

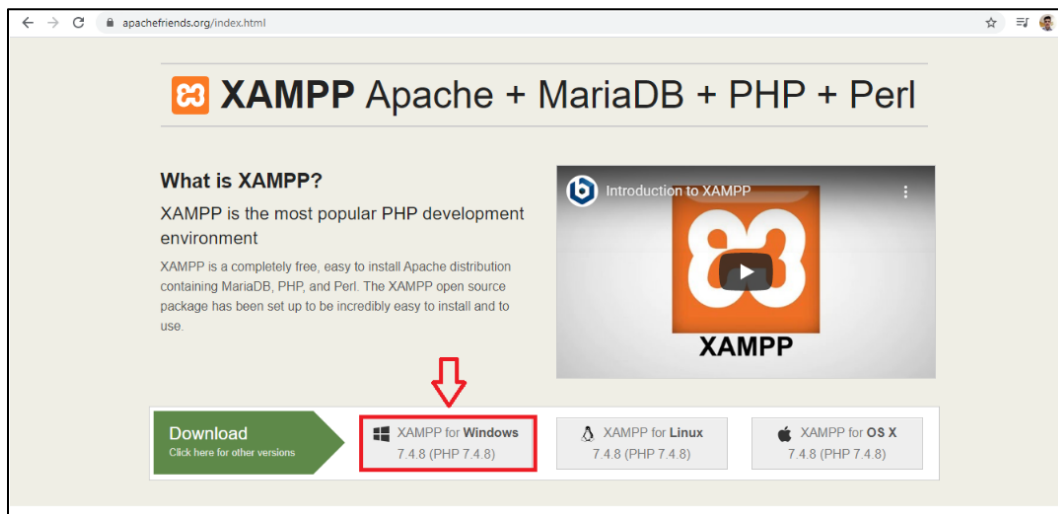
Installations (Laravel & Visual Studio Code)

Part 1: Laravel Installation

Step 1: Install XAMPP on your computer using the following URL

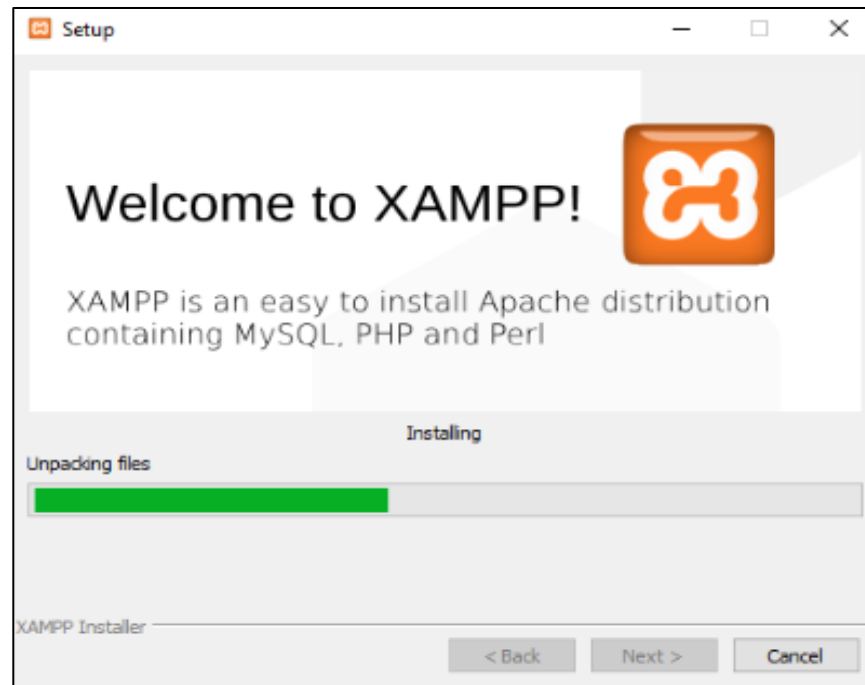
Latest Laravel version depends on the PHP version which is automatically installed during XAMPP installation.

URL: <https://www.apachefriends.org>

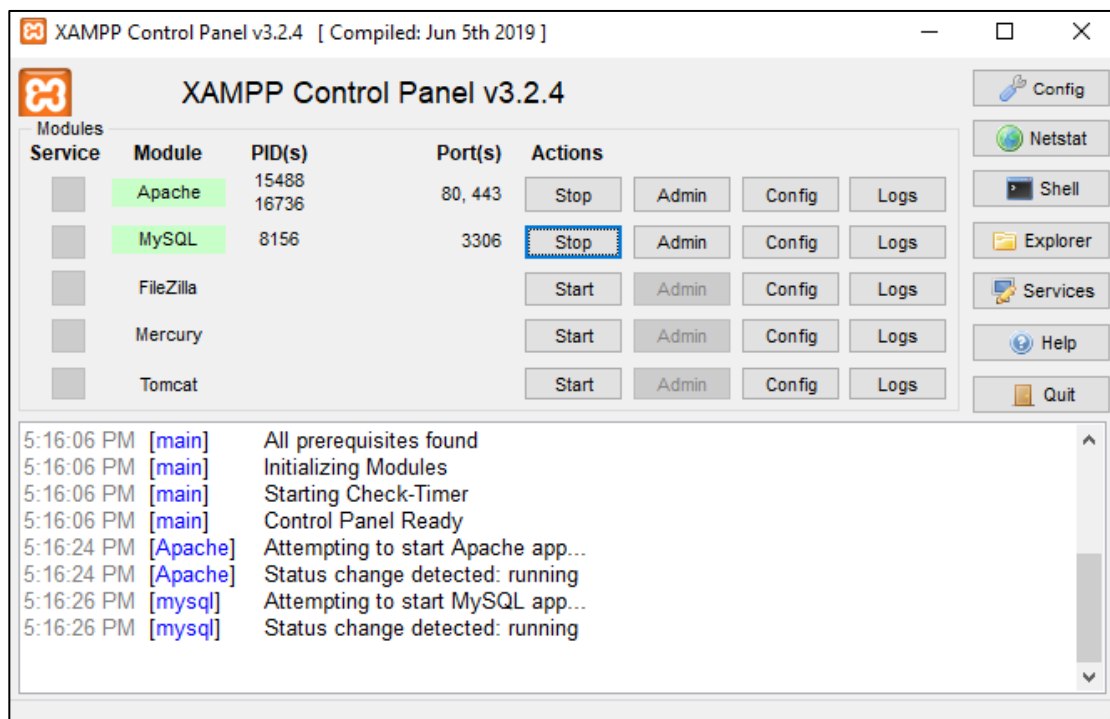


- To download, click **XAMPP for Windows**
- After downloading XAMPP, open the setup, press **Next** button to start installation





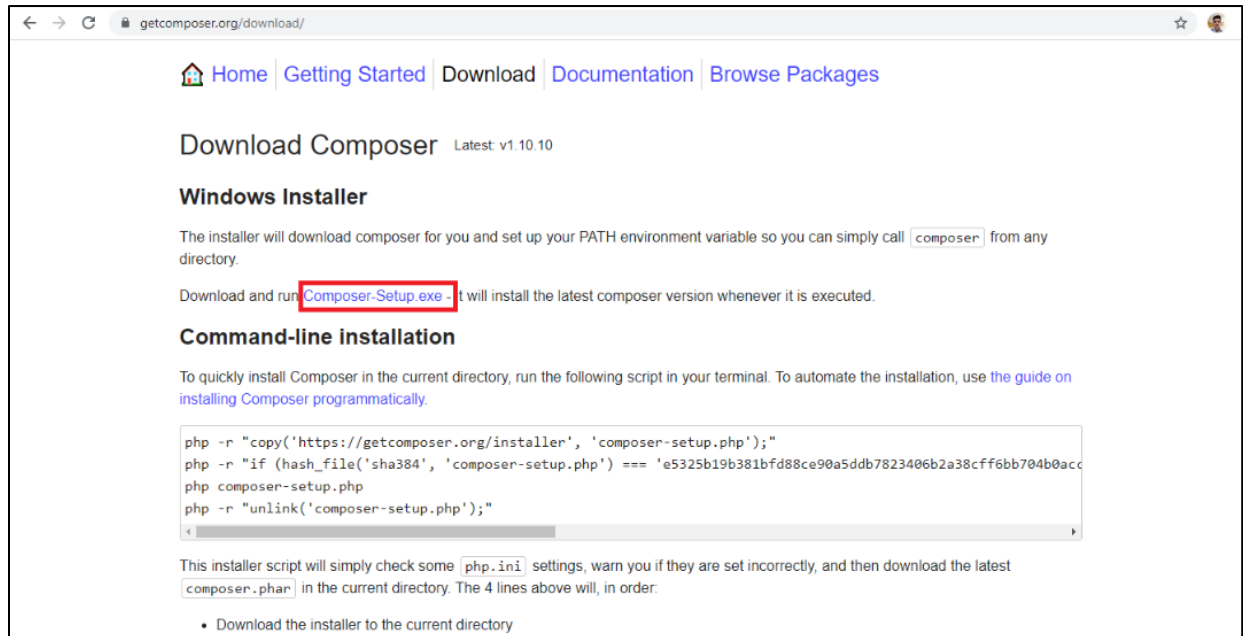
- After successful installation, XAMPP Control Panel would look like below.



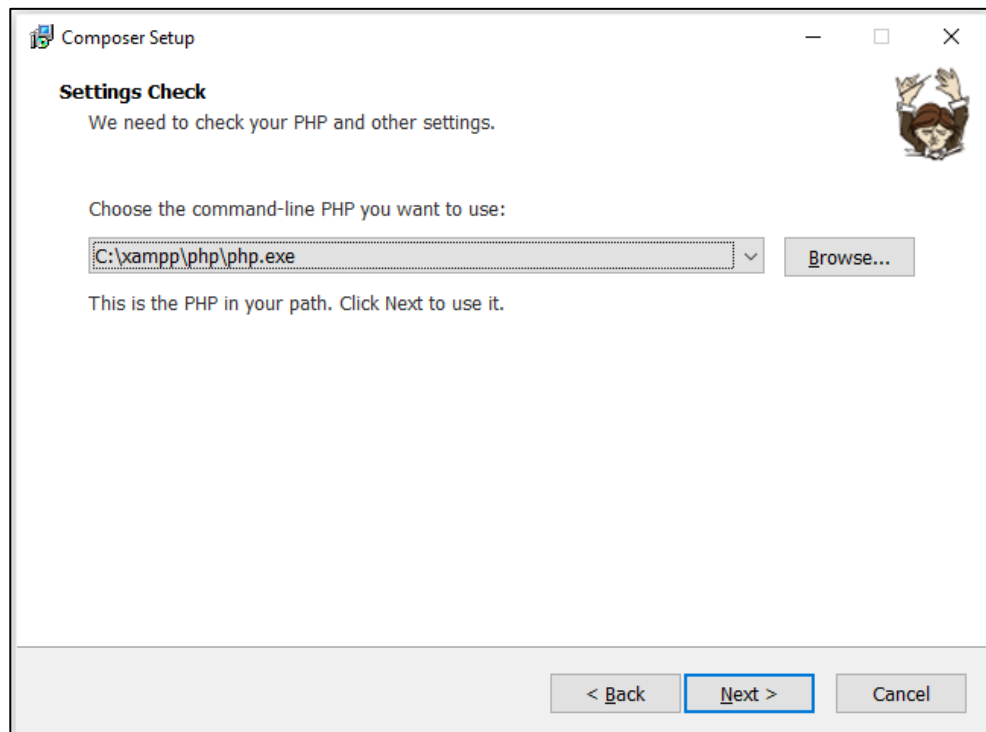
Step 2: Install composer program using the following URL

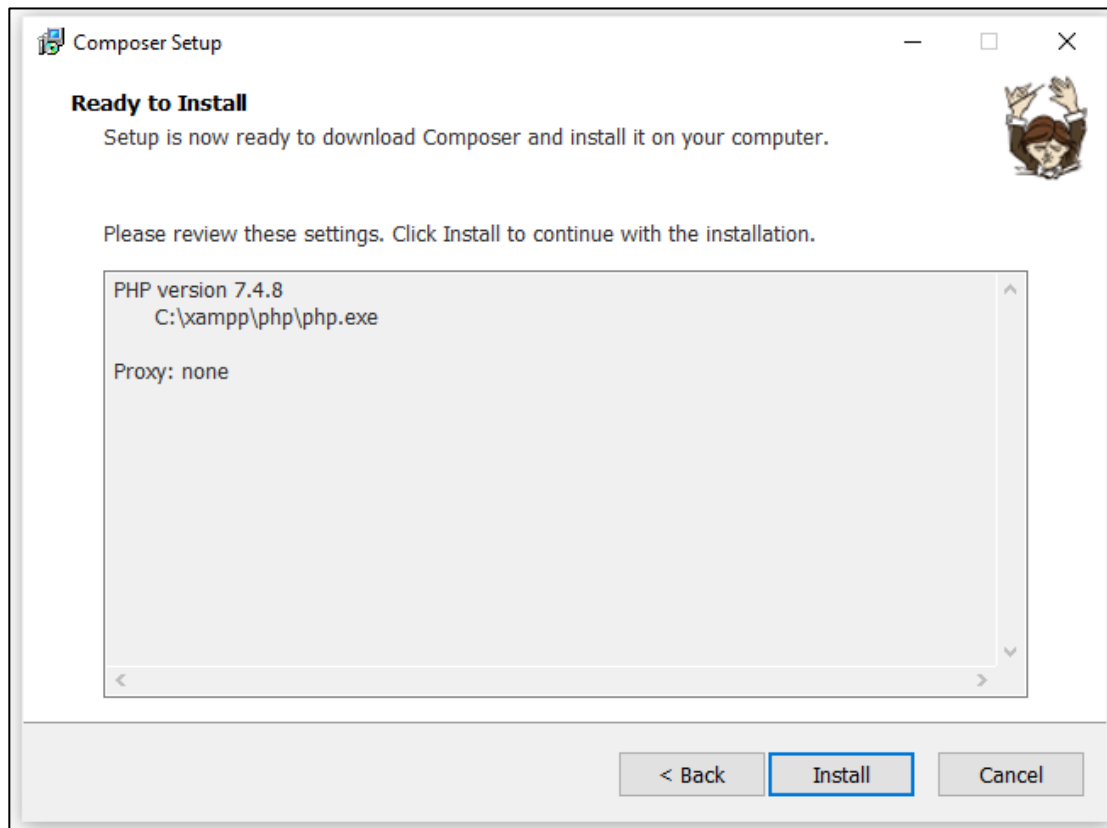
Laravel needs composer to manage different extensions

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



- Run composer.exe after download. During installation, if you are prompted for php.exe path, its address in your system is **xampp/php** by default.





- Once composer has been installed, test it by typing the command **composer** in the command prompt (cmd). It will display composer version which is **version 2.0.7** in my case.

```
Command Prompt
C:\xampp\htdocs>composer

Composer version 2.0.7 2020-11-13 17:31:06

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
  --profile                  Display timing and memory usage information
  --no-plugins               Whether to disable plugins.
  -d, --working-dir=WORKING-DIR If specified, use the given directory as working directory.
  --no-cache                 Prevent use of the cache
  -v|vv|vvv, --verbose      Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug

Available commands:
  about      Shows the short information about Composer.
  archive    Creates an archive of this composer package.
```

Step 3: Install Laravel (locally)


In global installation, Laravel is only installed once and then it is used for each new Project. In local installation, Laravel is installed for each new Project. Use the following URL to understand local and global installation of Laravel.

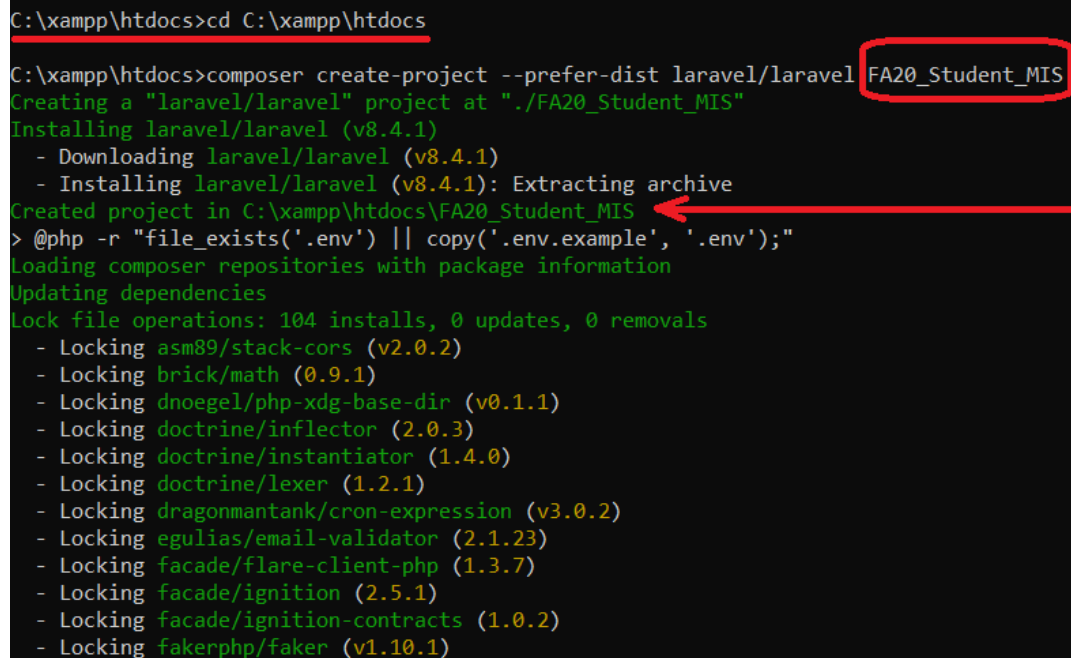
URL: <https://laravel.com/docs/7.x/installation>

We shall use local installation which means Project-wise Laravel installation.

- Type following commands in the Command Prompt to create a folder CRUD and install Laravel in this folder. The name of our Laravel Project is CRUD.

1. `cd c:\xampp\htdocs`
2. `composer create-project --prefer-dist laravel/laravel FA20_Student_MIS`

 Select Command Prompt



```
C:\xampp\htdocs>cd C:\xampp\htdocs
C:\xampp\htdocs>composer create-project --prefer-dist laravel/laravel FA20_Student_MIS
Creating a "laravel/laravel" project at "./FA20_Student_MIS"
Installing laravel/laravel (v8.4.1)
 - Downloading laravel/laravel (v8.4.1)
 - Installing laravel/laravel (v8.4.1): Extracting archive
Created project in C:\xampp\htdocs\FA20_Student_MIS
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies
Lock file operations: 104 installs, 0 updates, 0 removals
 - Locking asm89/stack-cors (v2.0.2)
 - Locking brick/math (0.9.1)
 - Locking dnoegel/php-xdg-base-dir (v0.1.1)
 - Locking doctrine/inflector (2.0.3)
 - Locking doctrine/instantiator (1.4.0)
 - Locking doctrine/lexer (1.2.1)
 - Locking dragonmantank/cron-expression (v3.0.2)
 - Locking egulias/email-validator (2.1.23)
 - Locking facade/flare-client-php (1.3.7)
 - Locking facade/ignition (2.5.1)
 - Locking facade/ignition-contracts (1.0.2)
 - Locking fakerphp/faker (v1.10.1)
```

- After few minutes, Laravel will be installed successfully as shown below.

```
cmd Select Command Prompt
- Installing phpunit/php-invoker (3.1.1): Extracting archive
- Installing phpunit/php-file-iterator (3.0.5): Extracting archive
- Installing theseer/tokenizer (1.2.0): Extracting archive
- Installing sebastian/lines-of-code (1.0.2): Extracting archive
- Installing sebastian/complexity (2.0.2): Extracting archive
- Installing sebastian/code-unit-reverse-lookup (2.0.3): Extracting archive
- Installing phpunit/php-code-coverage (9.2.3): Extracting archive
- Installing doctrine/instantiator (1.4.0): Extracting archive
- Installing phpspec/prophecy (1.12.1): Extracting archive
- Installing phar-io/version (3.0.2): Extracting archive
- Installing phar-io/manifest (2.0.1): Extracting archive
- Installing myclabs/deep-copy (1.10.2): Extracting archive
- Installing phpunit/phpunit (9.4.3): Extracting archive
79 package suggestions were added by new dependencies, use `composer suggest` to see details.
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: facade/ignition
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.
C:\xampp\htdocs>
```

- Laravel has been successfully installed in the project folder **FA20_Student_MIS**
- After installation, type the following commands in the Command Prompt to go the **FA20_Student_MIS** folder and start there the PHP development server, for development and testing.
 1. `cd FA20_Student_MIS`
 2. `php artisan serve`

```
cmd Command Prompt - php artisan serve
Use the `composer fund` command to find out more!
> @php artisan key:generate --ansi
Application key set successfully.

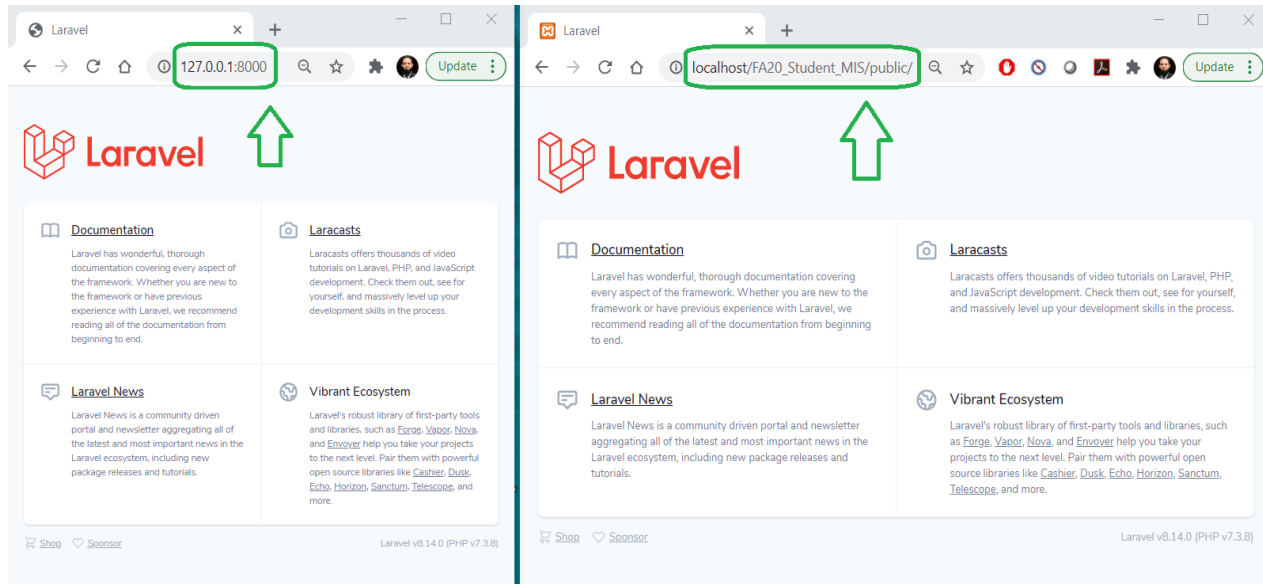
C:\xampp\htdocs>cd FA20_Student_MIS

C:\xampp\htdocs\FA20_Student_MIS>ls
README.md  artisan  composer.json  config  package.json  public  routes  storage  vendor
app        bootstrap  composer.lock  database  phpunit.xml  resources  server.php  tests  webpack.mix.js

C:\xampp\htdocs\FA20_Student_MIS>php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
```

- Now, open the browser and type one of the following in the addresses bar to start the Laravel application

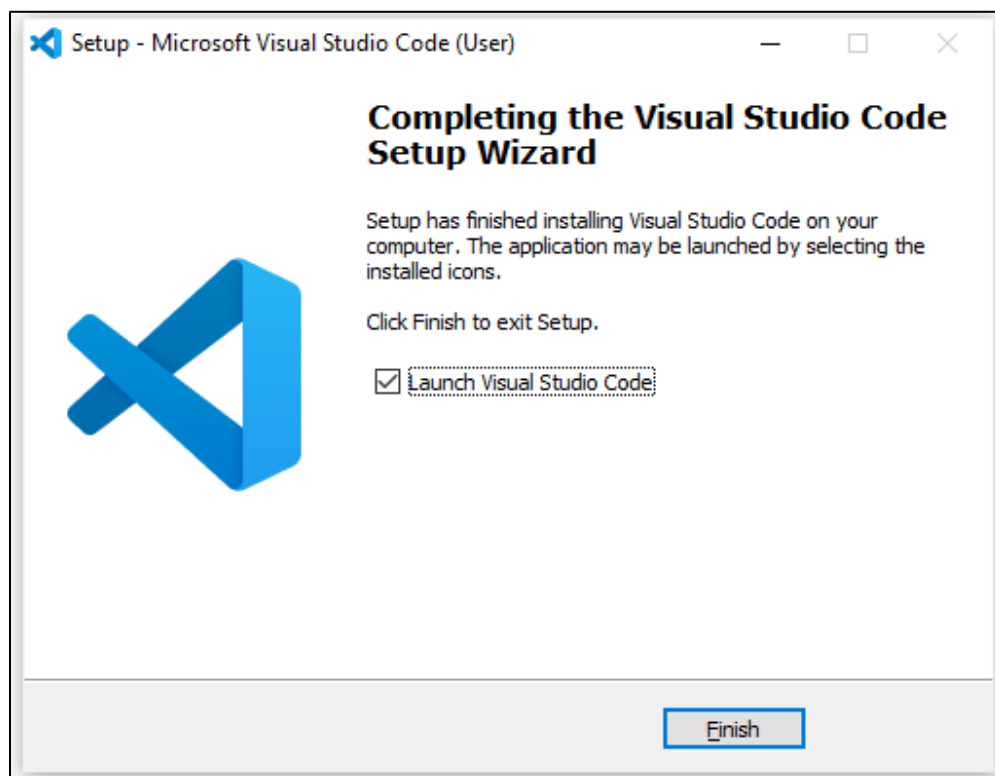
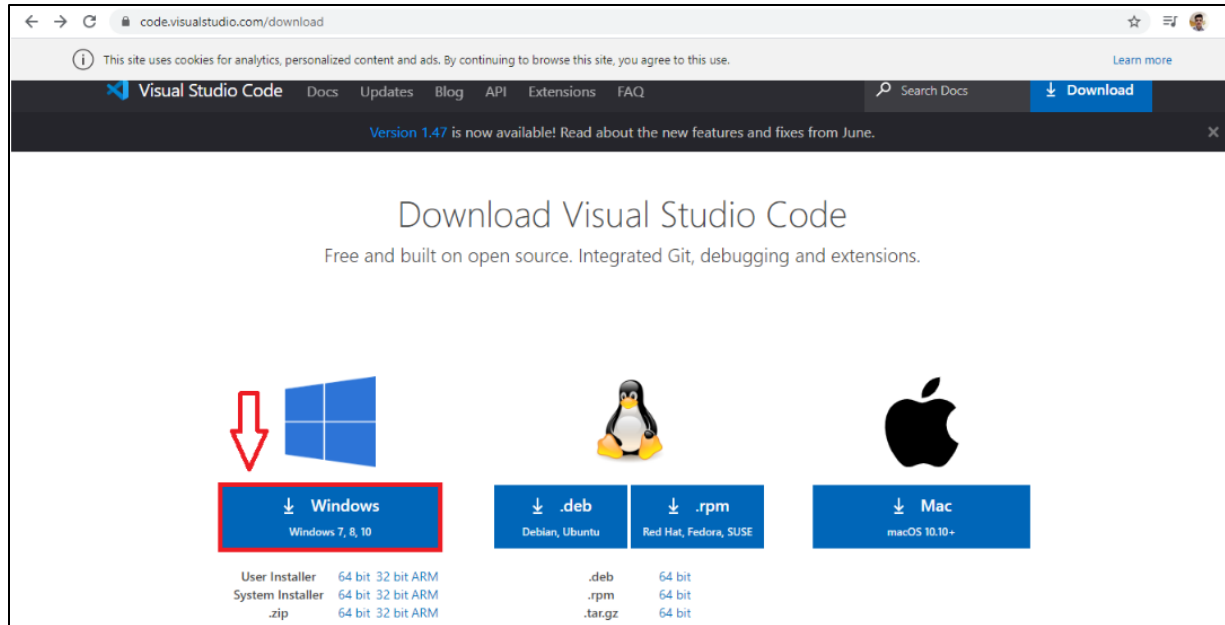
1. `http://127.0.0.1:8000`
2. `http://localhost/FA20_Student_MIS/public/`



Part 2: Microsoft Visual Studio Code (VS Code) Installation

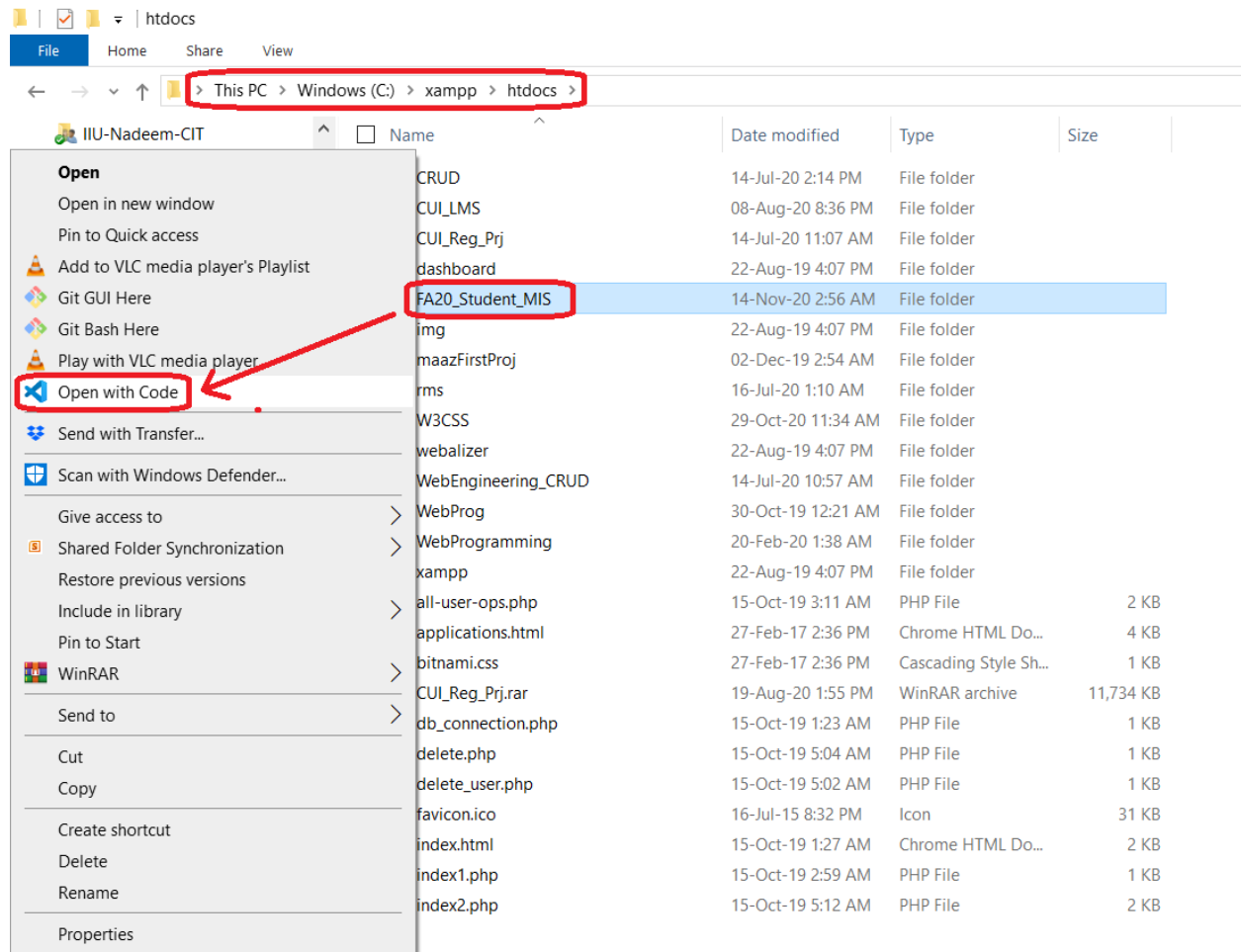
Step 1: Download VS Code from the following URL.

URL: <https://code.visualstudio.com/download>

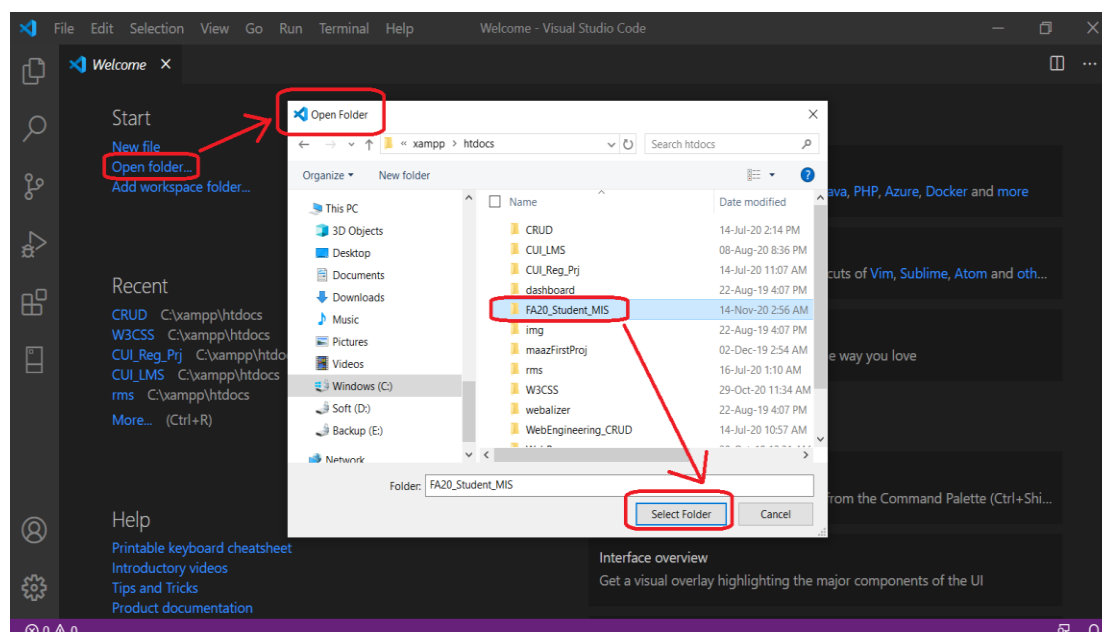


- After Successful installation, Open Laravel project using VS Code using one of the TWO methods.

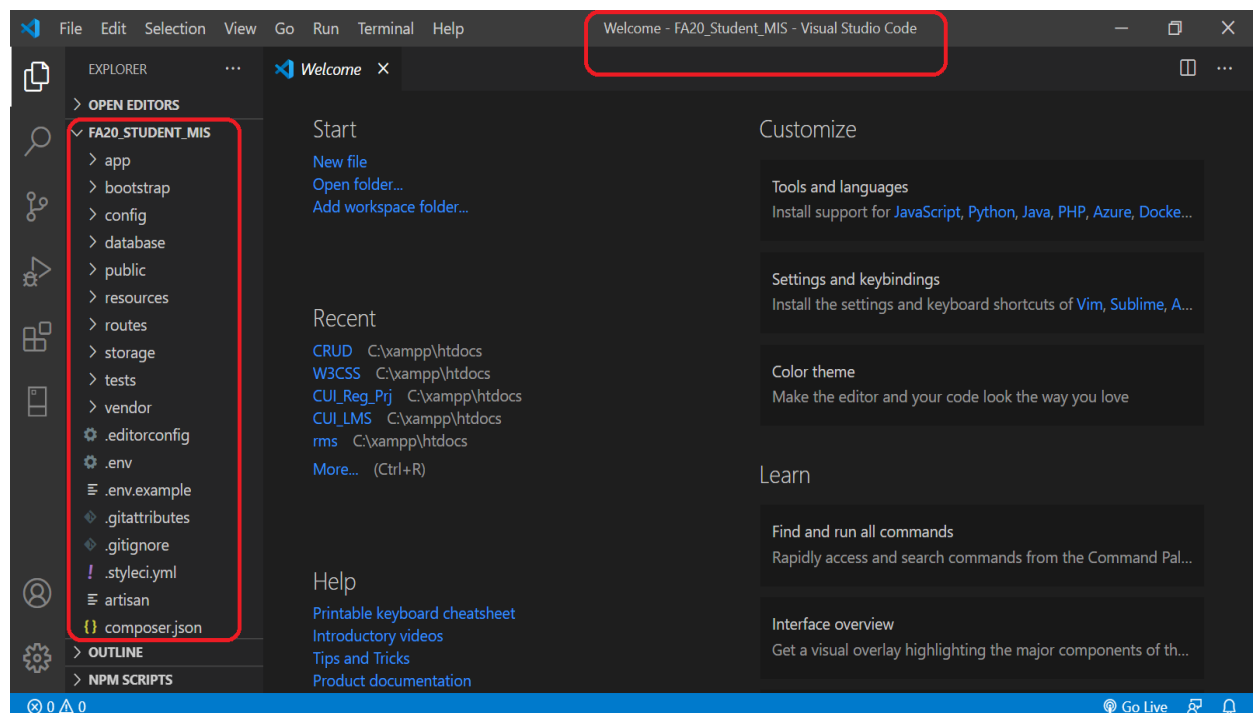
1. Using right-click on the **project folder** in **htdocs**



2. Using Open Project option of VS Code editor

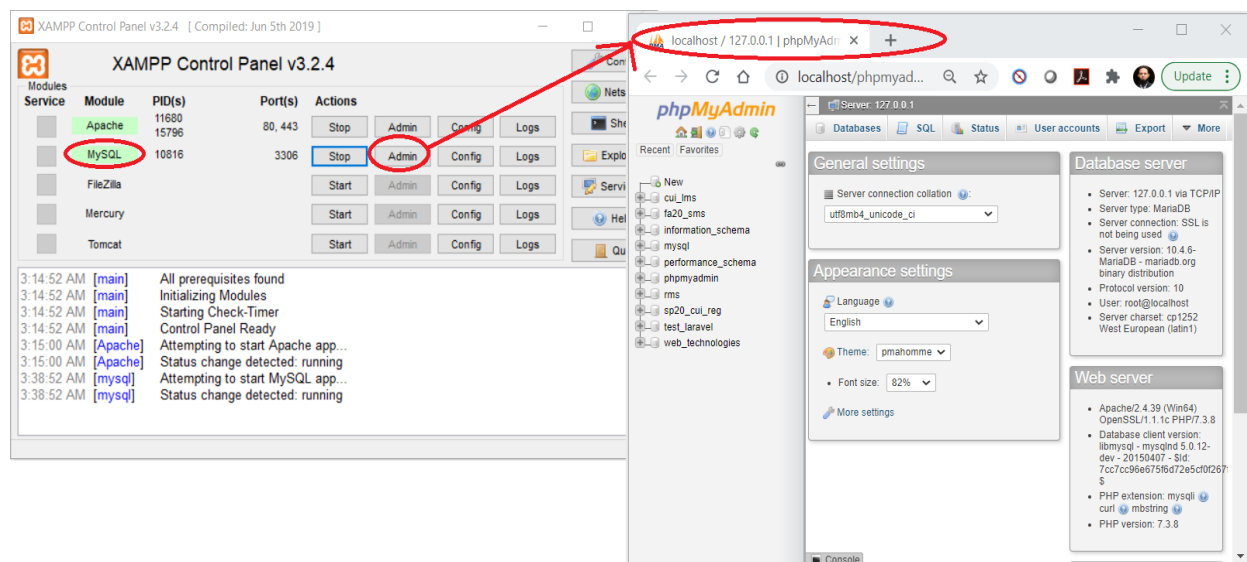


- Laravel project will look in VS Code as shown below

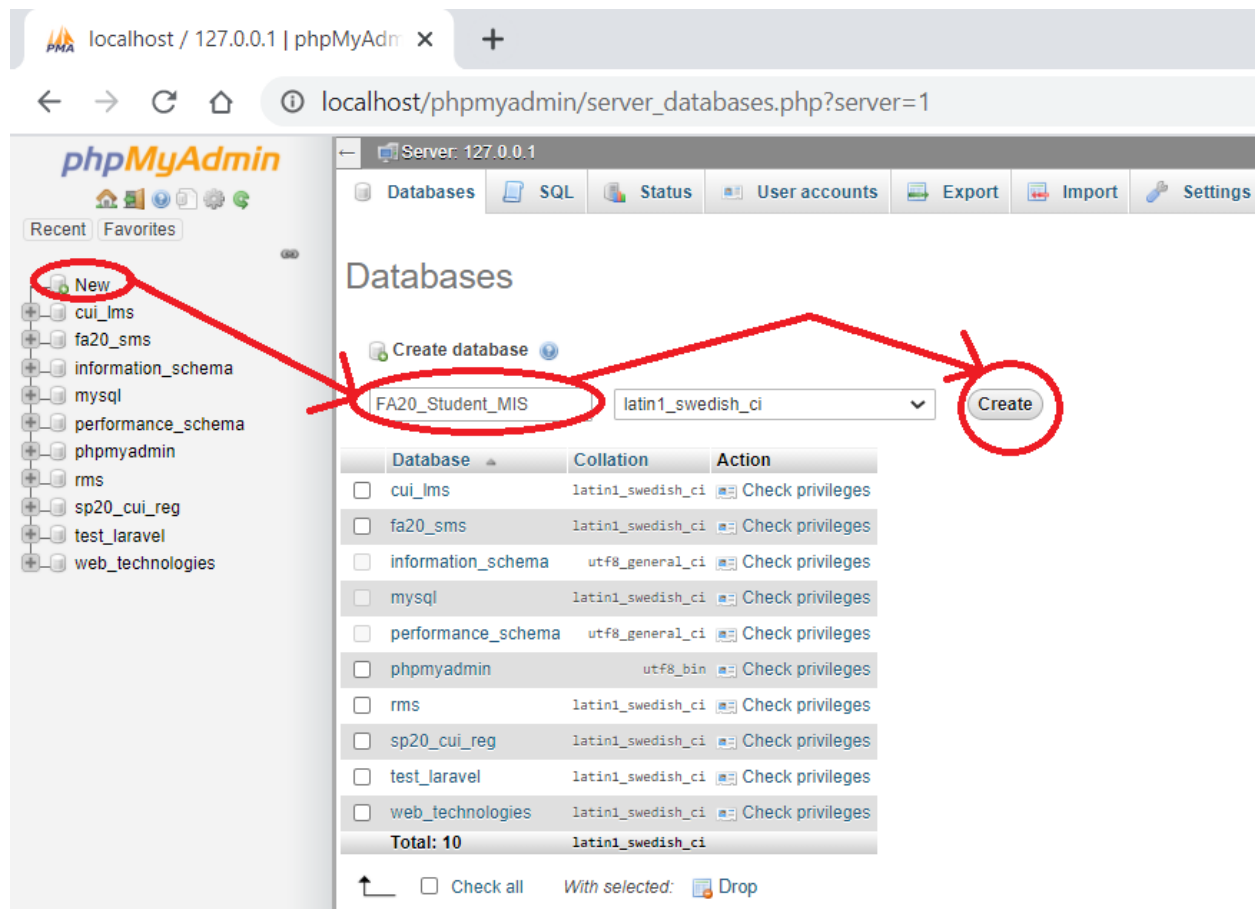


Part 3: Creating a Database in MySQL and linking with Laravel Application

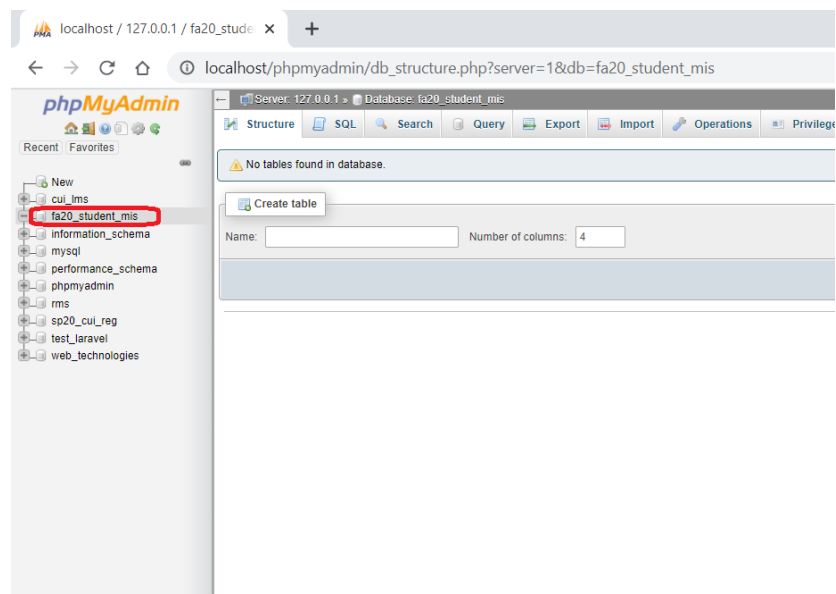
- Open XAMPP MySQL as shown in the image below.



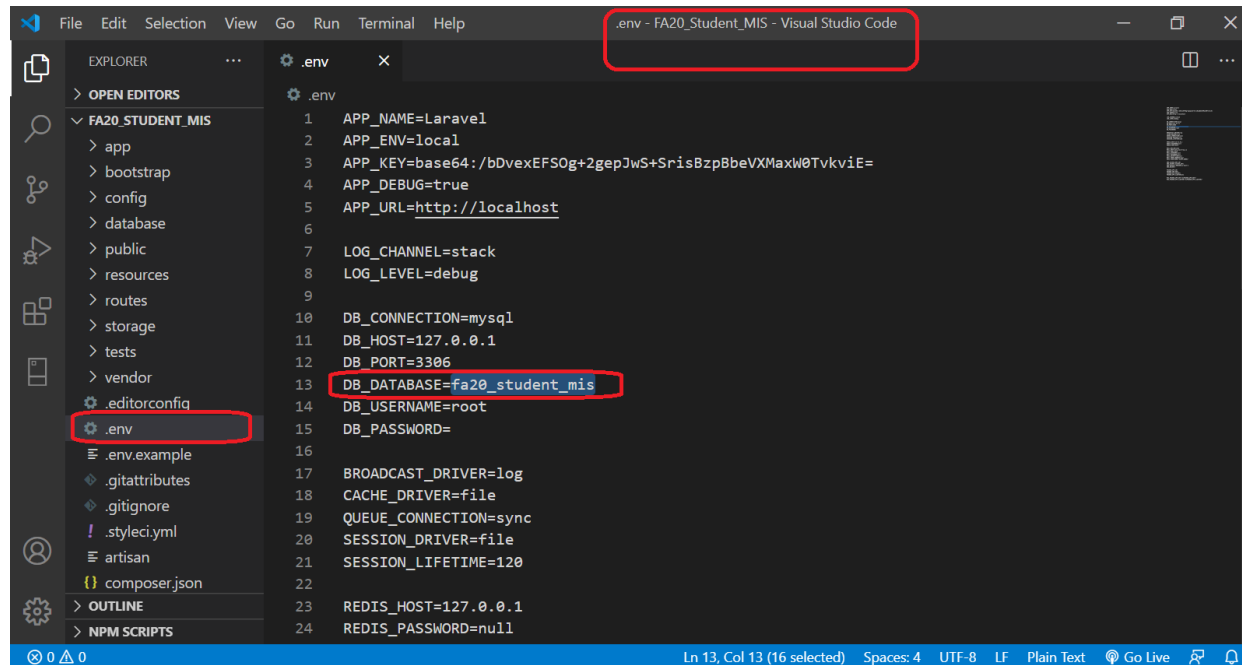
- Create a new database in your XAMPP MySQL



- Once the database is created, we need to tell our Laravel project the details about the database.



- Modify the following property in your **.env** file according to your database name. MY database name is **fa20_student_mis**.



```
.env - FA20_Student_MIS - Visual Studio Code

1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:/bDvexEFS0g+2gepJwS+SrisBzpBbeVXMaxW0TvkviE=
4 APP_DEBUG=true
5 APP_URL=http://localhost
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=fa20_student_mis
14 DB_USERNAME=root
15 DB_PASSWORD=
16
17 BROADCAST_DRIVER=log
18 CACHE_DRIVER=file
19 QUEUE_CONNECTION=sync
20 SESSION_DRIVER=file
21 SESSION_LIFETIME=120
22
23 REDIS_HOST=127.0.0.1
24 REDIS_PASSWORD=null
```
