Laravel Interview Questions and Answers (8+ Years Experience)

1. What are Laravel service providers?

Service providers are the central place for configuring your application. They bind services into the service container and are used to boot any application services.

2. Explain Laravels service container.

The service container is a powerful tool for managing class dependencies and performing dependency injection. Its used to bind interfaces to concrete classes.

3. What is a facade in Laravel?

Facades provide a static interface to classes that are available in the service container. They are syntactic sugar over the service container bindings.

4. What are Laravel contracts?

Contracts are a set of interfaces that define the core services provided by Laravel. They allow for flexibility and are useful for swapping implementations.

5. How does middleware work in Laravel?

Middleware filters HTTP requests entering the application. It can be used for authentication, logging, CORS, etc.

6. How can you optimize Laravel application performance?

Use route caching, config caching, eager loading, minimize DB queries, use queues, implement Redis or Memcached.

7. What are Laravel queues?

Queues allow deferred execution of time-consuming tasks like email sending or video processing. Laravel supports Redis, SQS, Database, etc.

8. What is the difference between Auth::attempt() and Auth::login()?

Auth::attempt() checks credentials and logs in the user. Auth::login() logs in the user without validating credentials.

9. What is a repository pattern and why use it in Laravel?

It abstracts the data layer, separating business logic from data access logic, making the code more testable and maintainable.

10. What is Laravel Passport?

Laravel Passport is an OAuth2 server implementation for API authentication using access tokens.

11. How do you handle API rate limiting in Laravel?

Use Laravel's built-in rate limiting via middleware like throttle:60,1 or define custom limits using the RateLimiter facade.

12. Explain Laravel event and listener system.

Events are used to trigger actions and listeners respond to them. It promotes decoupling and clean code.

13. What is the use of policy and gate in Laravel?

Policies are classes that organize authorization logic around models. Gates provide simple closure-based authorization.

14. What are jobs in Laravel?

Jobs are used to represent a unit of work in the application, which can be dispatched immediately or queued for later processing.

15. How can you use dependency injection in controllers?

By type-hinting class dependencies in the controller constructor or method, Laravel automatically resolves them from the container.

16. How do you implement a custom Artisan command?

Run php artisan make:command CommandName and define your logic in the handle() method.

17. What is the difference between hasOneThrough and hasManyThrough relationships?

hasOneThrough defines a one-to-one relationship via an intermediate model. hasManyThrough defines a one-to-many relationship through a third model.

18. How does Laravels scheduler work?

The scheduler allows you to fluently and expressively define command-based tasks in app/Console/Kernel.php.

19. What are Laravel observers?

Observers are classes that listen for model events like created, updated, deleted, etc.

20. How do you use form requests for validation?

Create a custom request using php artisan make:request. Define rules in the rules() method and authorize in authorize().

21. What is the difference between update() and save() in Eloquent?

update() performs a direct DB update. save() updates an existing model instance or creates a new one.

22. What is Laravel Mix?

Laravel Mix is a wrapper around Webpack that provides a clean API for defining asset build steps like compiling CSS and JS.

23. How do you implement multi-auth in Laravel?

Define guards in config/auth.php and use middleware to handle access per user type.

24. How does Route::resource work?

It defines a conventional set of routes for a resource controller (index, create, store, show, edit, update, destroy).

25. How can you prevent N+1 query problems in Laravel?

Use eager loading via with() or load() to preload related data.

26. Explain Laravels broadcasting.

Broadcasting allows real-time data broadcasting using drivers like Pusher, Redis, or Ably via WebSockets.

27. How do you handle file uploads in Laravel?

Use request()->file('input')->store('path'). Validate with mimes, max etc.

28. How do you implement localization?

Store translations in resources/lang and use __('key') or @lang('key') for string output.

29. How do you handle database transactions in Laravel?

Use DB::transaction(function() { ... }); to ensure atomicity of DB operations.

30. What is the difference between first(), get() and pluck() in Eloquent?

first() retrieves the first matching record. get() retrieves all matching records as a collection. pluck() retrieves a single column's values.