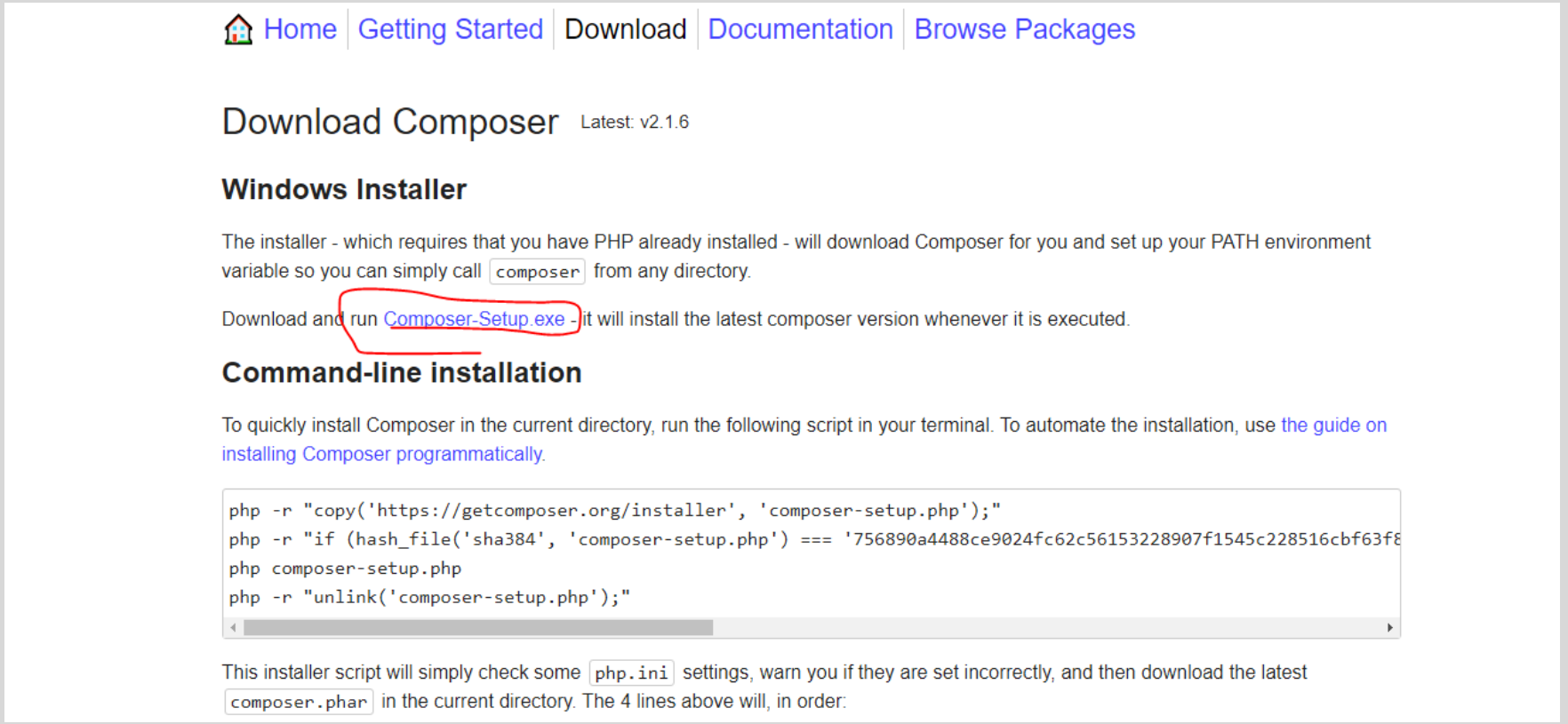
# LARAVEL

Presented By: Miss Tazeen Zafar

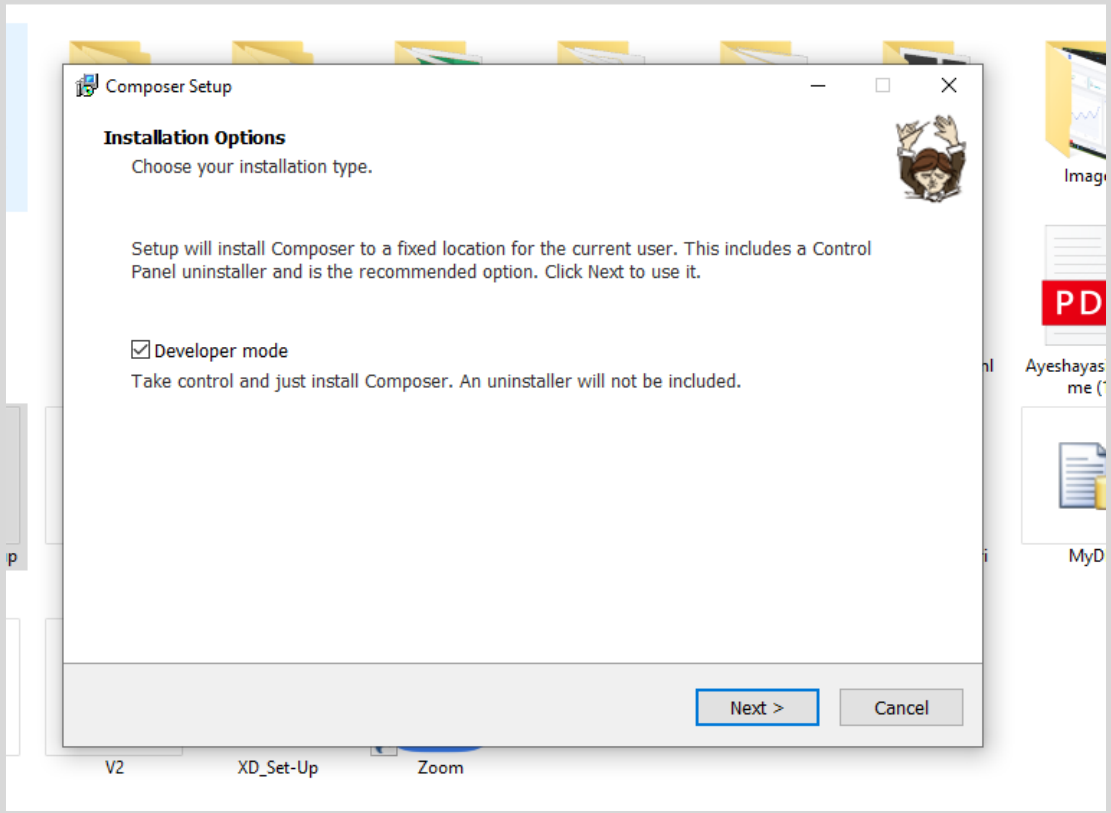
# COMPOSER INSTALL

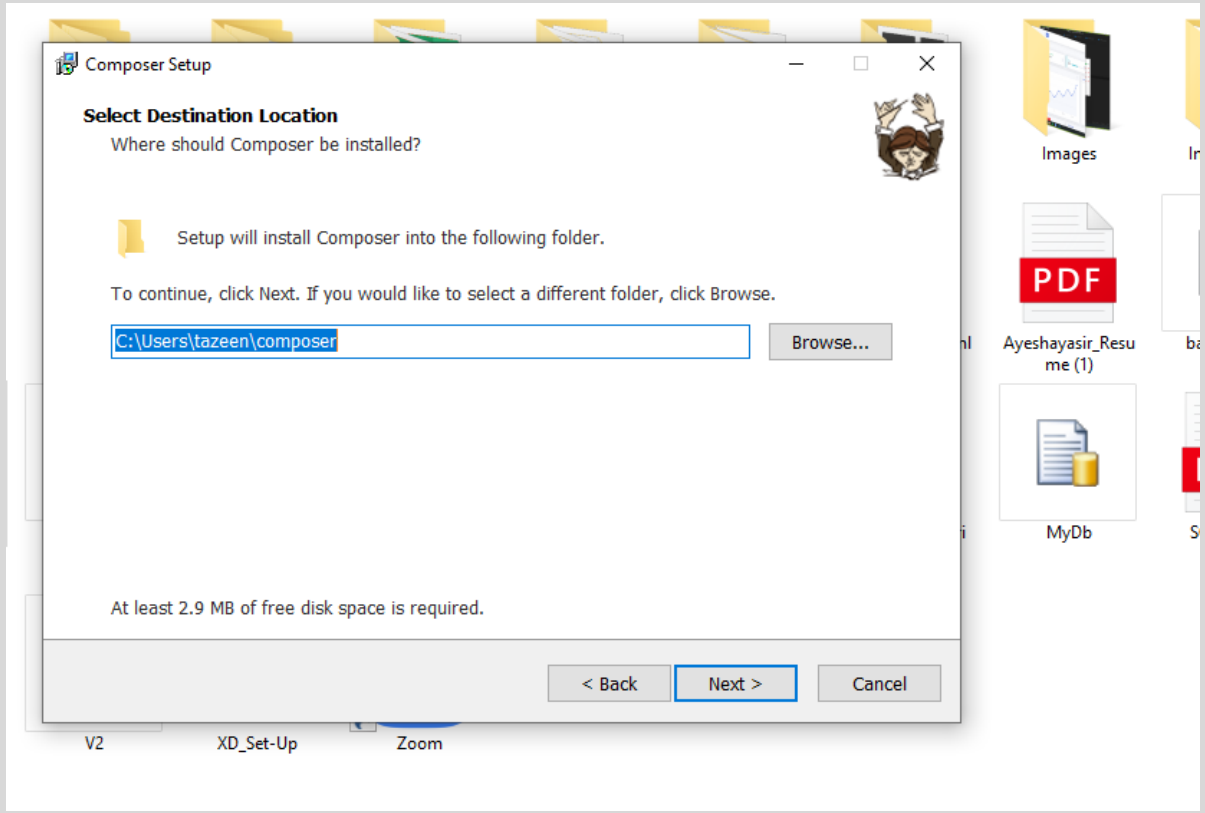
- First of all download Composer from google



# COMPOSER

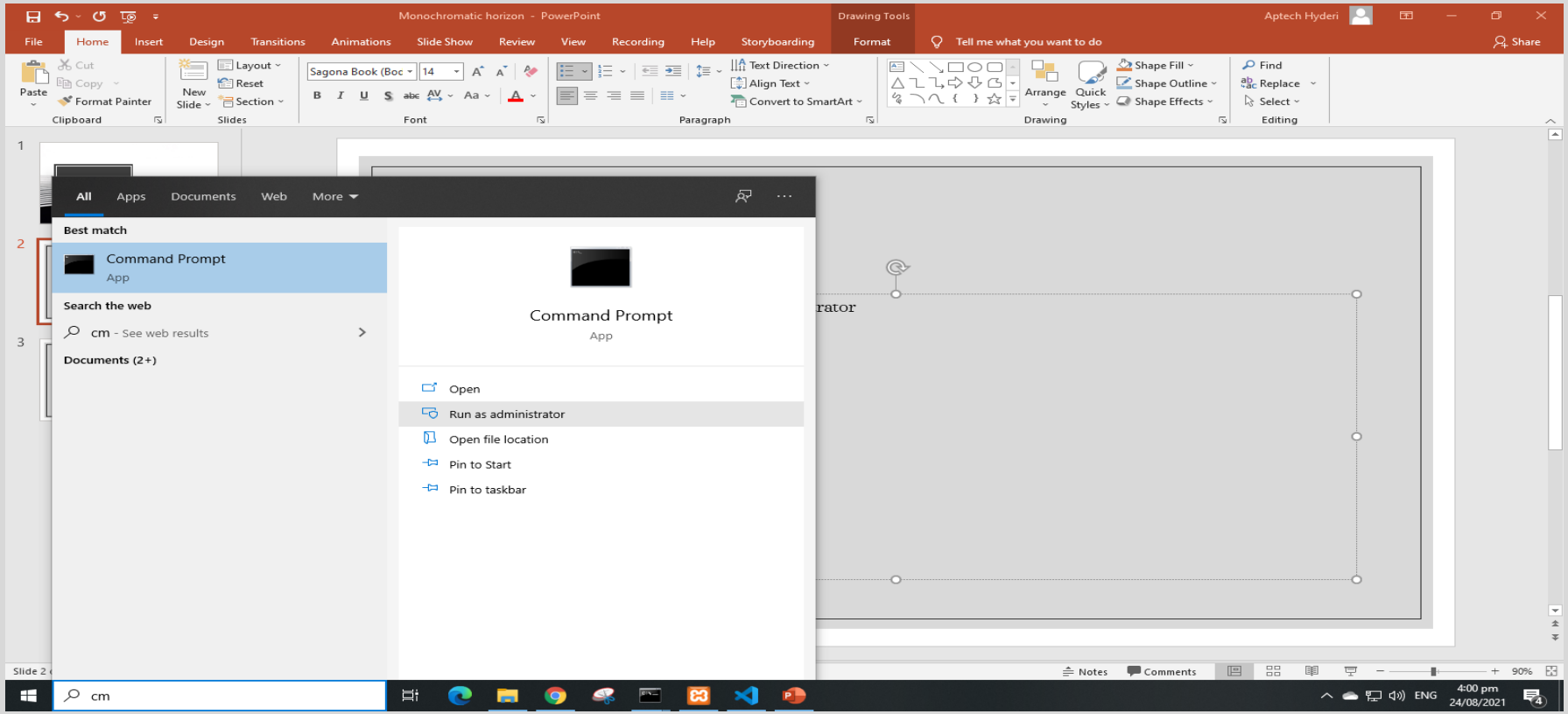
Click Developer Mode and then, next next next.



# Open CMD screen:

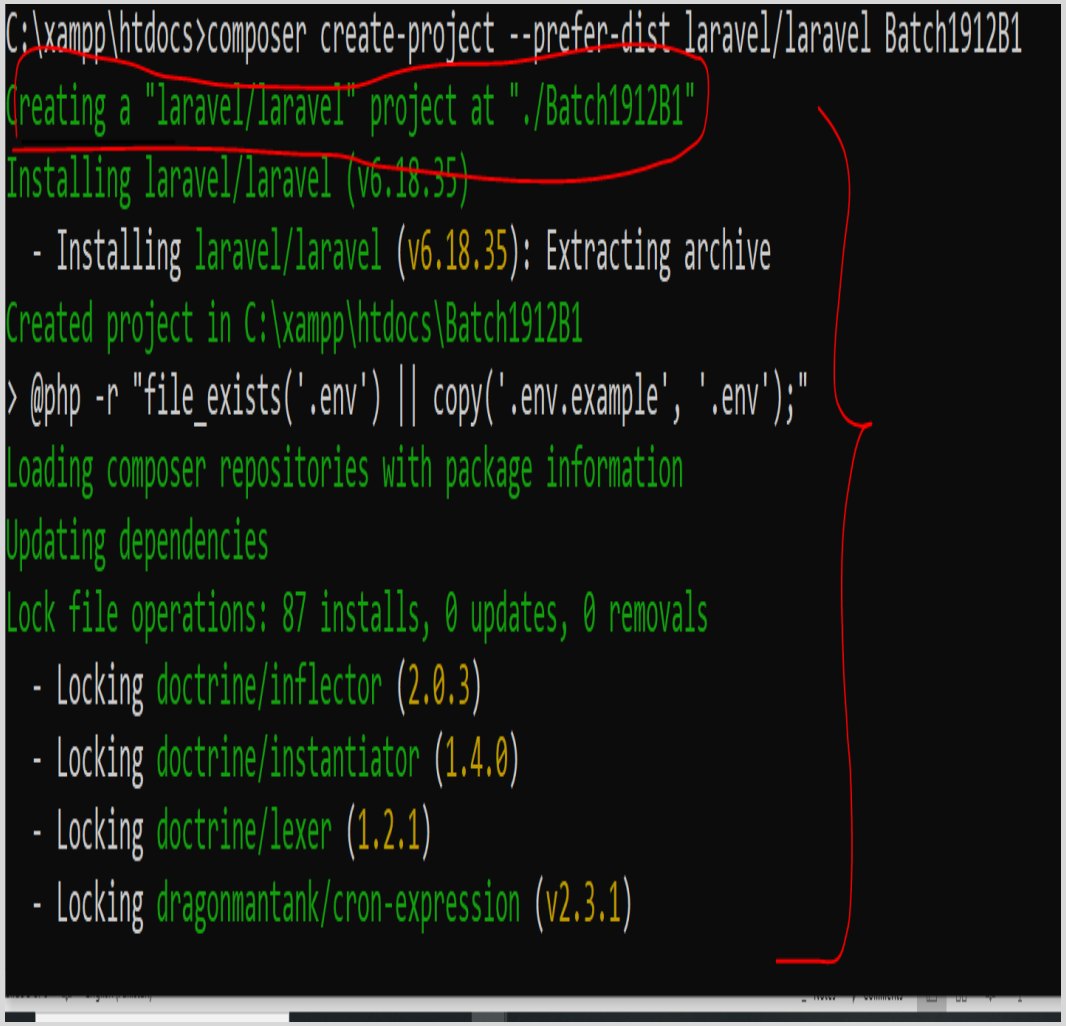
- Search Cmd in search bar and run as administrator

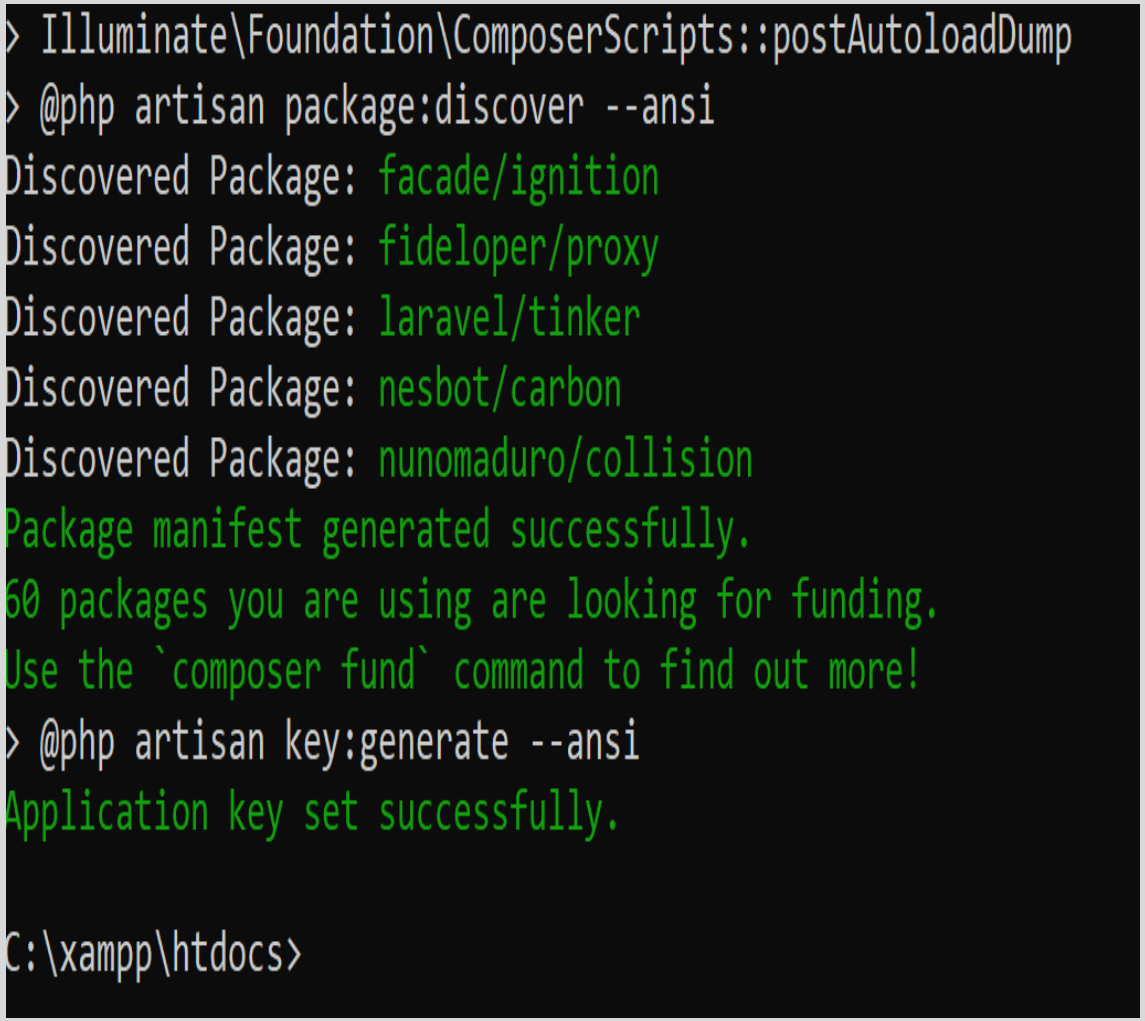


# FOLLOW THESE COMMAND

```
Microsoft Windows [Version 10.0.18363.1556]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd ../

C:\Windows>cd ../

C:\>cd ./xampp
```

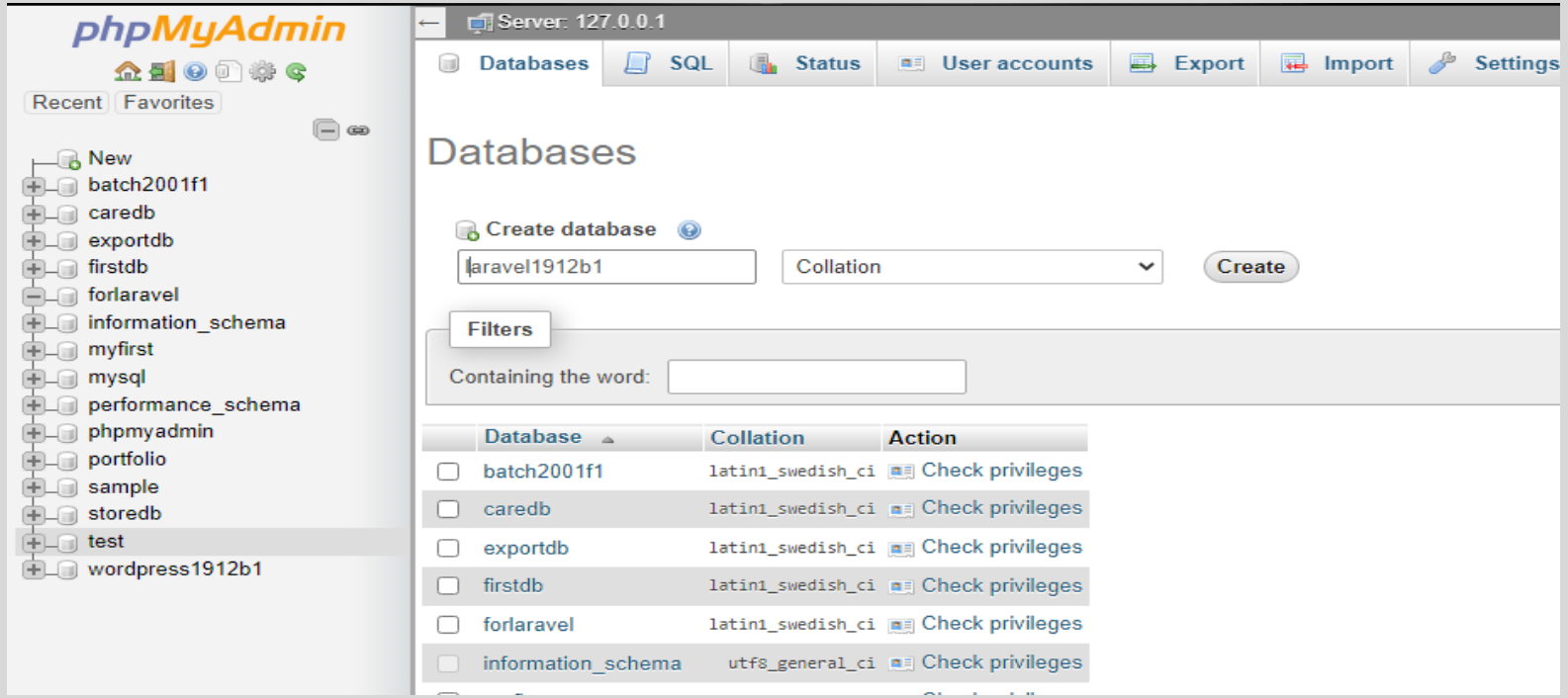
```
C:\>cd ./xampp

C:\xampp>cd ./htdocs

C:\xampp\htdocs>composer create-project --prefer-dist laravel/laravel YourProjName
```


C:\xampp\htdocs>composer create-project --prefer-dist laravel/laravel Batch1912B1
Creating a "laravel/laravel" project at "./Batch1912B1"
Installing laravel/laravel (v6.18.35)
- Installing laravel/laravel (v6.18.35): Extracting archive
Created project in C:\xampp\htdocs\Batch1912B1
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies
Lock file operations: 87 installs, 0 updates, 0 removals
- Locking doctrine/inflector (2.0.3)
- Locking doctrine/instantiator (1.4.0)
- Locking doctrine/lexer (1.2.1)
- Locking dragonmantank/cron-expression (v2.3.1)


> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: facade/ignition
Discovered Package: fideloper/proxy
Discovered Package: laravel/tinker
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Package manifest generated successfully.
60 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.
C:\xampp\htdocs>

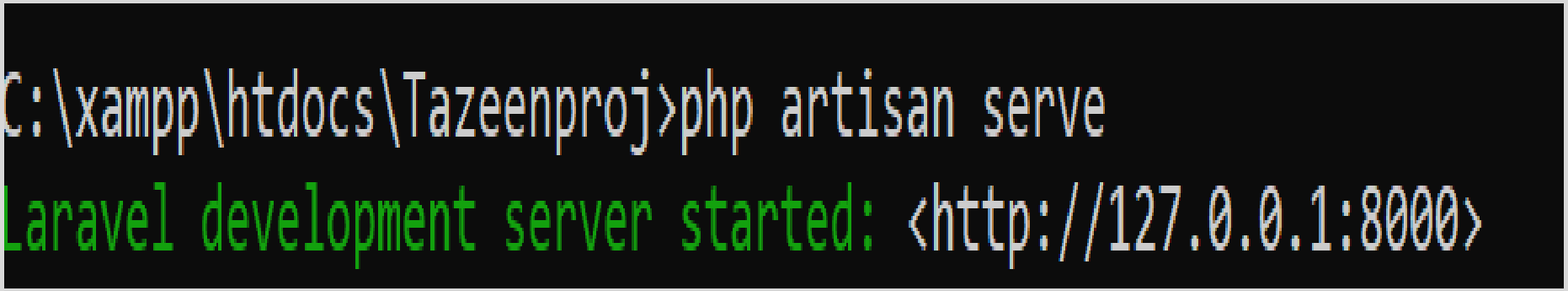
# After PROJECT created

- After Creating a project, you must create a database on phpMyAdmin

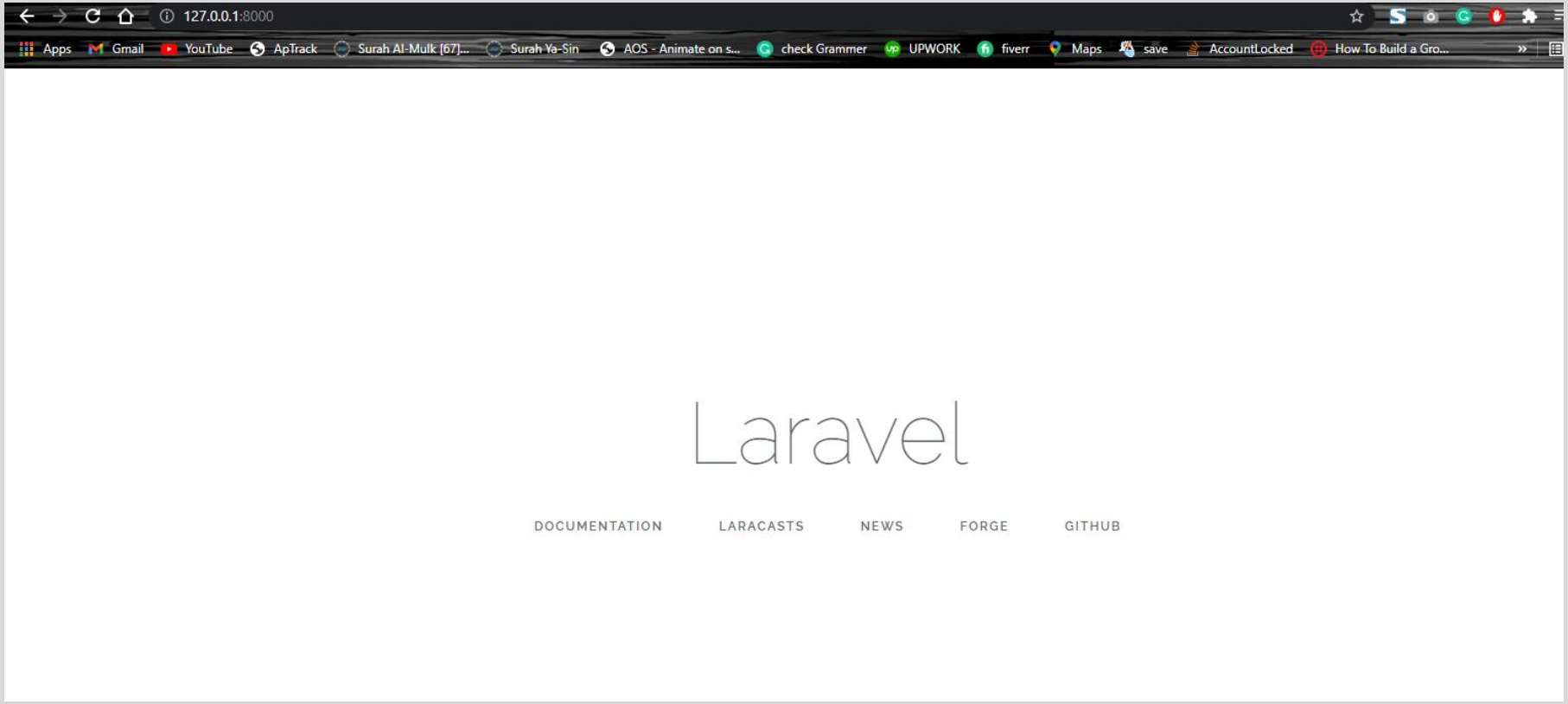


# OPEN LARAVEL PROJECT

- for opening the Laravel demo project write:
  - Php artisan serve like this:
  - So, it will give your project Link



# Project OPEN



# CONTROLLER

- Follow the following command for creating controller
- In the End of **ProjectController** you'll need to add **-resource** after it.
  It will automatically create all type of action method

```
C:\xampp\htdocs\Batch1912B1>php artisan make:controller ProjectController
Controller created successfully.
```

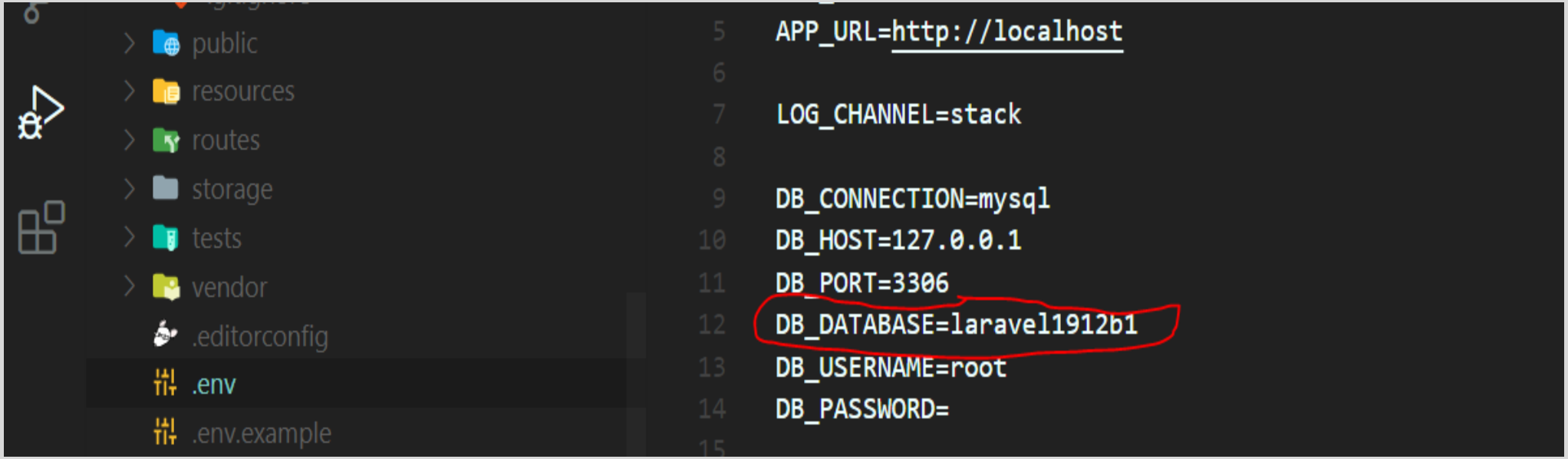
# MODEL

```
C:\xampp\htdocs\Batch1912B1>php artisan make:model Mymodel -m
```

```
C:\xampp\htdocs\Batch1912B1>php artisan make:model Mymodel -m
Model created successfully.
Created Migration: 2021_08_24_114014_create_mymodels_table
```

# DATABASE CONFIGURATION

- Now, you must set the database configuration in **.env** File.
  - In .env file set the Following configs:

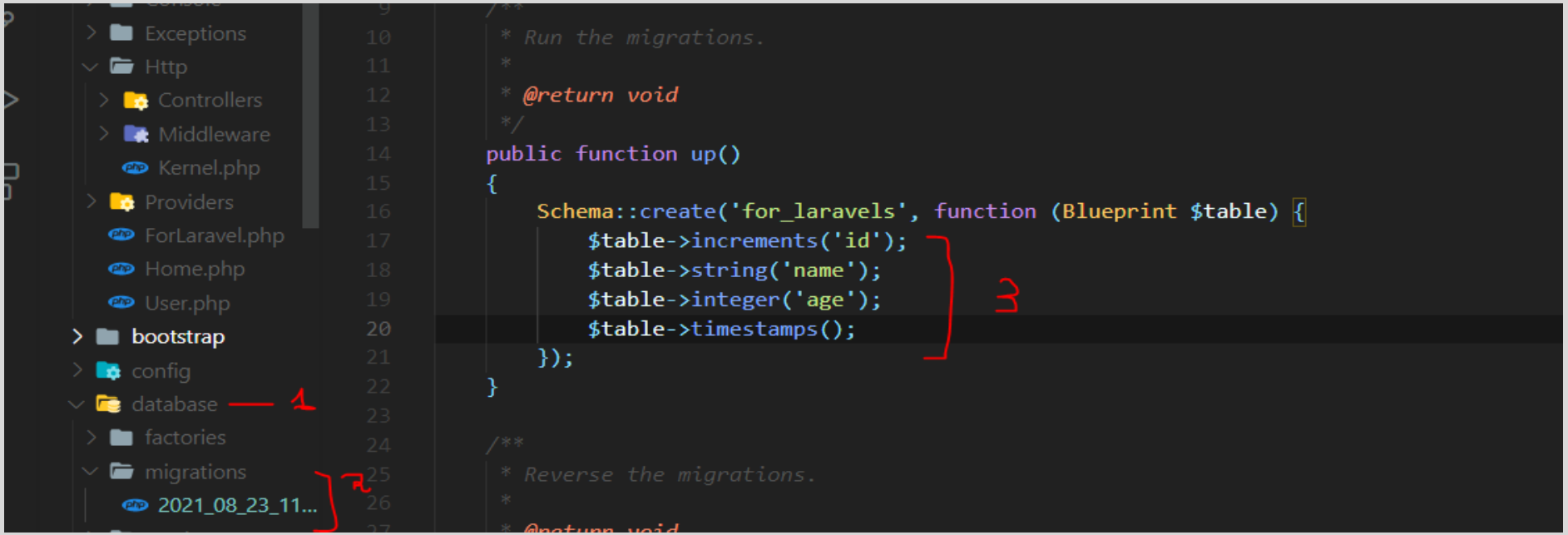


# TABLE

When you create a model **, -m** command automatically created same **model's name** table

Following are the command for **migration table.**

Before Migrate the table first you add the **add table columns** in **Migration Table File**

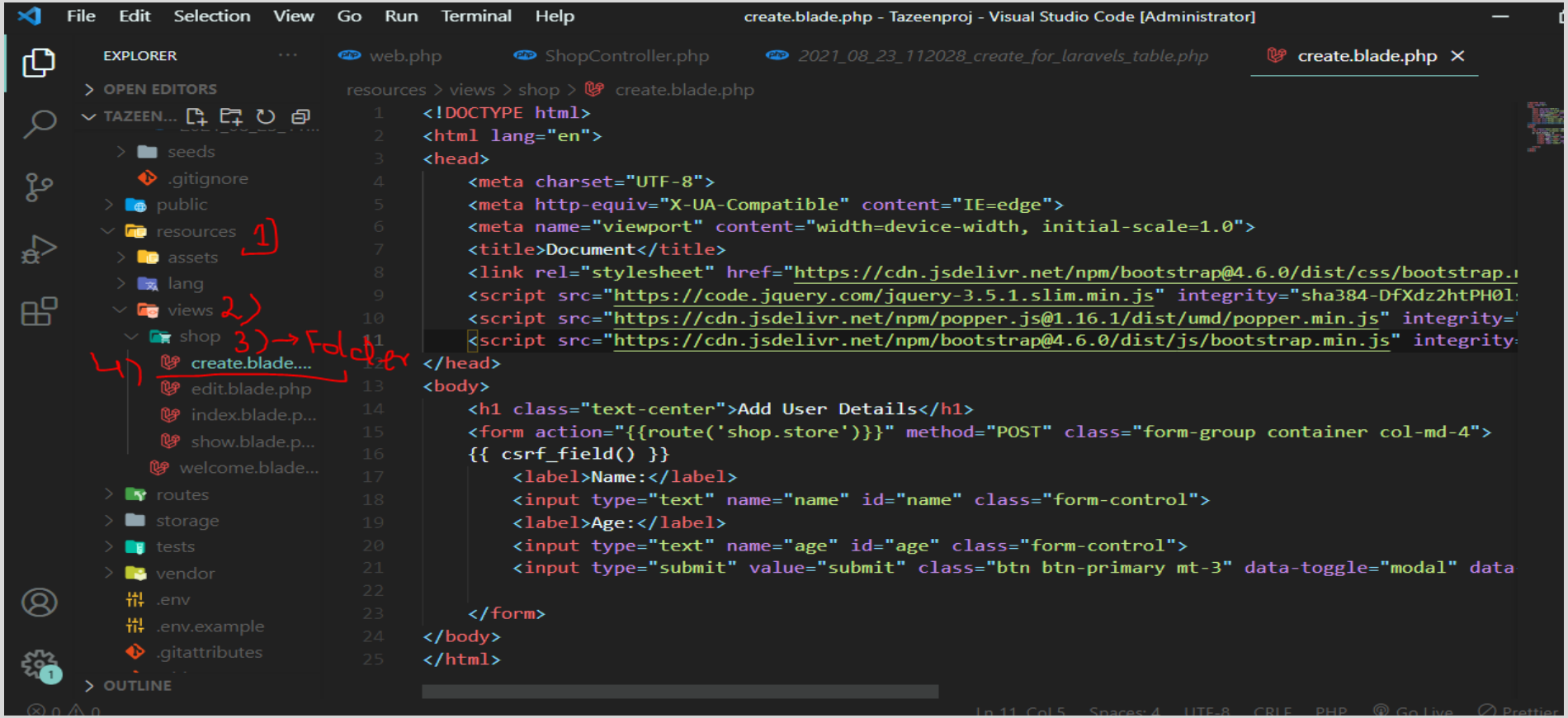


# MIGRATE TABLE

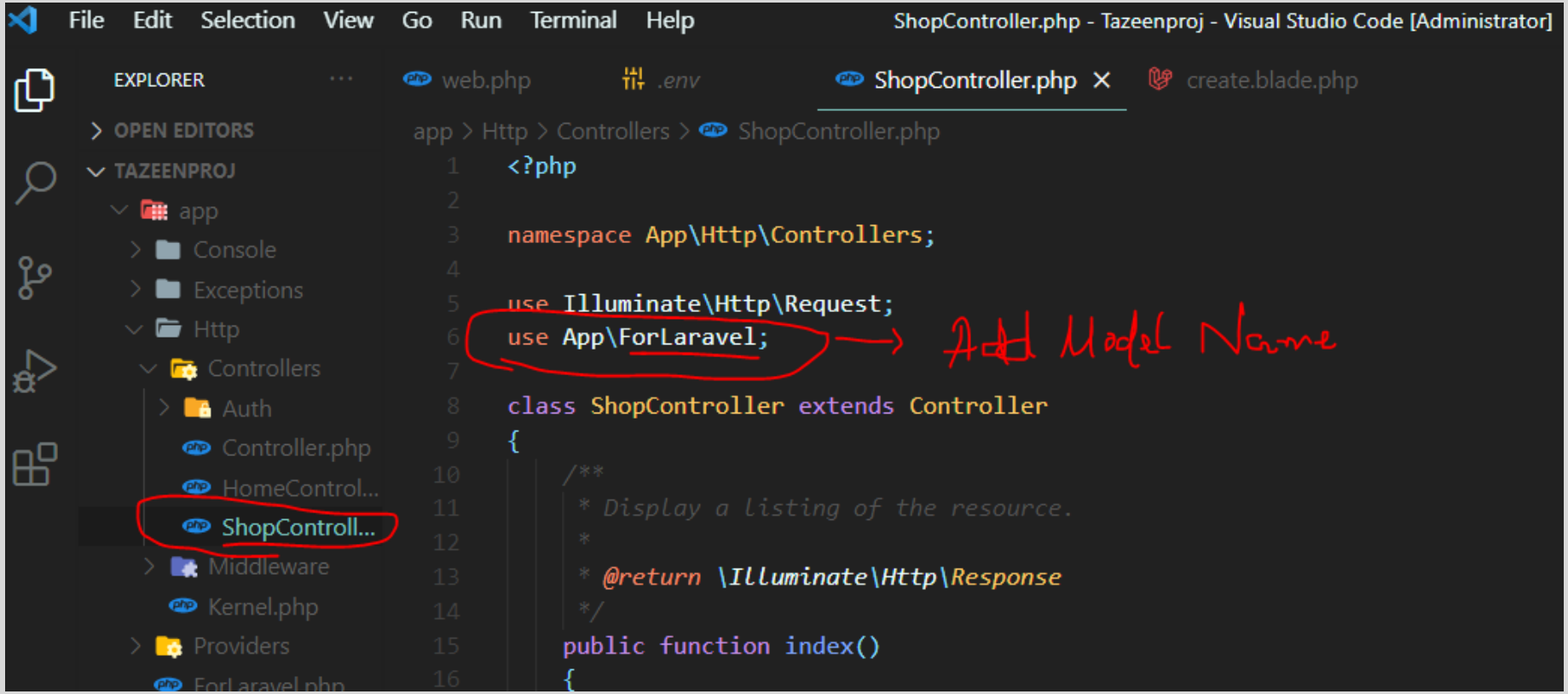
- After adding Fields, You must migrate table to phpMyAdmin.
- Following command migrate the table
  If you'll see any type of error then, delete the users and password migration table from migration folder .
-

```
C:\xampp\htdocs\Batch1912B1>php artisan migrate
Migration table created successfully.
Migrating: 2021_08_24_114014_create_mymodels_table
Migrated:  2021_08_24_114014_create_mymodels_table (0.17 seconds)

C:\xampp\htdocs\Batch1912B1>
```

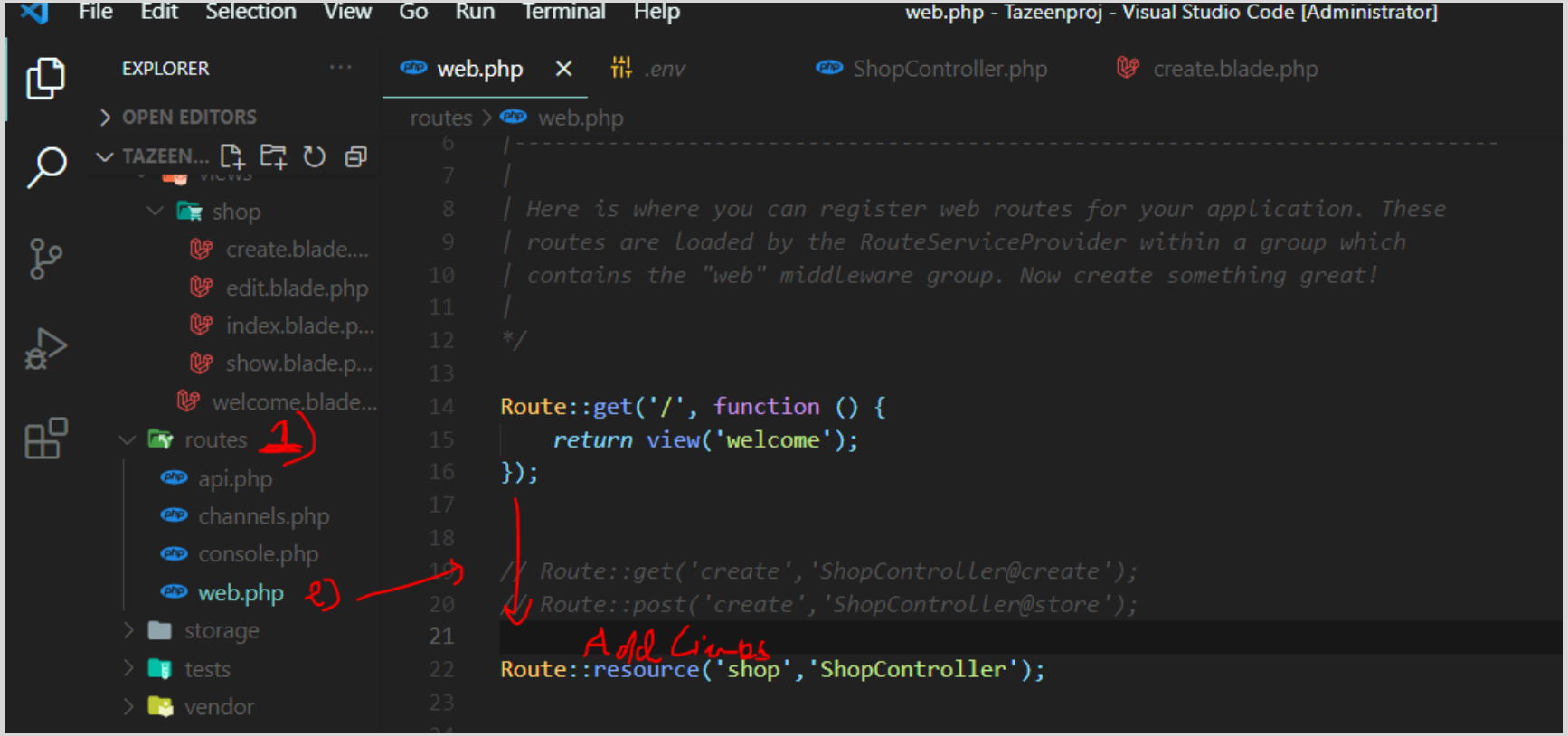
# Create View:


File Edit Selection View Go Run Terminal Help
create.blade.php - Tazeenproj - Visual Studio Code [Administrator]
EXPLORER
web.php
ShopController.php
2021_08_23_112028_create_for_laravels_table.php
create.blade.php
resources > views > shop > create.blade.php
OPEN EDITORS
TAZEEN...
seeds
.gitignore
public
resources
assets
lang
views
shop
create.blade....
edit.blade.php
index.blade.p...
show.blade.p...
welcome.blade...
routes
storage
tests
vendor
.env
.env.example
.gitattributes
OUTLINE
1)
2)
3) → Folder
4)
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/css/bootstrap.
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0l
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity=
<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/js/bootstrap.min.js" integrity
</head>
<body>
<h1 class="text-center">Add User Details</h1>
<form action="{{route('shop.store')}}" method="POST" class="form-group container col-md-4">
{{ csrf_field() }}
<label>Name:</label>
<input type="text" name="name" id="name" class="form-control">
<label>Age:</label>
<input type="text" name="age" id="age" class="form-control">
<input type="submit" value="submit" class="btn btn-primary mt-3" data-toggle="modal" data
</form>
</body>
</html>
Ln 11, Col 5 Spaces: 4 UTF-8 CRLF PHP Go Live Prettier

# SET up the Controller Actions METHOD:



# Add ACTION METHOD CREATE AND STORE

```php
public function create()
{
    return view ('shop.create');
}

/**
 * Store a newly created resource in storage.
 *
 * @param   \Illuminate\Http\Request   $request
 * @return  \Illuminate\Http\Response
 */
public function store(Request $request)
{
    $obj = new ForLaravel;
    $obj->name = $request->name;
    $obj->age = $request->age;
    $obj->save();
    return view('shop.create');
```

# SET UP the ROUTE CONFIGUARTION