24 01:59:35 2020] PHP 7.4.10 Development Server (http://127.0.0.1:8000) started
```

7. Use web browser to open <http://127.0.0.1:8000>



Or open by Apache http server, <http://localhost/MyLaravelProject/server.php>

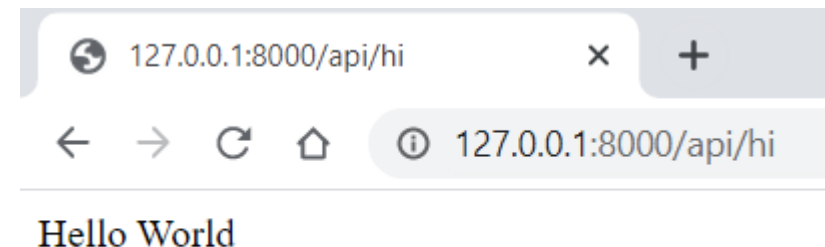


8. Use netbeans or another editor of your choice to open the project folder which is just created.

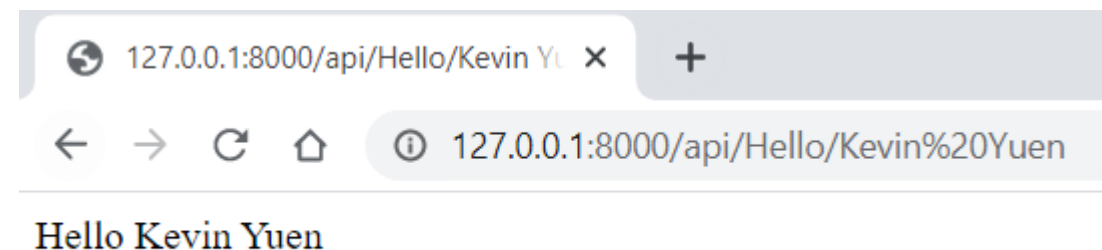
## Basic Routing

Api.php

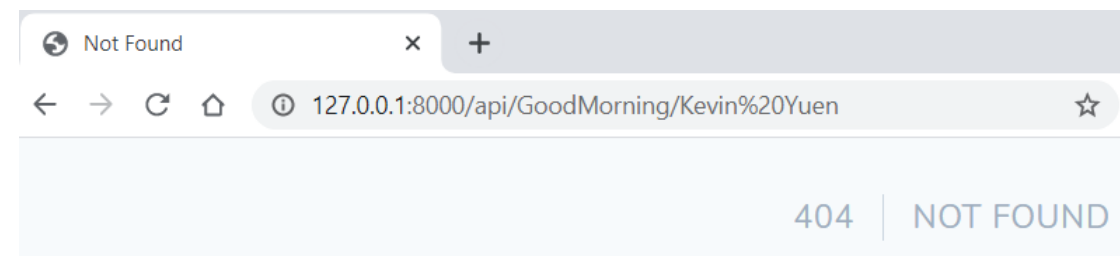
```
21 Route::get('hi', function () {
22 return 'Hello World';
23 });
```



```
25 Route::get('/Hello/{name?}', function ($name) {
26 return 'Hello ' . $name;
27 });
```



```
29 Route::get('/GoodMorning/{name?}', function ($name = null) {
30 return 'Good Morning ' . $name;
 })->where('name', '[a-zA-Z]+');
```



# controller

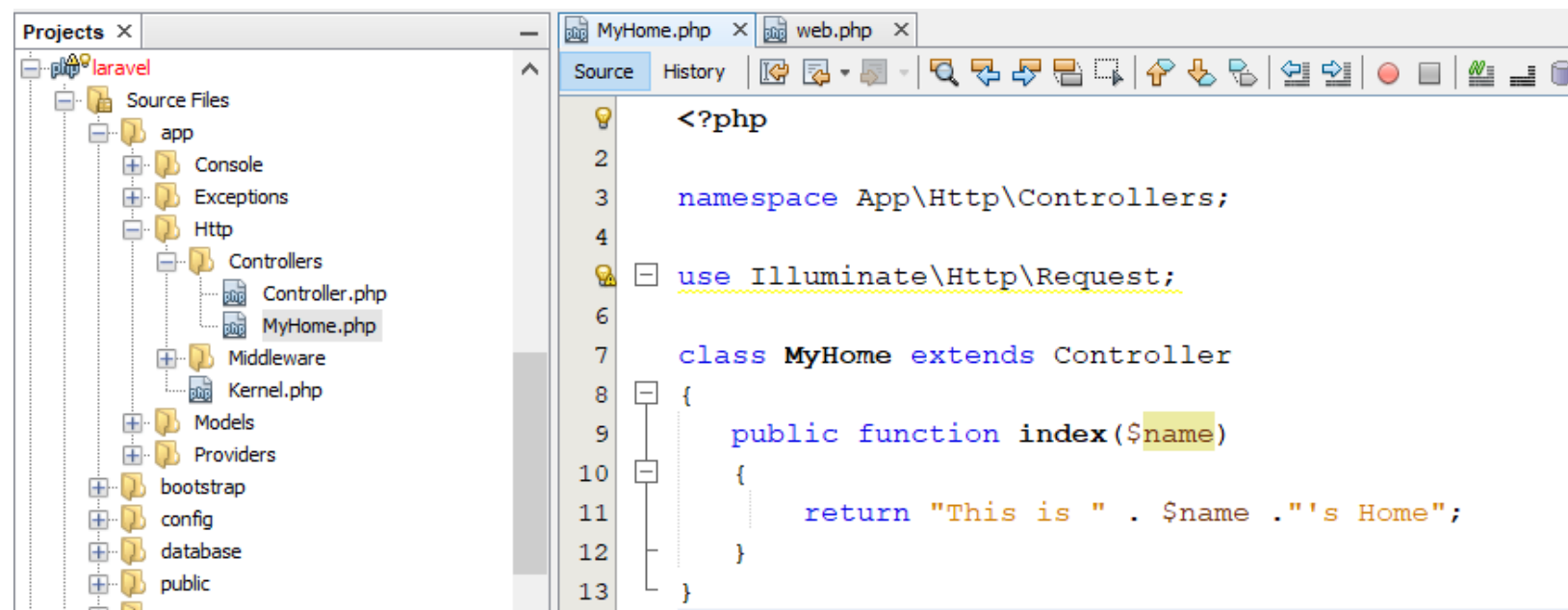
1. php artisan make:controller MyHome

2. php artisan serve

```
D:\eBusinessKevinYuenLabs\Apache24\htdocs\MyLaravelProject>php artisan make:controller MyHome
Controller created successfully.

D:\eBusinessKevinYuenLabs\Apache24\htdocs\MyLaravelProject>php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
[Tue Nov 24 03:03:59 2020] PHP 7.4.10 Development Server (http://127.0.0.1:8000) started
```

3. Add code as shown

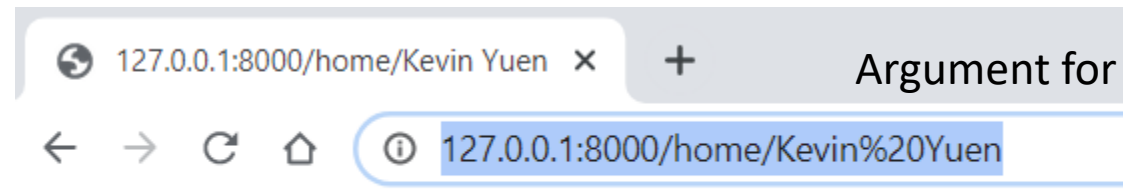




routes/ web.php

Pass parameter

```
20
21 Route::get('/home/{name}', [App\Http\Controllers\MyHome::class, 'index'])->name('home.index');
22
```

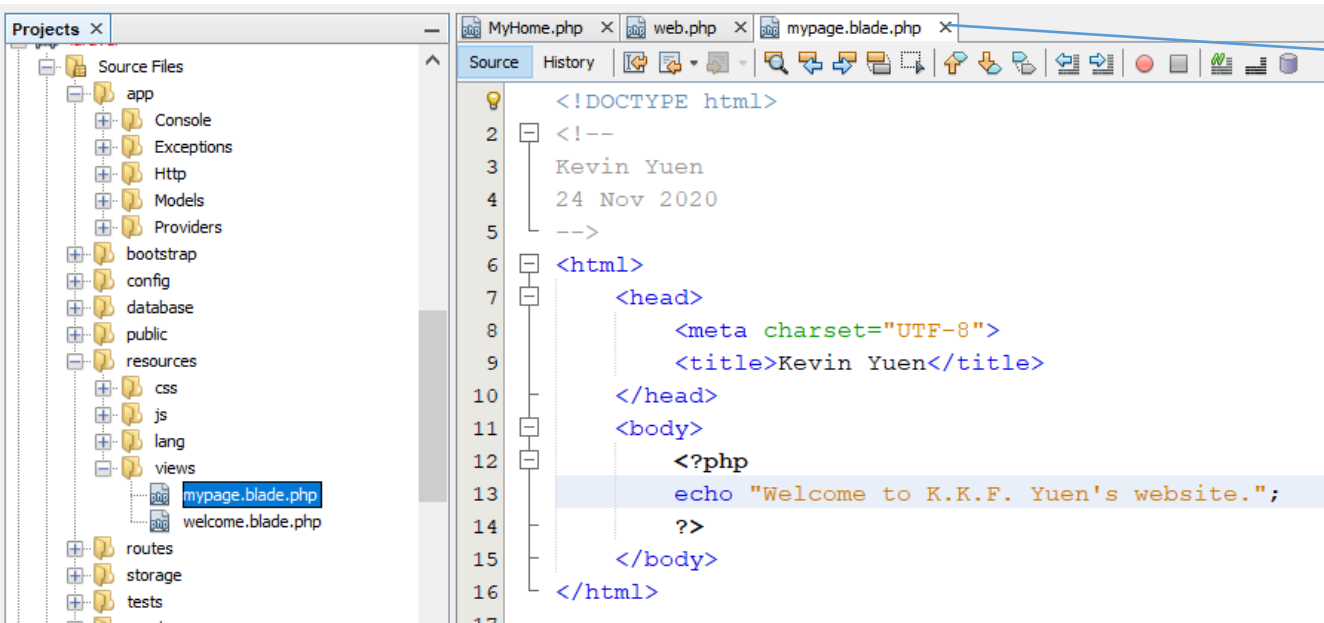


Argument for the parameter

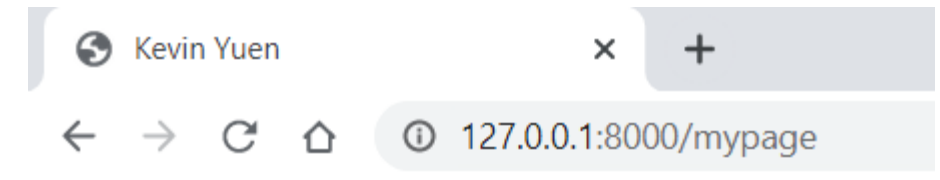
This is Kevin Yuen's Home

# view

## Web.php



```
Route::get('/mypage', function () {
 return view('mypage');
});
```



1. Copy phoneObjects.php and classlib.php in L12 to resources/views of the current project;
2. Rename phoneObjects.php as phoneObjects.blade.php;
3. Copy img in L12 to public/img of the current project;
4. Change the codes for phoneObjects.blade.php as below.

```
15 // root in the "public" folder
16 include '../resources/views/classLib.php';
```

```
68 $html .= "<td><img src='../img\\" . $phoneInfo['Photo'];
69 $html .= "' width='300' height='200' /> </td></tr>";
```

5. Add the code below for web.php;

```
28 Route::get('/phoneObjects', function () {
29 return view('phoneObjects');
});
```

L12

127.0.0.1:8000/phoneObjects

Key=Brand, Value=Apple  
Key=Colour, Value=White  
Key=Capacity, Value=64GB  
Key=Price, Value=HK\$5999  
Key=Photo, Value=p1.webp

Key=ID, Value=2  
Key=Name, Value=Galaxy A42 5G Smartphone  
Key=Brand, Value=SAMSUNG  
Key=Colour, Value=White  
Key=Capacity, Value=8GB +128GB  
Key=Price, Value=HK\$3198  
Key=Photo, Value=p2.webp

| Description                                                                                                                                                                            | Photo                                                                                             |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>Name: iPhone 12 mini</li> <li>Brand: Apple</li> <li>Colour: White</li> <li>Capacity: 64GB</li> <li>Price: HK\$5999</li> </ul>                   | <p>包裝隨附內容</p>  |
| <ul style="list-style-type: none"> <li>Name: Galaxy A42 5G Smartphone</li> <li>Brand: SAMSUNG</li> <li>Colour: White</li> <li>Capacity: 8GB +128GB</li> <li>Price: HK\$3198</li> </ul> |               |

1. Copy connection.php and phoneMySQL.php in L12 to resources/views of the current project;
2. Rename phoneMySQL.php as phoneMySQL.blade.php;
3. Copy img in L12 to public/img of the current project if the tasks are not finished in previous exercise;
4. Change the codes for phoneMySQL.blade.php as below.

```
15 | | include '../resources/views/classLib.php';
16 | | include '../resources/views/connection.php';
```

5. Change the code for the include connection.php, if needed

```
include '../resources/views/pwd.php';
```

6. Add the code below for web.php;

```
32 | Route::get('/phoneMySQL', function () {
33 | return view('phoneMySQL');
 | });
```

| Product Display                                                                                                                                                                           |                                                                                       |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| 127.0.0.1:8000/phoneMySQL                                                                                                                                                                 |                                                                                       |
| Description                                                                                                                                                                               | Photo                                                                                 |
| <ul style="list-style-type: none"> <li>Name: iPhone 12 mimi</li> <li>Brand: Apple</li> <li>Colour: White</li> <li>Capacity: 64GB</li> <li>Price: HKD\$5999</li> </ul>                     |    |
| <ul style="list-style-type: none"> <li>Name: alaxy A42 5G Smartphone</li> <li>Brand: SAMSUNG</li> <li>Colour: White</li> <li>Capacity: 8GB +128GB</li> <li>Price: HKD\$3198</li> </ul>    |    |
| <ul style="list-style-type: none"> <li>Name: VIVO 3 China version Smartphone</li> <li>Brand: VIVO</li> <li>Colour: Black</li> <li>Capacity: 128GB</li> <li>Price: HKD\$4198</li> </ul>    |    |
| <ul style="list-style-type: none"> <li>Name: Mi 10T Lite 5G Smartphone</li> <li>Brand: Mi</li> <li>Colour: Blue</li> <li>Capacity: 128GB + 6GB</li> <li>Price: HKD\$2099</li> </ul>       |   |
| <ul style="list-style-type: none"> <li>Name: Redmi Note 9 Pro Smartphone</li> <li>Brand: Redmi</li> <li>Colour: White</li> <li>Capacity: 128GB + 6GB</li> <li>Price: HKD\$1899</li> </ul> |  |

## Reference

- <https://laravel.com/docs/master>
- <https://www.tutorialspoint.com/laravel/index.htm>



Q&A  
Thank you