

Php Laravel Questionnaire

- [PHP Laravel Questionnaire](#)

PHP Laravel Questionnaire

1). What is the latest version of Laravel?

Answer: Laravel 8 is the latest version.

2). What is the minimum compatible version of PHP for Laravel 7 and 8?

Answer: The minimum compatible PHP version for Laravel 7 is PHP 7.2.5 and for Laravel 8 is PHP 7.3.0

3) What are the popular features of Laravel?

Answer: There are several popular features in Laravel. These are enlisted below.

- Eloquent ORM
- Query builder
- Reverse routing
- Class auto-loading
- Restful controllers
- Blade template engine
- Lazy collection
- Unit testing
- Database seeding
- Migrations

4) What are the new features of Laravel 8?

Answer: Laravel 8 released on the 8th of September 2020 with new additional features and some modifications to the existing features.

The following list shows the new features of Laravel 8:

- Laravel Jetstream
- Models directory
- Model factory classes

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

5) Does Laravel support Bootstrap?

Answer: Yes, Laravel supports the Bootstrap CSS framework.

6) Name a few competitors of Laravel?

Answer: The following list shows the top competitors. They are all among the top 10 PHP frameworks in 2020.

- CodeIgniter
- Symfony
- Yii
- CakePHP
- Zend Framework
- Phalcon
- FuelPHP

7) What are the differences between Laravel and CodeIgniter frameworks?

Answer: There are several differences between Laravel and CodeIgniter frameworks, and some main differences are shown in the below table.

Laravel Framework	CodeIgniter Framework
Relational object-oriented	Object-oriented
Supports custom HTTPS routes	Does not support HTTPS routes fully
Has authentication class features	No built-in authentication features
Has an inbuilt unit testing feature	No inbuilt unit testing feature
Use blade templates	Does not use blade templates
Not easy to learn for beginners	Easy to learn for beginners

Easy to develop REST APIs	Not easy to develop REST APIs
Supports ORM	Does not support ORM

8) What is MVC architecture?

Answer: MVC architecture is a design pattern that is used to develop web applications. It consists of three components named Model, View and Controller. MVC design pattern also helps to speed up the development of the web application.

- Model: In MVC architecture, the letter M stands for Models. Model is the central component of the MVC design pattern. It manages the data in the application.
- View: In MVC architecture, the letter V stands for Views. A view displays data to the user.
- Controller: In MVC architecture, the letter C stands for Controllers. A controller is used to handle user requests.

9) What is the command you can use to check whether you have installed the composer on your computer?

Answer: You can run the following command in the command prompt to check whether you have successfully installed the composer on your computer.

10) What are the server requirements for Installing Laravel version 8?

Answer: Installing Laravel Homestead will full-fill server requirements for installing Laravel 8.

If you are not using Laravel Homestead, your server should meet the following requirements:

- PHP version 7.3 or above version
- PHP extensions
 - BCMath PHP Extension
 - CType PHP Extension
 - Fileinfo PHP extension

- JSON PHP Extension
- Mbstring PHP Extension
- OpenSSL PHP Extension
- PDO PHP Extension
- Tokenizer PHP Extension
- XML PHP Extension

11) Consider a situation where you have already installed Laravel 8 on your machine, and want to install a Laravel 7 project without uninstalling Laravel 8 from your machine. So, how are you going to install a Laravel 7 project?

Answer: It is simple. We can run the following command in the command prompt to install a Laravel 7 project.

```
composer create-project --prefer-dist laravel/laravel name_of_the_project "7.*"
```

12) What are bundles in Laravel?

Answer: Bundles are used to increase the functionality of Laravel. In Laravel, bundles are popularly known as packages. It contains configuration, routes, migrations, views, etc.

13) What are the two main routing files found in Laravel?

Answer: The two main routing files are,

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

14) What are the available router methods in Laravel? #

Answer: The following list shows the available router methods in Laravel:

- Route::get(\$uri, \$callback);
- Route::post(\$uri, \$callback);
- Route::put(\$uri, \$callback);
- Route::patch(\$uri, \$callback);
- Route::delete(\$uri, \$callback);
- Route::options(\$uri, \$callback);

15) How to create a route? Briefly describe with an example.

Answer: A route can be created by using controllers or by adding the code directly to the route.

The following example shows how to create a route by adding the code directly to the route.

Example: Replace the code in routes/web.php file by adding the following code segment.

```
<?php

use Illuminate\Support\Facades\Route;

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

return "Welcome!";

});
```

Then, run the project on the browser. You will see Welcome! as the output.

16) What is Middleware?

Answer: Middleware behaves like a bridge and a filtering mechanism between a request and a response.

17) What are the databases supported by the Laravel framework?

Answer: The following list below shows the supported databases:

- MySQL 5.6+
- PostgreSQL (Postgres) 9.4+
- SQLite 3.8.8+
- SQL Server 2017+

18) What are the aggregate methods provided by the query builder in Laravel?

Answer: The following list shows the aggregate methods provided by the query builder:

- count()
- max()
- min()
- avg()
- sum()

19) Rahul wrote the following validation rules for a file uploading field.

```
$request->validate(['file' => 'required|mimes:doc,pdf|max:2048']);
```

Briefly explain the above validation rules.

Answer: In the above validation, Rahul used three validation rules. They are,

1. required: The required validation rule prevents the user from submitting the form without uploading a file. In other words, the file field is mandatory.
2. mimes:doc,pdf: The mimes:doc,pdf validation rule only allows the user to upload a file which has .doc extension or .pdf extension.
3. max:2048: The max:2048 validation rule only allows the user to upload a file with a maximum size of 2048 bytes.

20) What is the purpose of a session in Laravel?

Answer: A session is used to store data and keeps track of users.

21) What is Laravel authentication?

Answer: Laravel authentication is the process of verifying application users. It can be achieved by identifying the user's username and password. Some other parameters may also use for authentication. If user credentials are valid, then the user is authenticated.

Laravel uses guards and providers for the authentication process. Guards define how users are authenticated for each request while providers define how users are retrieved from your persistent storage.

22) What is a CSRF token?

Answer: CSRF is an abbreviation for Cross-Site Request Forgery. A CSRF token is a unique value that is generated by the server-side of the application and sent to the client.

CSRF token helps to protect web applications from attacks which force a user to perform an unwanted action (commonly known as CSRF attacks).

The following code segment shows how a CSRF token can be used when creating a form in Laravel.

```
<form action="/user" method="POST">
```

```
@csrf
```

```
...
```

```
</form>
```

23) What is authorization?

Answer: Authorization is the process of verifying whether authenticated users have the required permission to access the requested resources. Laravel uses gates for the authorization process.

24) What are the common tools used to send emails in Laravel?

Answer: The following list below shows some common tools that can be used to send emails in Laravel.

- Mailtrap
- Mailgun

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

25) What is Laravel Forge? #

Answer: It is a server management tool for PHP applications. It is a great alternative if you are not planning to manage your own servers.

Note: Click [here](#) (the official page of Laravel Forge) to learn more about Laravel Forge.

26) What is Eloquent in Laravel?

Eloquent is the ORM used to interact with the database using Model classes. It gives handy methods on class objects to make a query on the database.

It can directly be used to retrieve data from any table and run any raw query. But in conjunction with Models, we can make use of its various methods and also make use of relationships and attributes defined on the model.

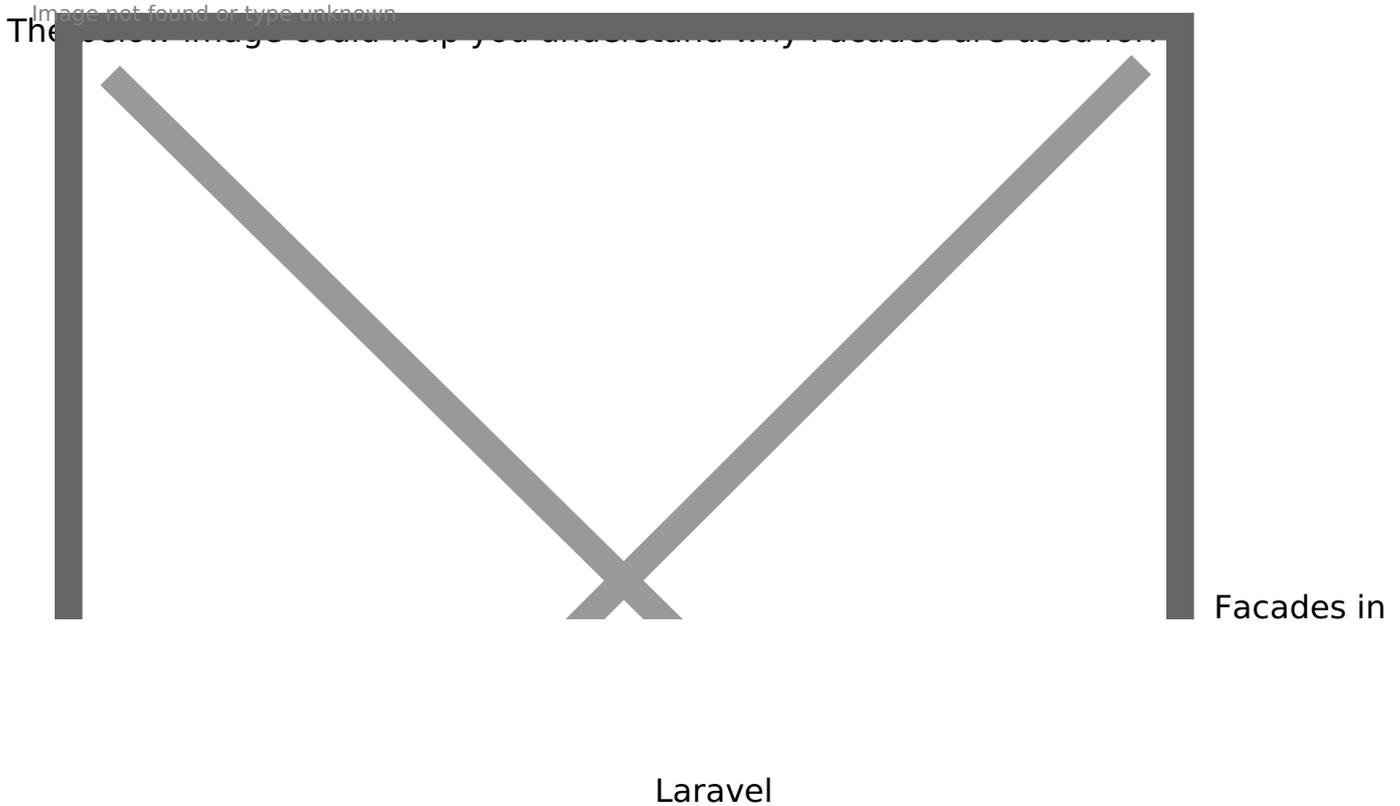
Some examples of using the Eloquent are below:

- ``DB::table('users')->get()``
- ``User::all()``
- ``User::where('name', '=', 'Eloquent')->get()``

27) What are facades?

Facades are a way to register your class and its methods in Laravel Container so they are available in your whole application after getting resolved by Reflection.

The main benefit of using facades is we don't have to remember long class names and also don't need to require those classes in any other class for using them. It also gives more testability to the application.



28) What is Localization in Laravel?

Localization is a way to serve content concerning the client's language preference. We can create different localization files and use a laravel helper method like this:

`__('auth.error')` to retrieve translation in the current locale. These localization files are located in the `resources/lang/[language]` folder.

29) What is a Service Provider?

A Service Provider is a way to bootstrap or register services, events, etc before booting the application. Laravel's own bootstrapping happens using Service Providers as well. Additionally, registers service container bindings, event listeners, middlewares, and

even routes using its service providers.

If we are creating our application, we can register our facades in provider classes.

30) What is the register and boot method in the Service Provider class?

The register method in the Service Provider class is used to bind classes or services to the Service Container. It should not be used to access any other functionality or classes from the application as the service you are accessing may not have loaded yet into the container.

The boot method runs after all the dependencies have been included in the container and now we can access any functionality in the boot method. Like you can create routes, create a view composer, etc in the boot method.

31) What are named routes?

A named route is a route definition with the name assigned to it. We can then use that name to call the route anywhere else in the application.

```
Route::get('/hello', 'HomeController@index')->name('index');
```

This can be accessed in a controller using the following:

```
return redirect()->route('index');
```

32) What do you know about CSRF token in Laravel? How can someone turn off CSRF protection for a specific route?

CSRF protection stands for Cross-Site Request Forgery protection. CSRF detects unauthorized attacks on web applications by the unauthorized users of a system. The

built-in CSRF plug-in is used to create CSRF tokens so that it can verify all the operations and requests sent by an active authenticated user.

To turn off CSRF protection for a specific route, we can add that specific URL or Route in `$except` variable which is present in the `app\Http\Middleware\VerifyCsrfToken.php` file.

Example

1. `class VerifyCsrfToken extends BaseVerifier`
2. `{`
3. `protected $except = [`
4. `'Pass here your URL',`
5. `];`
6. `}`

33) How will you explain dd() function in Laravel?

`dd` stands for "Dump and Die." Laravel's `dd()` function can be defined as a helper function, which is used to dump a variable's contents to the browser and prevent the further script execution.

Example

1. `dd($array);`

34) How will you describe Fillable Attribute in a Laravel model?

In eloquent ORM, `$fillable` attribute is an array containing all those fields of table which can be filled using mass-assignment.

Mass assignment refers to sending an array to the model to directly create a new record in Database.

Code Source

1. class User extends Model {
2. protected \$fillable = ['name', 'email', 'mobile'];
3. // All fields inside \$fillable array can be mass-assigned
4. }

35) How will you explain Guarded Attribute in a Laravel model?

The guarded attribute is the opposite of fillable attributes.

In Laravel, fillable attributes are used to specify those fields which are to be mass assigned. Guarded attributes are used to specify those fields which are not mass assignable.

Code Source

1. class User extends Model {
2. protected \$guarded = ['role'];
3. // All fields inside the \$guarded array are not mass-assignable
4. }

If we want to block all the fields from being mass-assigned, we can use:

1. protected \$guarded = ['*'];

\$fillable serves as a "white list" whereas \$guarded functions serves like a "black list".

One should use either \$fillable or \$guarded.

36) What does PEAR stand for?

PEAR means “PHP Extension and Application Repository”. It extends PHP and provides a higher level of programming for web developers.

37) What is the correct and the most common way to start and finish a PHP block of code?

The two most common ways to start and finish a PHP script are:

`<?php [--- PHP code----] ?>` and `<? [--- PHP code ---] ?>`

38) What type of operation is needed when passing values through a form or an URL?

If we would like to pass values through a form or an URL, then we need to encode and to decode them using `htmlspecialchars()` and `urlencode()`.

39) What is the use of the function ‘`imagetypes()`’?

`imagetypes()` gives the image format and types supported by the current version of GD-PHP.

40) What is the main difference between `require()` and `require_once()`?

`require()`, and `require_once()` perform the same task except that the second function checks if the PHP script is already included or not before executing it.

(same for `include_once()` and `include()`)

41) How can we connect to a MySQL database from a PHP script?

To be able to connect to a MySQL database, we must use `mysqli_connect()` function as follows:

```
<!--?php $database = mysqli_connect("HOST", "USER_NAME", "PASSWORD");  
mysqli_select_db($database,"DATABASE_NAME"); ?-->
```