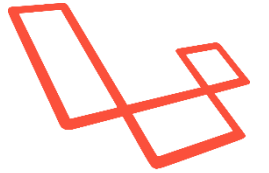


**SESSION
02**

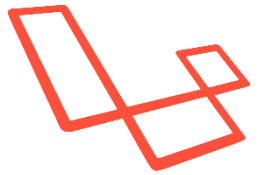
INTRODUCTION TO LARAVEL



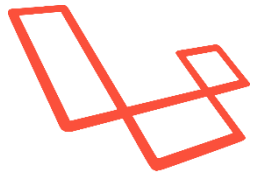
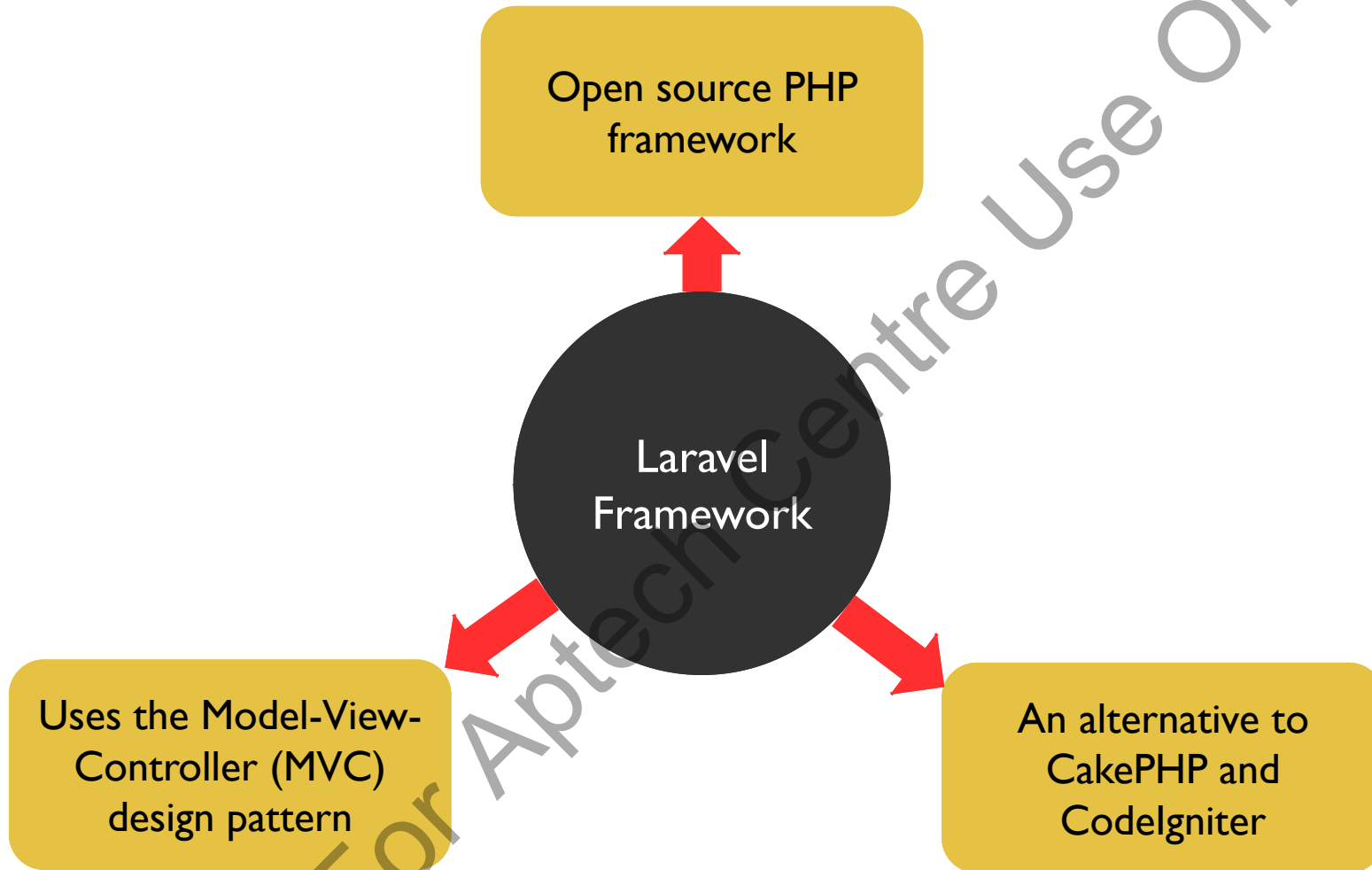
OBJECTIVES

- Explain the details of Laravel framework
- Understand the need of frameworks
- Describe the features of Laravel
- Learn to view the Laravel Directory structure

For Aptech Centre Use Only



LARAVEL FRAMEWORK



NEED FOR FRAMEWORKS (1-2)

Framework is a basic templating structure that is used to develop, test, and deploy applications rapidly and efficiently.

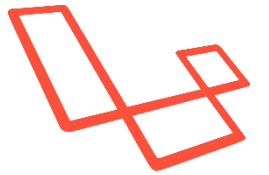
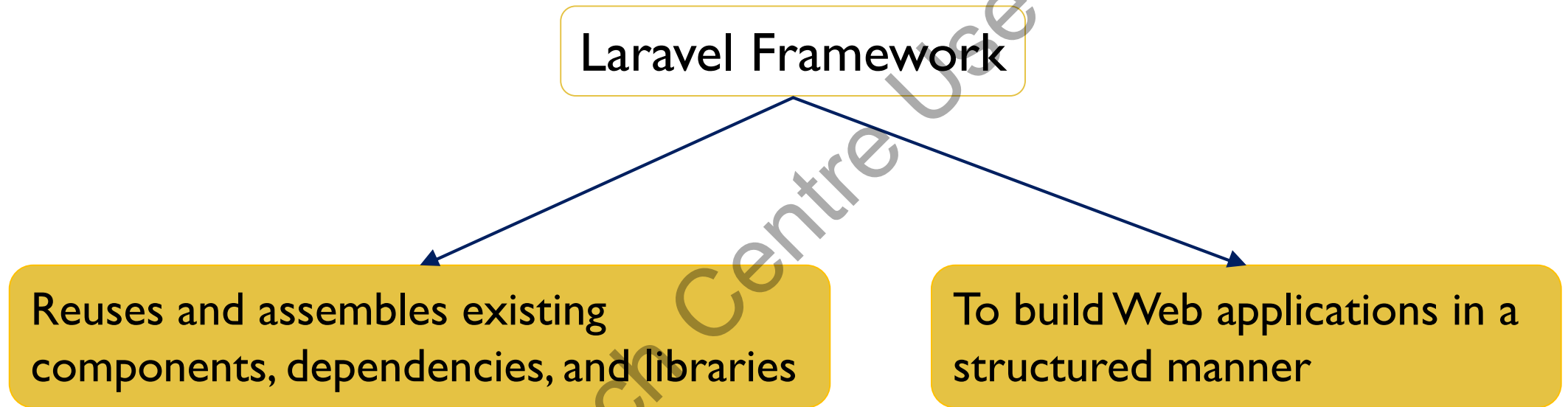
Purpose of Frameworks

Provide abstraction

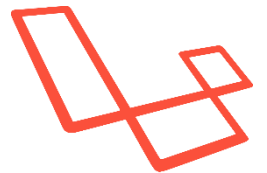
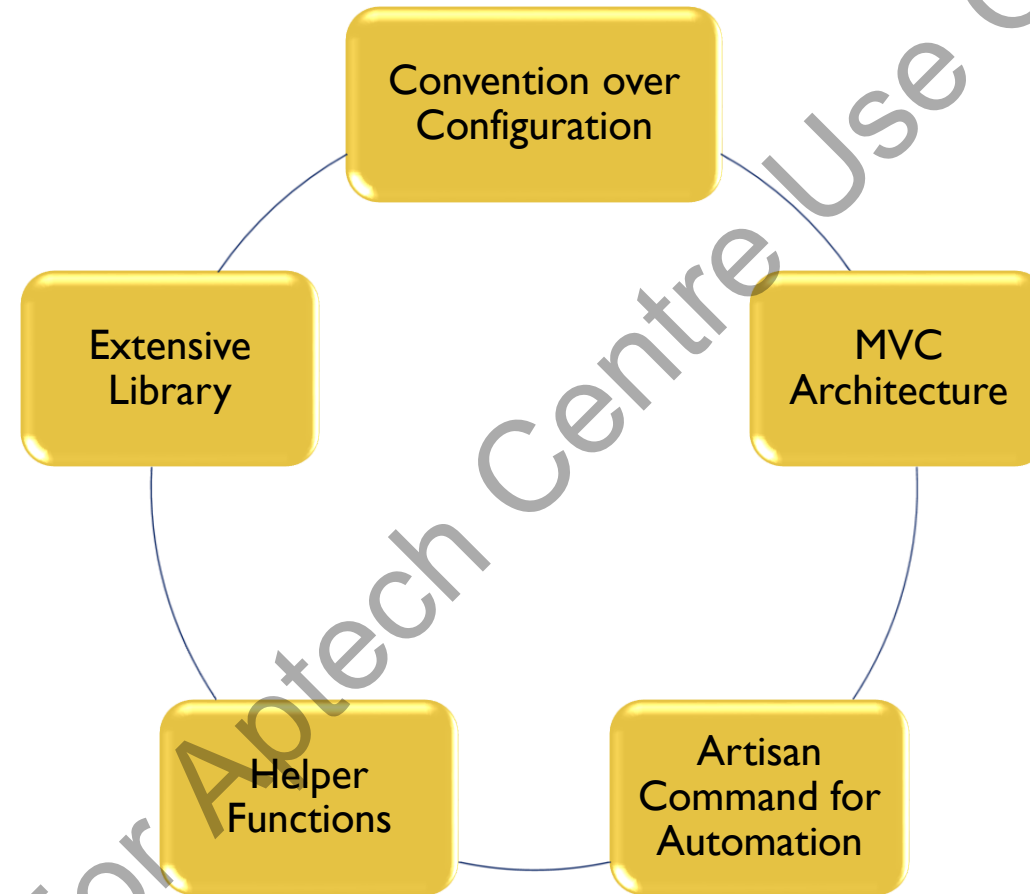
Handles repetitive and menial tasks

Arranges things in a logical structure

NEED FOR FRAMEWORKS (2-2)



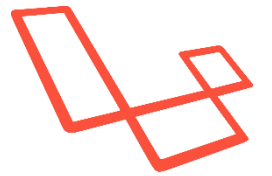
FEATURES OF LARAVEL (1-7)



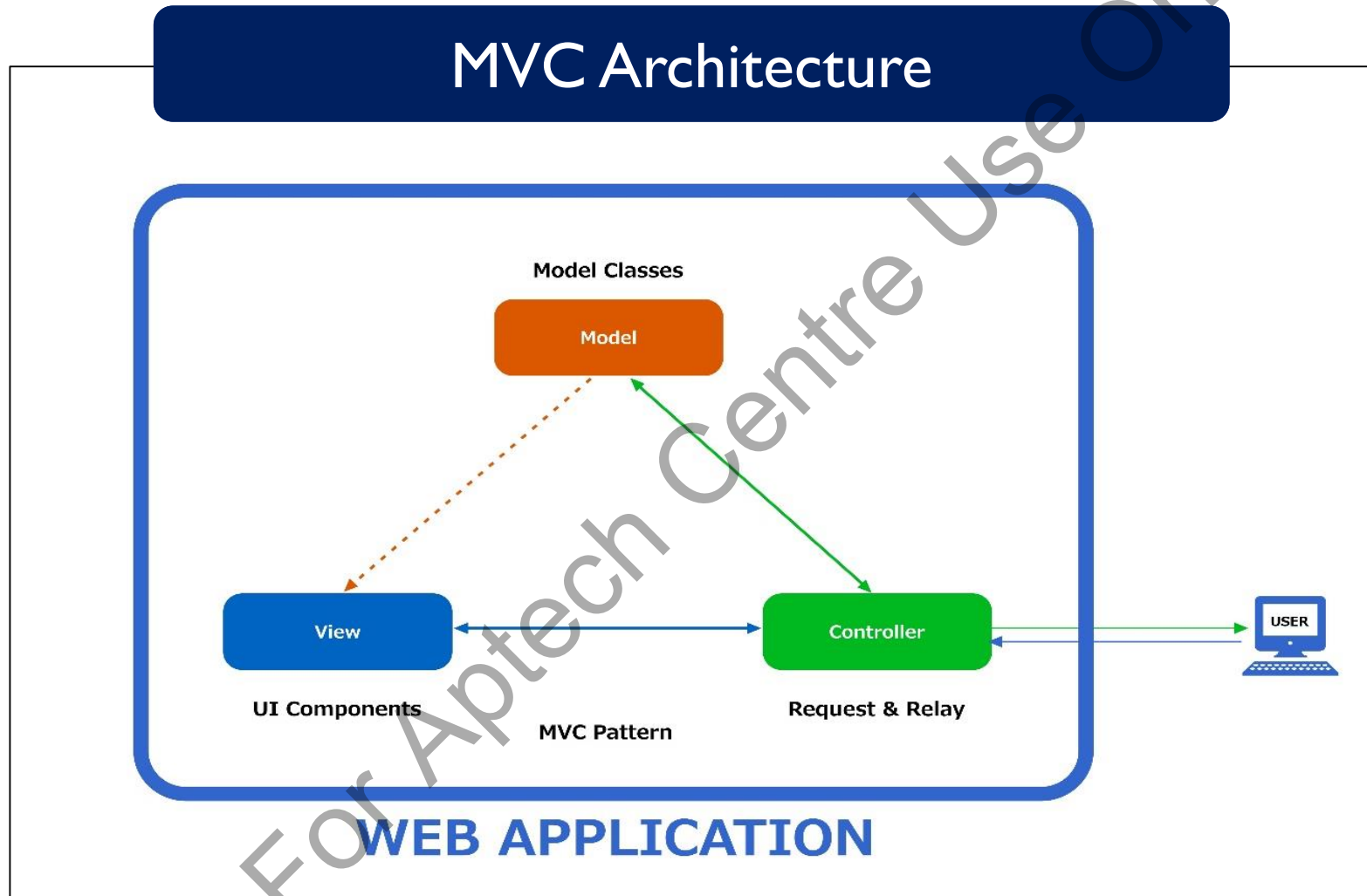
FEATURES OF LARAVEL (2-7)

Convention over Configuration

- Can predict the dependencies
- Automate the configuration related tasks
- Reduces the time and effort required to set up and configure services
- Suitable for Rapid Application Development



FEATURES OF LARAVEL (3-7)



FEATURES OF LARAVEL (4-7)

Artisan Commands

- Automation tool
- Used to create models, views, and controllers
- Used to create dummy server
- Perform database operations

FEATURES OF LARAVEL (5-7)

Code snippet to get a list of commands available for artisan: `php artisan list`

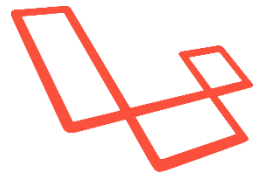
```
vagrant@homestead: ~/laravelProject
vagrant@homestead:~/laravelProject$ php artisan list
Laravel Framework 5.8.15

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
      --env[=ENV]      The environment the command should run under
  -v|vv|vvv, --verbose Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debugging

Available commands:
  clear-compiled  Remove the compiled class file
  down           Put the application into maintenance mode
  dump-server    Start the dump server to collect dump information.
  env           Display the current framework environment
  help          Displays help for a command
  inspire       Display an inspiring quote
  list          Lists commands
  migrate       Run the database migrations
  optimize      Cache the framework bootstrap files
  preset       Swap the front-end scaffolding for the application
  serve        Serve the application on the PHP development server
  tinker        Interact with your application
  up           Bring the application out of maintenance mode

  app
  app:name     Set the application namespace
  auth
  auth:clear-compiled  Flush expired password reset tokens
  cache
  cache:clear    Flush the application cache
  cache:forget   Remove an item from the cache
  cache:table    Create a migration for the cache database table
  config
  config:cache   Create a cache file for faster configuration loading
  config:clear   Remove the configuration cache file
  db
  db:seed       Seed the database with records
  event
  event:cache   Discover and cache the application's events and listeners
  event:clear   Clear all cached events and listeners
  event:generate Generate the missing events and listeners based on registration
  event:list    List the application's events and listeners
```



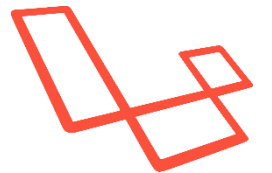
FEATURES OF LARAVEL (6-7)

Helper Function

- Global PHP functions
- Reduces time and effort

Helper functions are classified as follows:

- Arrays and Objects
- Paths
- Strings
- URLs
- Miscellaneous



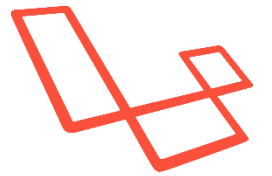
FEATURES OF LARAVEL (7-7)

Extensive Library

- Includes a lot of popular and commonly used libraries
- Reduces time and effort

Code snippet that allows inclusion of external libraries using the composer package manager:

```
Composer require <library-name>
```



LARAVEL DIRECTORY STRUCTURE (1-2)

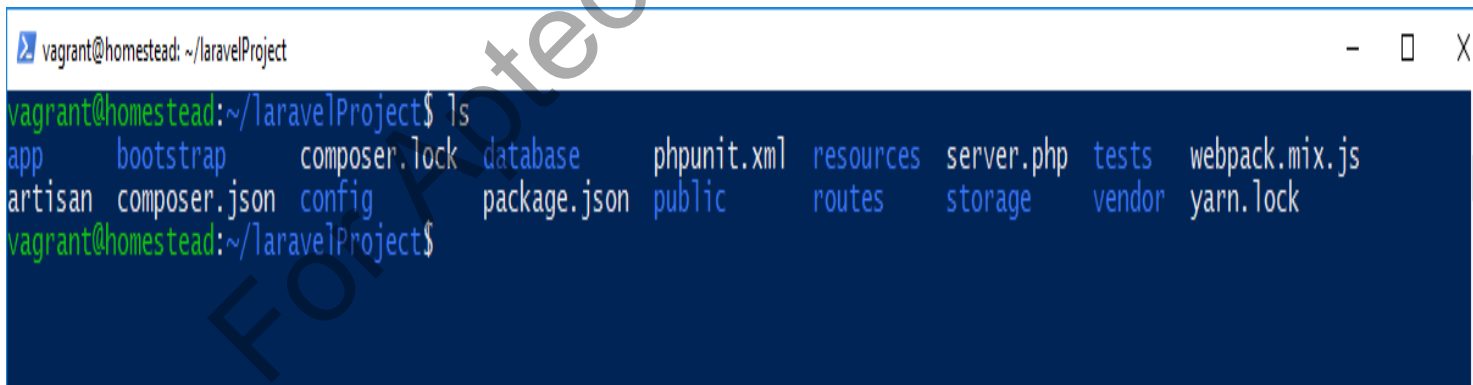
Creating a Laravel application:

```
laravel new laravelProject
```

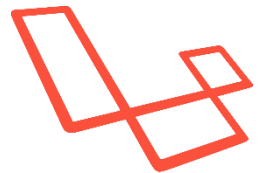
View the directory structure:

```
cd laravelProject  
ls
```

Output of the command :

A terminal window screenshot showing the output of the 'ls' command. The terminal title is 'vagrant@homestead: ~/laravelProject'. The command 'ls' has been executed, and the output lists the following files and directories: app, bootstrap, composer.lock, database, phpunit.xml, resources, server.php, tests, webpack.mix.js, artisan, composer.json, config, package.json, public, routes, storage, vendor, and yarn.lock. The prompt 'vagrant@homestead: ~/laravelProject\$' is visible at the bottom.

```
vagrant@homestead: ~/laravelProject  
vagrant@homestead:~/laravelProject$ ls  
app      bootstrap  composer.lock  database  phpunit.xml  resources  server.php  tests  webpack.mix.js  
artisan  composer.json  config        package.json  public       routes     storage     vendor  yarn.lock  
vagrant@homestead:~/laravelProject$
```



LARAVEL DIRECTORY STRUCTURE (2-2)



SUMMARY

- Laravel is an open source PHP framework for users working on Web application development.
- A framework is a basic templating structure that is used to develop, test, and deploy applications rapidly and efficiently.
- The Laravel framework reuses and assembles existing components, dependencies, and libraries to automate time-consuming tasks.
- Laravel includes a consistent API.
- Laravel has many conventions, if followed, can predict the dependencies and automate the configuration related tasks.
- Laravel is based on the Model–View–Controller (MVC) architecture.
- Laravel utilizes helper functions in its classes, functions, and even configurations.

