Top Laravel Interview Questions

1. What is the latest Laravel version?

The latest Laravel version is 9. It was launched on February 8th, 2022.

2. Define Composer.

Laravel is a popular web application framework that allows you to build dynamic websites and applications.

A composer is a tool that includes all the dependencies and libraries. It helps the user to develop a project concerning the mentioned framework. Third-party libraries can be installed easily using composer.

Composer is used to managing its dependencies, which are noted in the composer.json file and placed in the source folder.

3. What is the templating engine used in Laravel?

The Laravel Blade templating engine is a powerful piece of the framework that allows you to easily create powerful templates with a syntax that's simple and intuitive.

The Blade templating engine provides structure, such as conditional statements and loops. To create a blade template, you just need to create a view file and save it with a .blade.PHP extension instead of a .php extension. The blade templates are stored in the /resources/view directory. The main advantage of using the blade template is that we can create the master template, which other files can extend.

4. What are available databases supported by Laravel?

Laravel has you covered. The database configuration file is app/config/database.php. You can define your database connections in this file and specify which you should use reference. Examples for all of the supported database systems are provided in this file.

Laravel supports four database systems: [MySQL](https://www.simplilearn.com/tutorials/mysql-tutorial), [Postgres](https://www.simplilearn.com/tutorials/sql-tutorial/postgresql-tutorial), [SQLite](https://www.simplilearn.com/tutorials/sql-tutorial/what-is-sqlite), and [SQL Server.](https://www.simplilearn.com/what-is-microsoft-sql-server-architecture-article)

5. What is an artisan?

The artisan script is a command-line interface included with Laravel. It's the first thing you'll see when you run composer create-project, or PHP artisan serve.

Artisan is made up of commands and is one of your best friends for developing and managing your Laravel applications. You can view a list of all available Artisan commands by running PHP artisan list.

6. How to define environment variables in Laravel?

 In Linux, you have probably become familiar with environment variables. You can check the available environment variables with the printenv command.

To define an environment variable in Linux, use the export command followed by your new variable name: export name=Simplilearn.

The .env file holds your env variables for your current environment. The DotEnv Library powers it.

As the .env file often holds sensitive information like API keys or database credentials, you should never commit it to Git and push it to GitHub.

7. Can we use Laravel for Full Stack Development (Frontend + Backend)?

Laravel is a great choice for building full-stack web applications. With Laravel, you can create a backend that will be scalable, and the frontend can be built using blade files or SPAs using [Vue.js,](https://www.simplilearn.com/vue-js-interview-questions-article) which is provided by default. But it can also be used to just provide APIs for a SPA application.

Learn the Ins & Outs of Software Development

Caltech Coding Bootcamp[EXPLORE PROGRAM](https://www.simplilearn.com/coding-bootcamp?source=GhPreviewCTABanner)

Learn the Ins & Outs of Software Development

8. How to put Laravel applications in maintenance mode?

Laravel makes it easy to manage your deployment with minimal effort. Laravel allows you to quickly and easily disable your application while updating or performing maintenance when you need to make changes to your server or database.

To enable maintenance mode, the following are some helpful laravel commands related to maintenance mode:

# enable maintenance mode

php artisan down

# disable maintenance mode

php artisan up

# if you want the client to refresh

# page after a specified number of seconds

php artisan down --retry=60

9. What are the default route files in Laravel?

You can define Laravel routes in your routes/web.php file or create a separate file for other types of routes.

All routes are defined in your route files, located in the routes directory. The Laravel framework automatically loads these files. The routes/web.php file defines routes for your web interface. These routes are assigned to the web middleware group, providing features like session state and CSRF protection. The routes in routes/api.php are stateless and set in the API middleware group.

For most applications, you will begin by defining routes in your routes/web.php file. You may access the routes described in routes/web.php by entering the designated route's URL in your browser or through one of your controllers' actions or methods (explained later).

10. What are migrations in Laravel?

Migration is a feature of Laravel that allows you to modify and share the application's database schema. It will enable you to alter the table by adding a new column or deleting an existing column.

If you have ever had to tell a teammate to add a column to their local database schema manually, you've faced the problem that database migrations solve. Migrations are like version control for your database, allowing your team to modify and share the application's database schema. Migrations are typically paired with Laravel's schema builder to build your application's database schema.

The Laravel Schema facade provides database agnostic support for creating and manipulating tables across all of Laravel's supported database systems.

11. What are seeders in Laravel?

Laravel's database seeding feature allows you to quickly insert data into your database. It is helpful for development environments where you may not have access to your production database.

Laravel includes the ability to seed your database with data. By default, a Database seeder class is defined for you. You may use the call method from this class to run other seed classes. All seed classes are stored in the database/seeders directory.

A seeder class only contains one method: run. This method is called when the db:seed Artisan command is executed. You may use the query builder to insert data or Eloquent model factories.

12. What are the factories in Laravel?

Laravel has an excellent model factory feature that allows you to build fake data for your models. It is beneficial for testing and seeding counterfeit data into your database to see your code in action before any accurate user data comes in.

By default, Laravel's database seeding feature will create a new row in the database table and insert the value into each field. But sometimes, you might want a few extra areas or some sort of random string instead of a numeric value. That's where model factories come in handy!

Model Factories allow you to create a new model instance using their rules. You can do anything from creating an empty model instance to creatinbuildingth all fields filled out with values or even random ones!

13. How to implement soft delete in Laravel?

 Laravel 5.6 has a new feature called soft deletes. When soft deleted models, they aren't removed from our database. Instead, a deleted\_at timestamp is set on the record.

To enable soft deletes for a model, you have to specify the soft delete property on the model like this:

Use Illuminate\Database\Eloquent\SoftDeletes;

Use SoftDeletes; in our model property.

After that, when you use the delete() query, PHP will not remove records from the database. Then a deleted\_at timestamp is set on the record.

Learn the Ins & Outs of Software Development

Caltech Coding Bootcamp[EXPLORE PROGRAM](https://www.simplilearn.com/coding-bootcamp?source=GhPreviewCTABanner)

Learn the Ins & Outs of Software Development

14. What are Models?

 Laravel is a framework that follows the Model-View-Controller design pattern. All your models, views, and controllers are stored in their directories, making it easy to keep track of everything.

You'll use controllers to handle user requests and retrieve data by leveraging models. Models interact with your database and recover your objects’ information. Finally, views render pages.

Laravel comes with a fantastic, built-in command line interface called Artisan CLI that provides complete commands to help you build your application.

15. What is the Laravel Framework?

Laravel is an open-source [PHP framework](https://www.simplilearn.com/tutorials/php-tutorial/php-framework), which is robust and easy to understand. It follows a model-view-controller design pattern. Laravel reuses the existing components of different frameworks which helps in creating a web application. The web application thus designed is more structured and pragmatic.

Laravel offers a rich set of functionalities that incorporates the basic features of PHP frameworks like CodeIgniter, Yii, and other [programming languages](https://www.simplilearn.com/best-programming-languages-start-learning-today-article) like [Ruby on Rails.](https://www.simplilearn.com/tutorials/java-tutorial/what-is-ruby-on-rails) Laravel has a very rich set of features that will boost the speed of web development.

With Laravel, you can build applications for any type of business or organization. Whether it’s eCommerce, social media marketing, or an online ticketing system, you can create any type of web application with Laravel because it’s flexible and scalable enough to accommodate any size project easily.

16. What's New in Laravel 8?

Laravel 8.0 is the latest version of Laravel Framework. The Laravel 8.0 was released on September 8th, 2020, with the latest and unique features making it one of the top frameworks in today's world. This latest version brings many new features, such as Laravel Jetstream, model directory, migration squashing, rate limiting improvements, model factory classes, time testing helpers, dynamic blade components, and more.

Following are some of the new features in Laravel 8

* Time Testing Helpers
* Models Directory
* Migration Squashing
* Laravel Jetstream
* Rate Limiting Improvements
* Model Factory Classes
* Dynamic Blade Components

17. How to enable query log in laravel?

Our first step should be

DB::connection()->enableQueryLog();

After our query, you should place it

$querieslog = DB::getQueryLog();

After that, you should place it

dd($querieslog)

18. What is Middleware in Laravel?

Middleware in laravel is a platform that works as a bridge between the request and the response. The main aim of middleware is to provide the mechanism for investigating HTTP requests entering into your application. For instance, middleware in laravel ensures that the user of your particular application is authenticated. If they find that the user is not authenticated, it will redirect the user to the main login page of the application.

Middleware in laravel also helps you to handle a request from a user who has already been authenticated. For example, if you want to display information about a user who has already been established, then middleware will help you by providing this functionality within your application.

Learn the Ins & Outs of Software Development

Caltech Coding Bootcamp[EXPLORE PROGRAM](https://www.simplilearn.com/coding-bootcamp?source=GhPreviewCTABanner)

Learn the Ins & Outs of Software Development

19. What is reverse Routing in Laravel?

Reverse routing is the process used to generate the URLs based on the names or symbols. It is also known as backtracking or reverse mapping.

You can do reverse routing in two ways:

1. Using route parameters in routes
2. Using route names in routes

When you use the URL structure, you can simply add a symbol or name at the end of your URL. It will let you create more readable URLs in your application and make it easier for users to understand them.

Route:: get(‘list’, ‘blog@list’);

{{ HTML::link\_to\_action('blog@list') }}

20. What is a Service container?

A Service container is one of the most powerful tools used to manage dependencies over the class and perform dependency injections.

* A service container can be used as a registry to keep track of all the classes in use within your application.
* In addition, it also helps in binding interfaces to concrete classes.

21. What is Auth? How is it used?

Laravel Auth is an in-built functionality provided by Laravel to identify the user credentials with the database. It takes input parameters like username and password for user identification. If the settings match, then the user is said to be authenticated.

If you want to authenticate your laravel application, then you can use the auth function.

22. How to mock a static facade method in Laravel?

 Laravel facades provide a static interface to classes available inside the application's service container. They're used to provide a simple way to access complex objects and methods, and they're often used to centralize the configuration of those objects.

Facades can be mocked in Laravel using the shouldRecieve method, which returns an instance of a facade mock.

$value = Cache::get('key');

Cache::shouldReceive('get')->once()->with('key')->andReturn('value');

23. What is composer lock in laravel?

After you run composer install in your project directory, the Composer will generate a composer.lock file. It will record all the dependencies and sub-dependencies installed by the composer.json.

24. What is Dependency injection in Laravel?

In Laravel, dependency injection is a term used for the activity of injecting components into the user application. It’s a critical element of agile architecture. The Laravel service container is a powerful tool that manages all class dependencies and performs dependency injection.

The Laravel service container is a tool that manages all class dependencies and performs dependency injection.

public function \_\_construct(UserRepository $data)

{

    $this->userdata = $data;

}

25. How to use skip() and take() in Laravel Query?

 If you're looking for a way to limit the number of results in your query, skip() and take() are two great options!

skip() is used to skip over several results before continuing with the query. Take() is used to specify how many results you want from the query.

For example, if you want to get only five results from a query that usually returns 10, you'd use skip(5). If you're going to start at the 6th result in the question and get everything after it (skipping 1), then you'd use take(1).

26. What is the Repository pattern in laravel?

The repository pattern decouples data access layers and business logic in our application. It allows objects without having to know how these objects persisted. Your business logic does not need to understand how data is retrieved. The business logic relies on the repository to get the correct data.

This pattern makes our code cleaner and easier to maintain because we can change the implementation of our data layer without changing any related code.

Learn the Ins & Outs of Software Development

Caltech Coding Bootcamp[EXPLORE PROGRAM](https://www.simplilearn.com/coding-bootcamp?source=GhPreviewCTABanner)

Learn the Ins & Outs of Software Development

27. What is the Singleton design pattern in laravel?

 The Singleton Design Pattern in Laravel is one where a class presents a single instance of itself. It is used to restrict the instantiation of a class to a single object. It is useful when only one example is required across the system. When used properly, the first call shall instantiate the object. After that, Laravel shall return all calls to the same instantiated object.

When working with an application that requires multiple instances of an object, it can be hard to manage them all and ensure another part of your codebase is not changing them.

There are many ways you could handle this issue:

* Use a static variable inside your class and update it manually each time you want to use it
* Use a global variable and check if it's set before using it
* Use dependency injection (you'll need this if you want to inject services into your classes)

28. What are the advantages of Queue?

Laravel queues are a great way to handle time-consuming tasks. They allow you to offload work from your web server, so your users don't have to wait for a response from your API before getting the next page.

Queues are also helpful if your application has multiple servers and you want to use them all without having jobs interfere with each other. For example, if one server is running PHP code while another is running Python code, they shouldn't be able to interfere with each other's processes.

29. What is tinker in Laravel?

Laravel Tinker is a powerful REPL tool used to interact with the Laravel application with the command line in an interactive shell. Tinker came with the release of version 5.4 and is extracted into a separate package.

Tinker is an excellent tool for debugging and exploring your code. You can use it to inspect variables, models, classes, and methods at runtime. You can change anything in the console, and it will instantly update the page you are working on.

For Installing tinker, you can use composer require laravel/tinker.To execute tinker, we can use the php artisan tinker command.

30. What is a REPL?

 REPL stands for Read—Eval—Print—Loop. It's a type of interactive shell that takes in single user inputs, processes them, and returns the result to the client.

If you've ever used a programming language like Ruby or Python, you've probably used a REPL. They're extremely useful for testing snippets of code and running little experiments on your computer.

31. What are the basic concepts in laravel?

* Blade Templating
* Routing
* Eloquent ORM
* Middleware
* Artisan(Command-Line Interface)
* Security
* In-built Packages
* Caching
* Service Providers
* Facades
* Service Container

32. What is a lumen?

 Lumen is a micro PHP framework introduced by Taylor Otwell, the creator of Laravel. It is a faster, smaller, and leaner version of a full web application framework that uses the same components as Laravel, but especially for microservices.

Lumen is a lightweight framework for building web applications in PHP. It is built on top of Laravel and follows the same best practices that you have come to know in Laravel. In fact, Lumen was designed to be used as an alternative to Laravel when you only need to build small applications and services.

Lumen is ideal for both new projects and existing projects that require a microservice architecture or fast prototypes.

To install Lumen you can use the following command.

composer global require "laravel/lumen-installer=~1.0"

33. How do I stop Artisan service in Laravel?

If you're having trouble with your server, here are a few steps to help you troubleshoot.

First, try pressing Ctrl + Shift + ESC. This will open up the task manager, where you can locate the php system walking artisan process and kill it with a proper click.

Next, reopen your command line and begin again the server.

Note that it may be possible to kill the process just by sending it a kill sign with Ctrl + C.

34. What are the features of Laravel?

* Offers a rich set of functionalities like Eloquent ORM, Template Engine, Artisan, Migration system for databases, etc
* Libraries & Modular
* It supports MVC Architecture
* Unit Testing
* Security
* Website built in Laravel is more scalable and secure.
* It includes namespaces and interfaces that help to organize all resources.
* Provides a clean API.

35. What is validation in Laravel?

Laravel provides several different ways to validate your application's incoming data. The most common way is by creating a Form Request.

Form Requests allow you to validate incoming data with ease, so you don't have to worry about creating validation rules or manually checking for errors. Form Requests also support custom validation rules and custom error messages, which means you can build your own validations that are specific to your application.

Learn the Ins & Outs of Software Development

Caltech Coding Bootcamp[EXPLORE PROGRAM](https://www.simplilearn.com/coding-bootcamp?source=GhPreviewCTABanner)

Learn the Ins & Outs of Software Development

36. What is a yield in Laravel?

@yield is a Blade directive that allows you to pull content from a child page into a master page, and it's used in Laravel to define sections in a layout. When the Laravel framework performs the blade file, it first checks to see if you have extended a master layout. If you have, it will move on to the master layout and start grabbing content from @sections.

37. What is Nova?

Laravel Nova is an admin panel built on the Laravel Framework. It's perfect for managing your database records, and it's easy to install and maintain.

Laravel Nova comes with features that have the ability to administer your database records using Eloquent.

38. Explain ORM in Laravel.

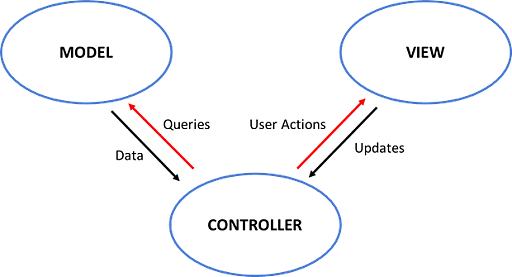
ORM stands for Object-Relational Mapping. It is a programming technique that is used to convert data between incompatible type systems in object-oriented programming languages.

The ORM is used to map objects in the application's domain model to relational database tables, and vice versa. In this way, the ORM lets you work with your domain objects as if they were an old-fashioned collection of fields and properties while keeping the more recent advantages of a relational database.

39. Explain MVC Architecture

MVC stands for Model View Controller. It segregates domain, applications, business and logic from the user interface. This is achieved by separating the application into three parts:

* Model: Data and [data management](https://www.simplilearn.com/what-is-data-management-article) of the application
* View: The user interface of the application
* Controller: Handles actions and updates



Learn the Ins & Outs of Software Development

Caltech Coding Bootcamp[EXPLORE PROGRAM](https://www.simplilearn.com/coding-bootcamp?source=GhPreviewCTABanner)

Learn the Ins & Outs of Software Development

40. What is Routing?

[Routing](https://www.simplilearn.com/tutorials/angular-tutorial/angular-routing) refers to accepting requests and sending them to the relevant function of the controller. The route is a method of creating the request URL of the application. These URLs are readable and also SEO-friendly. Routes are stored under the /routes folder inside files. The default Routes in Laravel are used for registering:

* web.php: web routes
* api.php: API routes
* console.php: console-based commands
* channel.php: broadcasting channels supported by the application

41. What is the use of Bundles in Laravel?

Popularly known as packages, Bundles are a convenient way to group code. Bundles extend the functionality of Laravel since they can have views, configuration, migration, tasks and more. A Bundle can range from database ORM to an authentication system.

42. Explain Seeding.

Seeding refers to introducing test data to the database for testing the application. Developers can add dummy data to their database tables using a database seeder. Different data types allow the developer to detect bugs and improve performance.

43. How do you check the installed Laravel version of a project?

Using the command PHP Artisan --version or PHP Artisan -v

44. Which Artisan command gives a list of available commands?

PHP Artisan list

45. What is the difference between the Get and Post method?

Both Get and [Post](https://www.simplilearn.com/tutorials/php-tutorial/php-post-method) are used to retrieve input values in Laravel. A limited amount of data in the header is allowed in the Get method, whereas the Post method allows sending large amounts of data in the body.

46. What are some common Artisan commands in Laravel?

make:controller – Creates a new Controller file in App/Http/Controllers folder

make:model – Creates a new Eloquent model class

make:migration – Creates a new migration file

make:seeder – Creates a new database seeder class

make:request – Creates a new form request class in App/Http/Requests folder

make:command – Creates a new Artisan command

make:mail – Creates a new email class

make:channel – Creates a new channel class for broadcasting

47. Explain the project structure in Laravel.

The following list of directories/folders shows the typical project structure in Laravel

* app: source code of the application resides here, and it has the following sub-folders:

1. Consoler
2. Exceptions
3. Http
4. Providers

* bootstrap: contains files required to bootstrap an application and configure auto-loading
* config: configuration files such as app.php, auth.php, broadcasting.php, cache.php, database.php and so on are stored in this folder
* database: holds the database files and has .gitignore file and the following sub-folders:

1. factories
2. migration
3. seeds

* public: contains files used to initialise the web application
* resources: The resource directory contains files to enhance the web application and has the following sub-folders:

1. Assets
2. Lang
3. Views

* routes: includes route definitions
* storage: stores the cache and session files and has sub-folders:

1. app
2. framework
3. logs

* test: holds the automated unit test cases
* vendor: contains the composer dependency packages

48. Give an example to describe how a Route is created.

Routes are created in the routes folder. The files web.php and api.php have the routes that have been created for websites and API, respectively.

Route::get('/', function () {

   return view('welcome');

});

The above Route is defined for the homepage, and it returns the view “Welcome” every time it gets a request for /.

49. Name the template engine used in Laravel?

The name of the template engine in Laravel is Blade. It enables you to use the mustache-like syntax with plain PHP. It then gets compiled and cached until the blade file changes. The file has a .blade.PHP extension.

50. What is soft delete in Laravel?

You don't delete data in soft delete but add a deleted tag. This way, you can quickly restore data when needed.

51. Describe localization in Laravel?

Serving the content according to the client's language preference is localization.

52. What are Laravel's registries?

You use Requests to interact with HTTP requests and session cookies. For doing this, you can use the class illuminate\Http\Request. After submitting a request to a Laravel route, the request object is available within the method using dependency injection.

53. How does request validation happen in Laravel?

You can either create a request validation class or a controller method for request validation. For example:

public function store (Request $request)

{

$validated = $request->validate([ 'title' => 'required|unique:posts|max"255', 'body' => 'required',]);

}

54. Describe the service provider in Laravel.

You can register services, events, etc., using a service provider before booting the application. They help inject Laravels services or your application services and dependencies.

55. Explain register and boot methods in the service provider class.

The register method in the Service Provider class binds classes and services to Service Containers. And the boot method runs after the program includes all the dependencies in the container. Then you can create routes and view composers in the boot method.

56. How to create a middleware in Laravel?

Using a middleware, you can filter and inspect HTTP requests. You can create a middleware using the following code.

PHP artisan make middleware AllowSmallFile

57. Define collections in Laravel.

A collection is an API wrapper for PHP array functions; you can generate it from an array. Using a collection, you can reduce or map arrays.

58. Explain queues in Laravel?

Time-consuming tasks can make an interface unresponsive. You must run such tasks in the background. In Laravel, queues are a way to run these tasks in the background so that your main threads will remain responsive.

59. Define accessors and mutators.

Accessors allow you to change the data after acquiring it from the database. And mutators enable you to modify data before saving it to a database.

60. What are relationships in Laravel?

If you're familiar with working with Eloquent models, you know that relationships are defined as methods in your Eloquent model classes. Since relationships also serve as influential query builders, defining relationships as methods provides powerful method chaining and querying capabilities.

Following are the types of relationships in Laravel

* One To One relationship.
* One To Many relationships.
* Many To Many relationships.
* Has Many Through relationships.
* Polymorphic relationships.
* Many To Many Polymorphic relationships.

61. What is Eloquent in Laravel?

 Laravel is a PHP framework that allows you to develop web applications quickly. It is easy to learn, comes with a lot of prebuilt functionality, and makes it easy for developers to create complex websites and other applications.

It's built on top of Symfony components, which you can use for different websites, including e-commerce sites and business apps. The framework also includes an ORM (Object Relational Mapper) called Eloquent, which is used to communicate with the database.

62. What is throttling and how to implement it in Laravel?

Throttling is a great way to rate-limit requests from a particular IP. This can be used to prevent DDOS attacks as well. Laravel provides a middleware that can be applied to routes, which can be added to the global middleware list as well.

You can implement throttling as below:

Route::middleware('auth:api', 'throttle:60,1')->group(function () {

    Route::get('/user', function () {

        //

    });

});

63. What are facades?

A facade is a way to access classes available in the application's service container. The service container holds all of your application's business logic, and facades provide a "static" interface to those classes.

There are lots of Laravel facades, and they're all over the place in the framework! You can see them in any controller or view file.

You can access a facade as mentioned below

use Illuminate\Support\Facades\Cache;

use Illuminate\Support\Facades\Route;

Route::get('/cache', function () {

   return Cache::get('key');

});

64. What are Events in Laravel?

Events are great for decoupling different parts of your application. For example, suppose you want to send a Slack notification each time an order has been shipped instead of coupling your order processing code to your Slack notification code. In that case, you can raise an App\Events\OrderShipped event that the listener can receive and use to dispatch a Slack notification.

Each time you generate an event or listener, the event will store it in its directory.

Events are typically stored in the app/Events directory, while their listeners are stored in app/Listeners. You can see these directories by running `PHP artisan make: event OrderShipped.`

You don't have to worry about these directories if you don't see them in your application. They'll be created for you as you generate events and listeners using Artisan console commands.

65. Explain logging in Laravel?

Logging can be a powerful tool for tracking down bugs, but it can also be overwhelming to manage. Laravel helps you make sense of your log messages by providing an easy-to-use logging system that allows you to write log messages to files, the system error log, Slack, and more.

Laravel logging is based on "channels." Each channel represents a specific way of writing log information. Log messages may be written to multiple channels based on their severity.

Under the hood, Laravel utilizes the Monolog library, which supports a variety of powerful log handlers. Laravel makes it a cinch to configure these handlers, allowing you to mix and match them to customize your application's log handling.

Frequently Asked Questions

1. Why is Laravel used?

Using Laravel, you can simplify the development process. It simplifies the regular tasks such as authentication, routing, and sessions. This is one of the most common Laravel Interview questions.

2. What is MVC in Laravel?

MVC means Model View Controller and is the architecture developers use when they build an application. MVC architecture will enable you to understand how the application's data flow works.

3. What is namespace in Laravel?

Namespaces are an element's class where each element has a unique name. You can share them with elements in other classes.

4. Does Laravel support Bootstrap?

Yes.

5. Name the aggregate methods of the Query Builder.

The Query Builder provides variety of aggregate methods such as:

count(), max(), min(), avg() and sum().

6. How is a Blade template file identified?

Blade template files have .blade.php extension and are located in the resources/views folder.

7. Name the ORM used in Laravel.

Eloquent is the ORM (Object Relational Mapper) used in Laravel.

8. What is Vapor?

A completely serverless deployment and auto-scaling platform for Laravel. It is powered by Amazon Web Services (AWS) Lambda.

9. Name some common tools used to send emails in Laravel.

* SwiftMailer
* SMTP
* Mailgun
* Mailtrap
* Mandrill
* Postmark
* Amazon SES (Simple Email Service)
* Sendmail

10. What is Forge?

Serve Management & Application Deployment Service.

11. Name a few competitors of Laravel.

* CodeIgniter
* Angular
* Symfony
* Slim Framework
* Modx
* Yii
* CakePHP
* Placon
* Scriptcase