**LARAVEL FRAMEWORK**

1. What is the latest Laravel version?

Laravel 10 was scheduled to be released on February 7, 2023, following the release of Laravel 9 on February 8, 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" \o "MySQL" \t "_blank), [Postgres](https://www.simplilearn.com/tutorials/sql-tutorial/postgresql-tutorial" \o "Postgres" \t "_blank), [SQLite](https://www.simplilearn.com/tutorials/sql-tutorial/what-is-sqlite" \o "SQLite" \t "_blank), 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.

### 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.