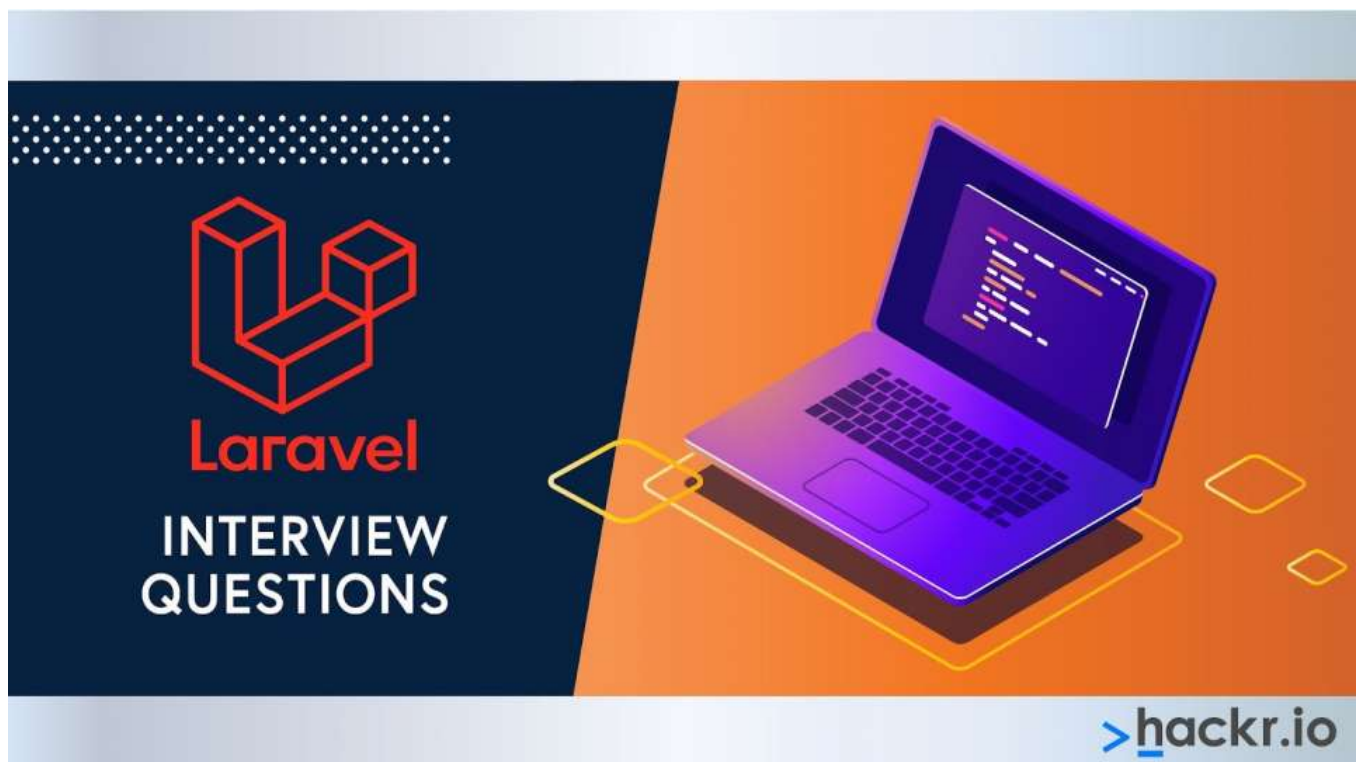


Are you looking to get a discount on popular programming courses? Then click here. [View offers](#)

Disclosure: Hackr.io is supported by its audience. When you purchase through links on our site, we may earn an affiliate commission.

Top 50 Laravel Interview Questions and Answers in 2022

Posted in [Laravel](https://hackr.io/blog/category/laravel) (<https://hackr.io/blog/category/laravel>)



Swapnil Banga

(<https://hackr.io/blog/author/swapnil-banga>)

Last Updated 03 Aug, 2022

Share:



(<https://twitter.com/intent/tweet?text=Top+50+Laravel+Interview+Questions+and+Answers+in+2022+https%3A%2F%2Fhackr.io%2Fblog%2Ftop-laravel-interview-questions-and-answers>)



(<https://www.linkedin.com/shareArticle?mini=true&url=https://hackr.io/blog/top-laravel-interview-questions-and-answers>)



(<http://www.reddit.com/submit?url=https://hackr.io/blog/top-laravel-interview-questions-and-answers>)



(<https://news.ycombinator.com/submitlink?u=https://hackr.io/blog/top-laravel-interview-questions-and-answers>)



(<https://api.whatsapp.com/send?text=https%3A%2F%2Fhackr.io%2Fblog%2Ftop-laravel-interview-questions-and-answers>)

Table of Contents



Laravel is an open-source PHP web framework (<https://hackr.io/tutorials/learn-laravel>) used for developing responsive web applications. It was created by Taylor Otwell and follows the Model-View-Controller (MVC) architecture design pattern. It is also a free web framework licensed under the MIT Licence.

Today, Laravel developers (<https://hackr.io/blog/laravel-cheat-sheet>) are in great demand. However, to become a PHP or Laravel developer, you have to have a good deal of knowledge and experience, including:

- Hands-on experience working with the Laravel framework
- A bachelor's degree in Computer Science or any other relevant course
- Sound knowledge of object-oriented PHP and the Laravel 5 PHP framework (<https://hackr.io/blog/best-php-frameworks>)
- Proficient in working with SQL schema design, REST API design, and SOLID principles
- Knowledge of software testing, MySQL profiling, and query optimization

This article explores frequently asked Laravel interview questions and answers. It includes both basic questions for beginners and more sophisticated ones as well. If you aim to appear for a Laravel interview and do well, the following questions and answers can definitely help.

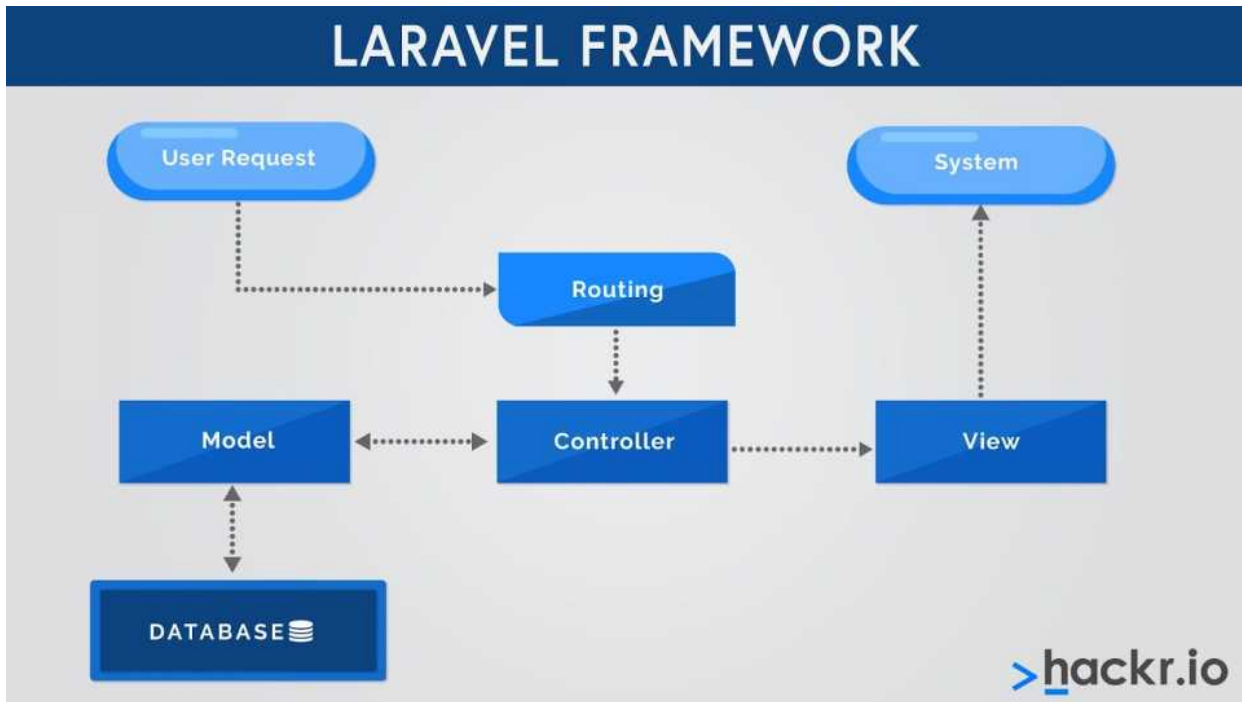
Let's get started.

Laravel Interview Questions

1. What is Laravel?

Laravel is an open-source and new generation web framework for PHP (<https://hackr.io/blog/best-php-frameworks>), developed by Taylor Otwell in 2011. It is specially designed to develop web-based applications, and follows the MVC model, suitable for creating simple, elegant, and well-structured applications. It has a current stable release with version 8 that was released on 8th Sept 2020.

Laravel comes with a framework called Lumen built on top of the Laravel components, making it a perfect option for creating a Laravel based microservices application.



2. What are the pros and cons of Laravel?

The following are the pros and cons of the Laravel framework.

Pros of Laravel:

- For managing the project dependencies, Laravel uses the Composer allowing developers to mention the package name, version, and the pre-built functionalities that are ready for use in your project, resulting in faster development of the application
- It comes with a blade templating engine that is easy to learn and understand. This is useful while working with the PHP/HTML languages. Web development via Laravel allows the composing of the plain PHP codes in a layout shape, thus helping in improving the execution of complex tasks.
- You can learn Laravel via Laracast (<https://laracasts.com/browse/frameworks>) that comes with both free and paid videos for easy learning
- Laravel has an object-relational mapper named Eloquent that helps implement the active record pattern and interact with the relational database. It is available as a package for Laravel.
- Laravel has a built-in command-line tool called Artisan support, where users can carry out repetitive tasks quickly and effortlessly
- With the help of the Laravel schema, developers can create database tables and add desired columns via writing a simple migration script
- It comes with a reverse routing feature that makes your application more flexible

Cons of Laravel:

- It is challenging to migrate legacy system to Laravel
- It comes with heavy documentation that might be difficult to understand for beginners
- Upgrades aren't smooth

3. What are events in Laravel?

Events are actions recognized and handled by the program. These events work on the Observer-subscriber pattern. All the events in Laravel are stored within the app/Events directory, and the lists are stored within the app/listeners directory. Events can decouple the applications' aspects as a single event, which is capable of handling multiple listeners.

4. What is validation in Laravel?

In Laravel, ValidatesRequests is a trait used by classes to validate the input provided by the user. To store the data, we normally use the create or store methods defined at the Laravel routes with the get method.

You can get the Laravel validate method in the Illuminate\Http\Request object. If there is no error in the rule and it passes successfully, the code will execute as expected. But if there is any failure while validation, the code will not run and the user will get the error response for the HTTP request.

Below is an example of how validation rules are defined in Laravel:

```
/**
 * Store a post.
 *
 * @param Request $request
 * @return Response
 */
public function store(Request $request)
{
    $validatedData = $request->validate([
        'title' => 'required|unique:posts|max:255',
        'body' => 'required',
    ]);
}
```

In the code above, the title and the body are the required fields. The validation rule is sequential, so if there is any failure in any validation, further validation will not be checked.

5. How do you install Laravel via composer?

The composer comes as a dependency manager. If it is not installed on your system, you can install it using this link (<https://getcomposer.org/download/>).

After successfully installing the composer on your system, you need to create a project directory for your Laravel project. Later, you need to navigate the path where you have created the Laravel directory and run the following command:

```
composer create-project laravel/laravel --prefer-dist
```

This command will help install the Laravel in the current directory. If you want to start Laravel, run the following command.

```
php artisan serve
```

Laravel will get started on the development server. Now, run <http://localhost:8000/> (<http://localhost:8000/>) on your browser. The server requirement is as follows for installing Laravel.

- PHP >= 7.1.3.
- OpenSSL PHP Extension
- PDO PHP Extension
- Mbstring PHP Extension

- Tokenizer PHP Extension
- XML PHP Extension
- Ctype PHP Extension
- JSON PHP Extension

6. What is a PHP artisan in Laravel?

PHP artisan is a command-line tool available in Laravel that comes with a variety of useful commands which help create an application quickly and hassle-free. You will get commands for every important task by default, like database seeding, migration, configuring cache, and many others.

The following are some important PHP artisan commands:

- php artisan make:controller : Used for creating a Controller file
- php artisan make:model : Used for making a Model file
- php artisan make:migration : Used for making the Migration file
- php artisan make:seeder : Used for making the Seeder file
- php artisan make:factory : Used for making the Factory file
- php artisan make:policy : Used for making the Policy file
- php artisan make:command : Used for making a new artisan command

7. What is middleware in Laravel?

Middleware provides a mechanism that helps filter the incoming HTTP request to your application. The basic middleware is explained with authentication. If the user is not authenticated, they will be redirected to the login page, and if the user is authenticated, they will be allowed for further processing. All this is possible with the help of the middleware.

Laravel has a php artisan make:middleware <middleware_name> command, helping define the new middleware within your application. By default, the middleware will get stored in the app/Http/Middleware directory.

If you want to run middleware for every HTTP request, list the middleware class within the \$middleware property of the app/Http/Kernel.php class. If you want to assign middleware specifically, assign it in the key-value pair at the app/Http/Kernel.php class \$routeMiddleware property.

8. What template is used by the Laravel engine?

Laravel comes with the Blade templating engine, which helps users use the plain PHP code in the view and then compiles and caches that view until the next modification. You can get the files related to the blade views in the resources/views directory with the extension .blade.php.

Below is an example of the Blade file.

```
<!-- /resources/views/alert.blade.php -->
<div class="alert alert-danger">
    {{ $slot }}
</div>
```

In the variable \$slot, you can assign any desired value to inject into the component.

The Component will look as follows:

```
@component('alert')
<strong>Whoops!</strong> Something went wrong!
@endcomponent
```

@component is the blade directive here.

9. Explain CSRF protection and CSRF token in Laravel.

CSRF is defined as a cross-site forgery attack, a type of malicious exploit where the authenticated users run unauthorized commands. In such a case, Laravel will generate CSRF tokens automatically for each active user's session. This token will verify the authenticated user who is making that unauthorized request to the application.

```
<form method="POST" action="/profile">
  @csrf
  ...
</form>
```

Suppose you are creating an HTML form for your application. Make sure to include a hidden CSRF token field in the form so that the middleware will check for the field and validate it. VerifyCsrfToken middleware is included within the web middleware group. You can use the @csrf blade directive for generating the token field on your application's form.

If you are using a JS-driven application, the JS HTTP library will automatically attach the CSRF token to every HTTP request.

10. Explain the Laravel facade.

The Laravel facade offers a static interface for the classes available in the service container of the application. In Laravel, all the facades are stored within the Illuminate\Support\Facades namespaces. You can implement the facades easily without the need for injection, allowing you to use multiple facades within the same class. It comes with an expressive syntax, ensuring higher flexibility than traditional static class's methods.

Access facade will look like:

```
use Illuminate\Support\Facades\Cache;
Route::get('/cache', function () {
    return Cache::get('key');
});
```

11. What is Eloquent in Laravel?

Before proceeding with the concept of Eloquent, it is important to understand what ORM (object-relational mapping) is. ORM is a programming method that helps users convert data between the relational database and object-oriented programming languages. You can refer to it as an object-relational mapper.

Eloquent is a type of ORM commonly used in Laravel, allowing users to work with the database efficiently. Every database has a specific model for interacting with the tables.

Eloquent has the following types of relationships:

- One to One
- Many to One
- One to Many
- Many to Many
- Has one Through
- Has_many Through

12. What are the advantages of the Laravel framework?

The following are the advantages of using the Laravel framework:

- Free
- Helpful for simple configurations of the applications

- This framework is based on the MVC model
- Comes with a wide range of modules and libraries, helping developers to speed up the development process
- Ensures high performance and makes the routing process easier
- Offers an Eloquent ORM, allowing you to handle various database-related tasks
- Inbuilt facility for supporting the unit tests
- Comes with strong community support

13. What are the features of the Laravel framework?

The following are the significant features of the Laravel framework:

- Offers Eloquent ORM for handling database-related tasks
- Comes with a Query builder
- Offers an easy process for Reverse routing
- Offers Class auto-loading and comes with Restful controllers
- Comes with the Blade template engine
- You will have the option for the Lazy collection
- Unit testing is easier
- Offers Database seeding and supports easy migrations

14. What are the features included in the latest version of Laravel?

The latest stable version of Laravel is Laravel 8. Here are some good features of Laravel 8:

- Laravel Jetstream
- Models directory

- Model factory classes
- Migration squashing
- Time testing helpers
- Dynamic blade components
- Rate limiting improvements

15. How can you check the installed version of Laravel?

Firstly, open the command line terminal and navigate to the project directory. Next, execute any of the following commands to check the installed version of Laravel.

```
php artisan --version
```

or

```
php artisan -v
```

16. What is the project structure of a Laravel project?

This is the directory structure of any Laravel project:

- **app folder:** It contains the source code of an application and consists of five sub-folders, namely the Console folder, Exceptions folder, Http folder, Models folder, and Providers folder. These sub-folders further contain exception handlers, controllers, middleware, service providers, and models.

Note: In Laravel 7, you do not have any folder called Models. All model files are present inside the app folder rather than the app/Models folder.

- **bootstrap folder:** Contains the bootstrap files
- **config folder:** Contains the configuration files

- **database folder:** Includes the database-related files and three sub-folders, namely the factories folder, migrations folder, seeders folder, and the .gitignore file. These sub-folders further store the large set of data, database migrations, and seeds.
- **public folder:** Has the files necessary for initializing the application

- **resources folder:** Contains the files for the HTML, CSS, and JavaScript. It contains four sub-folders, namely the CSS folder, js folder, lang folder, and views folder.
- **routes folder:** Contains the route definitions
- **storage folder:** Consists of cache files, session files, and more
- **tests folder:** Contains test files, like unit test files.
- **vendor folder:** Contains all composer dependency packages
- **.env file:** Contains the environmental variables
- **composer.json file:** Contains dependencies
- **package.json file:** This file is for an application's frontend and is similar to the composer.json file

17. What are bundles in Laravel?

The bundles in Laravel are used for increasing the functionality of an application. You can also refer to bundles as packages containing configurations, routes, migrations, views, etc.

18. What is routing?

Routing is a technique that accepts incoming requests and sends them to the relevant function within the controller. There are two types of routing files available in Laravel, as mentioned below:

- web.php file in the routes folder
- api.php file in the routes folder

19. How can you create a route in Laravel?

If you want to create a route in Laravel, use controllers or add the code directly to the route. Below is an example, helping you create a route by adding the code directly.

Example: Replace the code in routes/web.php file and add the following code segment.

```
<?php
use Illuminate\Support\Facades\Route;

Route::get('/', function () {
    return "Welcome!";
});
```

Run the project in the browser, and you will observe 'Welcome!' as the output.

20. What is authentication in Laravel?

In Laravel, authentication is the process where you verify the users of an application. You can achieve this by identifying the username and password of users. Also, you can use another parameter for authentication. If the provided credentials are right, the user is authenticated; otherwise, the user is unauthenticated.

Laravel uses guards and providers for the authentication process. Guards will specify how the users get authenticated for each request, whereas the providers will specify how the users get retrieved from the persistent storage.

21. What is the difference between GET and POST methods in Laravel?

Here are some major differences between GET and POST methods in Laravel:

GET Method	POST Method
This method will request the data from a specific resource.	It will send the data to a server.
It will include the parameters within the URL.	It will include the parameters in the body.
The URL will display the data.	The URL will not display the data.
This method will allow only ASCII characters.	This method will allow both ASCII characters and binary data.
You can only use the limited data for the GET method.	There is no limitation on the data being used.
You can check the request in the browser history.	You cannot check the request in the browser history.
You can bookmark the GET request.	You cannot bookmark the POST request.
It can be cached.	It cannot be cached.
It is less secure as compared to the POST method.	It is highly secure as compared to the GET method.
You cannot use it for sending sensitive data such as passwords.	You can use it for sending sensitive data such as passwords.

22. What are some tools for sending emails in Laravel?

The following are some commonly used tools for sending emails in Laravel:

- Mailtrap
- Mailgun
- Mailchimp
- Mandrill
- Amazon Simple Email Service (SES)
- Swiftmailer
- Postmark

23. Explain reverse routing in Laravel.

Reverse routing is the process of generating URLs based on names or symbols and route declarations. With the help of reverse routing, the application becomes more flexible and offers a better interface, which makes writing code easier.

Example:

```
Route:: get('list', 'blog@list');

{{ HTML::link_to_action('blog@list') }}
```

24. What are service providers in Laravel?

Service providers in Laravel play a central role, helping configure all applications and core services. They are robust tools that help maintain class dependencies and perform an injection. It also provides instructions to Laravel for binding the various components within Laravel's service container.

For generating a service provider, use the following artisan command:

```
php artisan make: provider ClientsServiceProvider
```

Every service provider in Laravel will extend the `Illuminate\SupportServiceProvider` class and comes with the following two functions:

- Register()
- Boot()

25. What is a homestead in Laravel?

It is a pre-packed, official virtual machine, allowing developers to use all the necessary tools for developing Laravel. It comes with Ubuntu, Gulp, Bower, and other essential development tools allowing the development of full-scale web-based applications. It will provide a development environment used by developers without the need for installing PHP, a web server, or other server-related software on your machine.

26. Why is Laravel preferred over other PHP frameworks?

Below are the reasons for preferring Laravel over other PHP frameworks:

- Laravel allows fast setup and customization as compared to other options
- Comes with multiple file systems.
- Offers pre-loaded packages, such as Laravel Socialite, Laravel cashier, Laravel passport, etc.
- Built-in authentication system
- Provides Eloquent ORM for handling database-related operations
- Offers an “Artisan” command-line tool for running various commands for carrying out various functions in Laravel

27. What is dd() in Laravel?

Laravel provides the dd() function, allowing users to dump the content of variables to a browser and then stop the execution of the further script. The dd stands for the dump and dies, and it first dumps a variable or object and kills (die) the execution of the script. If you want, you can easily isolate this function in a reusable function file or a class.

28. What is yield in Laravel?

Laravel provides @yield for defining a section in a layout, getting the content from the child page, and submitting it to a master page. So whenever you use Laravel to execute the blade file, it first checks whether you have extended the master layout or not. If so, it moves to the master layout and commences getting the @sections.

29. What are requests in Laravel?

In Laravel, requests are used to interact with incoming HTTP requests, along with the sessions, cookies, and even files, if they are submitted with the requests. The class `\Illuminate\Http\Request` is responsible for requests in Laravel.

Whenever you submit any request to the Laravel route, it goes to the controller method, and with the help of the dependency injection, the object of that request will be available in the controller method. You can perform various actions with a request, such as validating and authorizing.

You can even create a request validation class that will store validation rules and associated error messages. Consider the following example:

```
/**
 * Store a new blog post.
 *
 * @param \Illuminate\Http\Request $request
 * @return \Illuminate\Http\Response
 */
public function store(Request $request)
{
    $validated = $request->validate([
        'title' => 'required|unique:posts|max:255',
        'body' => 'required',
    ]);

    // The blog post is valid...
}
```

30. What are register and boot methods in the service provider class?

In the service provider class, the register method is used for binding a class or services to the service controller. You cannot use it for accessing any other functionality or any class from your application as the service that you want to access might not get loaded yet in the container.

The boot method in the service provider class helps in running all the dependencies included in a container and you can access functionalities in the boot method.

31. What is dependency injection in Laravel?

The service controller of Laravel helps in resolving all the dependencies in all controllers. So, you can easily type-hint the dependency in controller methods or constructors. The dependency in methods will get resolved and injected within the method, and this injection will resolve classes that are called dependency injection.

32. How can you turn off the CSRF protection for a specific route in Laravel?

If you want to turn off the CSRF protection for a specific route in Laravel, you need to add the below lines in the `app/Http/Middleware/VerifyCsrfToken.php` file.

```
//add an array of Routes to skip CSRF check
private $exceptUrls = ['controller/route1', 'controller/route2'];
//modify this function
public function handle($request, Closure $next) {
    //add this condition foreach($this->exceptUrls as $route) {
    if ($request->is($route)) {
        return $next($request);
    }
    }
    return parent::handle($request, $next);
}
```

33. How do you update Laravel?

If you want to update the Laravel framework to the latest version, you need to open the composer.json file and make the required changes to the version of the Laravel framework to the latest one. Execute the below command for updating the laravel framework:

```
composer update
```

34. What are closures in Laravel?

Closures are anonymous methods in Laravel used as a callback function, and you can also use it as a parameter in a function.

You can easily pass parameters into closure by changing the closure function call in the handle() method. With the help of the closures, you can access variables present outside the scope of the variables.

Example

```
function handle(Closure $closure) {  
    $closure();  
}  
handle(function(){  
    echo "Hello";  
});
```

You can add the closure parameter to the handle() method, and now you can call the handle() method and pass a service as a parameter.

35. What is with() in Laravel?

In Laravel, the with() function is used to eager load. Rather than using two or more separate queries for fetching the data from a database, you can use the with() method after the first command. You will get a better user experience as you do not have to wait longer to fetch the data from the database.

36. What is soft delete in Laravel?

In Laravel, soft delete is a feature that helps soft delete models rather than actually deleting them from the database. If you wish to enable the soft deletes for a model, you need to mention the soft delete property in the model, as shown below:

use **Illuminate\Database\Eloquent\SoftDeletes;**

and you can use this

use **SoftDeletes;** in our model property.

After using the delete() query, the deleted_at timestamp is set on the record if the record is not removed from the database.

37. What is a repository pattern in Laravel?

With the help of the repository pattern, you can use an object without knowing how it existed. It acts as an abstraction for the data layer, meaning there is no need to know how data persisted. Thus, the business logic depends on the repository for getting the right data. In simple words, it is used for decoupling the data access layers and the business logic in the application.

38. What is the singleton design pattern in Laravel?

In Laravel, the singleton design pattern is one where the class presents a single instance of itself. With this, you can restrict the creation of an instance of a class to a single object. You can use it whenever a single instance of the class is required within the system. If you have implemented it properly, the first call will instantiate the object, and the remaining calls will be returned to the same instantiated object.

39. What are views in Laravel?

The views in Laravel consist of the HTML code needed for your application. Also, we can define a view in Laravel as a method, separating the controller logic and the domain logic from the presentation layer. The resources folder holds views and its path is resources/views.

For example:

```
<html>
<body>
  <h1>Best Interview Question</h1>
</body>
</html>
```

40. What is method spoofing in Laravel?

Normally, HTML forms do not support PUT, PATCH, or DELETE actions. So, if you want to call these actions from an HTML form, you need to define their routes by adding the hidden `_method` field to that form. Thus, the value you send with the `_method` field will be used as the HTTP request method, as shown below:

```
<form action="/foo/bar" method="POST">
<input type="hidden" name="_method" value="PUT">
<input type="hidden" name="_token" value="{{ csrf_token() }}">
</form>
```

For generating the `_method` input, you need to use the `@method` Blade Directive, like this:

```
<form action="/foo/bar" method="POST">
@method('PUT')
@csrf
</form>
```

This is called method spoofing in Laravel.

41. What is tinkering in Laravel?

In Laravel, tinkering is a powerful REPL tool used to interact with the Laravel application using the command line in an interactive shell. It comes with the release version of 5.4, extracted in a separate package.

For installing Tinker, run the following command:

```
composer require laravel/tinker
```

For executing Tinker, execute the following command:

```
php artisan tinker
```

42. How do you clear cache in Laravel?

To clear the cache in Laravel, run the following commands in the same order:

```
php artisan config:clear
php artisan cache:clear
composer dump-autoload
php artisan view:clear
php artisan route:clear
```

43. What is REPL in Laravel?

REPL stands for Read-Eval-Print-Loop. It is an interactive shell, accepting single user input, processing it, and returning the result to the client.

44. What is the use of the updateOrinsert() method in Laravel?

This method is used to update an existing record in a database if the condition is matched or created if there is no matching record. It will return the boolean value.

You can use the following syntax:

```
DB::table('blogs')->updateOrInsert([Conditions],[fields with value]);
```

45. How do you change the default database type in Laravel?

For this, you need to update the following in the config/database.php file. You can choose a MySQL database.

```
'default' => env('DB_CONNECTION', 'mysql')
```


46. How do you stop an Artisan server in Laravel?

You can stop an Artisan server in three steps, as follows:

- First, press Ctrl + Shift + ESC together. Locate the php system walking artisan and kill it with proper click -> kill process.
- Later, reopen the command line and start the server again
- You can kill the manner by sending a kill sign with Ctrl + C

47. How do you generate the application key in Laravel?

Run the following command to generate the application key in Laravel:

```
php artisan key:generate
```

48. How can you extend the login expiration time in Auth?

If you want to extend the login expiration time, make the required changes to the config/session.php file. You need to update the value of the variable "lifetime", also you can update the variable as per your requirement.

49. How do you roll back the last migration in Laravel?

You can run the following artisan command to roll back the last migration in Laravel:

```
php artisan migrate:rollback --step=1
```

50. How do you check the current route name?

You can use the following method to check the current route name:

```
request()->route()->getName()
```

Laravel is one of the most commonly used web frameworks among all PHP web developers. There is a moderate difference between Laravel version 7 and Laravel version 8; however, the other features are still the same.

These Laravel Interview Questions should prepare you well.

The Laravel framework helps developers create responsive and reliable web-based applications seamlessly with the help of features such as routing, controllers, middleware, views, blade templates, and eloquent models - among so much more.







If you aim to become a Laravel or PHP developer and appear for an interview, this article should help you prepare well. Good luck!

Interested in learning Laravel for beginners? Check this course!

From zero fly to build Laravel CMS! (https://click.linksynergy.com?id=jU79Zysihs4&offerid=1045023.2275171&type=2&murl=https%3A%2F%2Fwww.udelaravel-for-beginner-from-zero-fly-to-the-heaven%2F&u1=blog%2Ftop-laravel-intans_amcid-pOBZEoxQUMz7YPVmeVNBU)

 [Laravel \(https://hackr.io/blog/tag/laravel\)](https://hackr.io/blog/tag/laravel)

Share:


 (<https://twitter.com/intent/tweet?text=Top+50+Laravel+Interview+Questions+and+Answers+in+2022+https%3A%2F%2Fhackr.io%2Fblog%2Ftop-laravel-interview-questions-and-answers>)
  (<https://www.linkedin.com/shareArticle?mini=true&url=https://hackr.io/blog/top-laravel-interview-questions-and-answers>)
  (<http://www.reddit.com/submit?url=https://hackr.io/blog/top-laravel-interview-questions-and-answers>)
  (<https://news.ycombinator.com/submitlink?u=https://hackr.io/blog/top-laravel-interview-questions-and-answers>)
  (<https://api.whatsapp.com/send?text=https%3A%2F%2Fhackr.io%2Fblog%2Ftop-laravel-interview-questions-and-answers>)



Swapnil Banga

(<https://hackr.io/blog/author/swapnil-banga>)

Software engineer, hardware enthusiast, writer by avocation and a gamer. Swapnil has been working on Hackr for a large part of his career. Primarily working on Laravel, he is also the author of our React Native Android app. When not in front of a screen, you will find him devouring a novel or listening to heavy metal. View all posts by the Author (<https://hackr.io/blog/author/swapnil-banga>)

Related Posts

(<https://hackr.io/blog/laravel-cheat-sheet>)

Laravel Cheat Sheet: Download PDF For Quick Reference

(<https://hackr.io/blog/laravel-cheat-sheet>) **Read More** (<https://hackr.io/blog/laravel-cheat-sheet>)

(<https://hackr.io/blog/django-vs-laravel>)

Django Vs Laravel: Which framework to choose?

(<https://hackr.io/blog/django-vs-laravel>) [Read More](#) (<https://hackr.io/blog/django-vs-laravel>)

(<https://hackr.io/blog/best-data-engineering-courses>)

10 Best Data Engineering Courses to Start a New Career in 2022

(<https://hackr.io/blog/best-data-engineering-courses>) [Read More](#) (<https://hackr.io/blog/best-data-engineering-courses>)

Leave a comment

Email address*

Your email will not be published

Name*

Comment*

Related Tutorials



CakePHP

(<https://hackr.io/tutorials/learn-cakephp>)



CodeIgniter

(<https://hackr.io/tutorials/learn-codeigniter>)



Laravel

(<https://hackr.io/tutorials/learn-laravel>)



PHP

(<https://hackr.io/tutorials/learn-php>)



Symfony

(<https://hackr.io/tutorials/learn-symfony>)

Recommended Learning

Laravel 5 Fundamentals (laracasts.com)

(<https://hackr.io/tutorial/laravel-5-fundamentals>)

PHP with Laravel for Beginners (www.udemy.com)

(<https://hackr.io/tutorial/php-with-laravel-for-beginners>)

Laravel 5.4 From Scratch (laracasts.com)

(<https://hackr.io/tutorial/laravel-54-from-scratch>)

VIEW MORE (<https://hackr.io/tutorials/learn-laravel>)

[Blog \(https://hackr.io/blog\)](https://hackr.io/blog) [Roadmaps \(https://hackr.io/roadmaps\)](https://hackr.io/roadmaps) [About Us \(https://hackr.io/about\)](https://hackr.io/about)

[Programming Tips \(https://chrome.google.com/webstore/detail/programming-tips/ooaiehbfgncjjeaiedpffeajkeleikpl\)](https://chrome.google.com/webstore/detail/programming-tips/ooaiehbfgncjjeaiedpffeajkeleikpl) [Help & FAQ \(https://hackr.io/help\)](https://hackr.io/help)

We ❤️ [Feedback](#) [Advertise / Partner \(https://hackr.io/advertise-partner-with-us\)](https://hackr.io/advertise-partner-with-us) [Write for us \(/blog/write-for-us\)](/blog/write-for-us)

[Privacy Policy \(https://hackr.io/privacy-policy\)](https://hackr.io/privacy-policy)

(<https://play.google.com/store/apps/details?id=io.hackr.hackr&hl=en>)

Disclosure: This page may contain affiliate links, meaning when you click the links and make a purchase, we receive a commission.