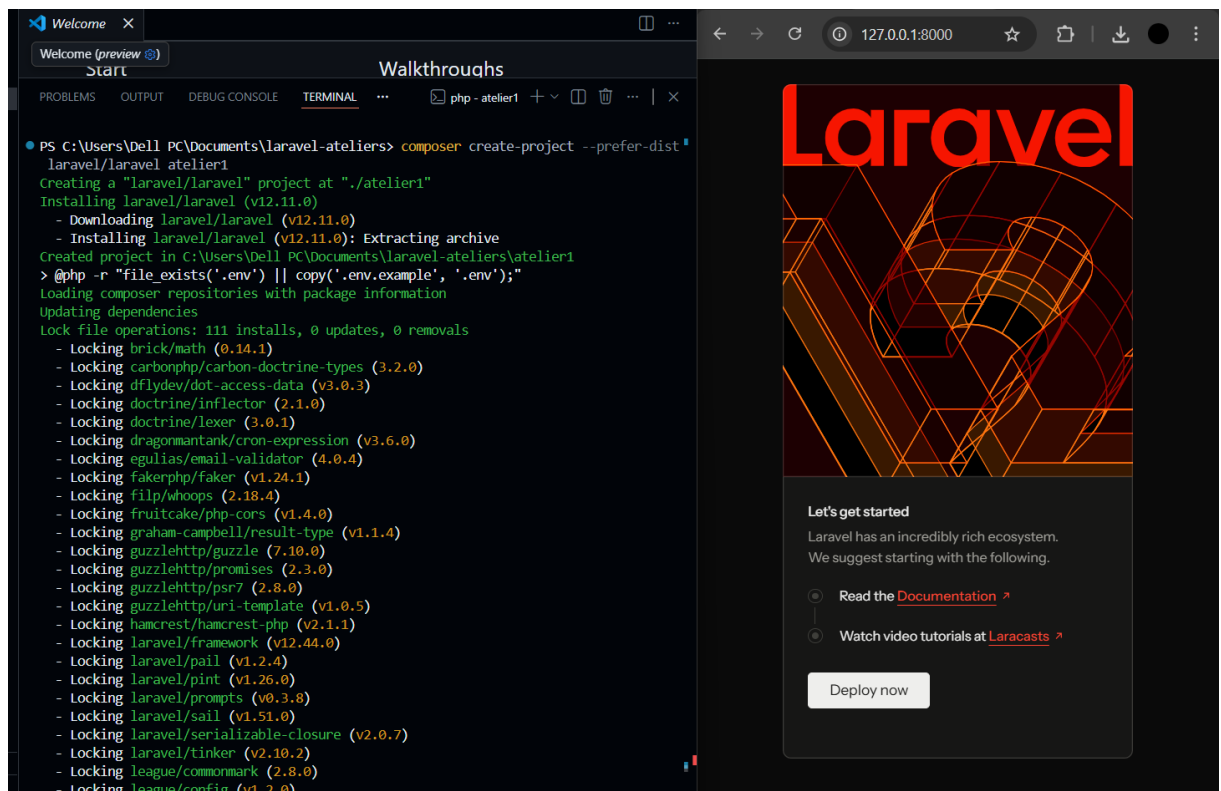


Atelier 1 Laravel: Introduction à Laravel

1. Créer un projet Laravel

2. Exécutez la commande suivante pour créer un projet Laravel (remplacez NomDuProjet par le nom que vous souhaitez donner à votre projet) :



The screenshot shows a VS Code terminal window on the left and a web browser on the right. The terminal displays the command `composer create-project --prefer-dist laravel/laravel atelier1` and its output, which includes creating the project, installing Laravel v12.11.0, and locking various dependencies. The browser on the right shows the Laravel website with the heading "Let's get started" and links to the documentation and Laracasts.

```
PS C:\Users\Dell PC\Documents\laravel-ateliers> composer create-project --prefer-dist laravel/laravel atelier1
Creating a "laravel/laravel" project at "./atelier1"
Installing laravel/laravel (v12.11.0)
- Downloading laravel/laravel (v12.11.0)
- Installing laravel/laravel (v12.11.0): Extracting archive
Created project in C:\Users\Dell PC\Documents\laravel-ateliers\atelier1
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies
Lock file operations: 111 installs, 0 updates, 0 removals
- Locking brick/math (0.14.1)
- Locking carbonphp/carbon-doctrine-types (3.2.0)
- Locking dflydev/dot-access-data (v3.0.3)
- Locking doctrine/inflector (2.1.0)
- Locking doctrine/lexer (3.0.1)
- Locking dragonmantank/cron-expression (v3.6.0)
- Locking egulias/email-validator (4.0.4)
- Locking fakerphp/faker (v1.24.1)
- Locking filp/whoops (2.18.4)
- Locking fruitcake/php-cors (v1.4.0)
- Locking graham-campbell/result-type (v1.1.4)
- Locking guzzlehttp/guzzle (7.10.0)
- Locking guzzlehttp/promises (2.3.0)
- Locking guzzlehttp/psr7 (2.8.0)
- Locking guzzlehttp/uri-template (v1.0.5)
- Locking hamcrest/hamcrest-php (v2.1.1)
- Locking laravel/framework (v12.44.0)
- Locking laravel/pail (v1.2.4)
- Locking laravel/pint (v1.26.0)
- Locking laravel/prompts (v0.3.8)
- Locking laravel/sail (v1.51.0)
- Locking laravel/serializable-closure (v2.0.7)
- Locking laravel/tinker (v2.10.2)
- Locking league/commonmark (2.8.0)
- Locking league/config (v1.2.0)
```

4. Démarrez le serveur de développement Laravel :



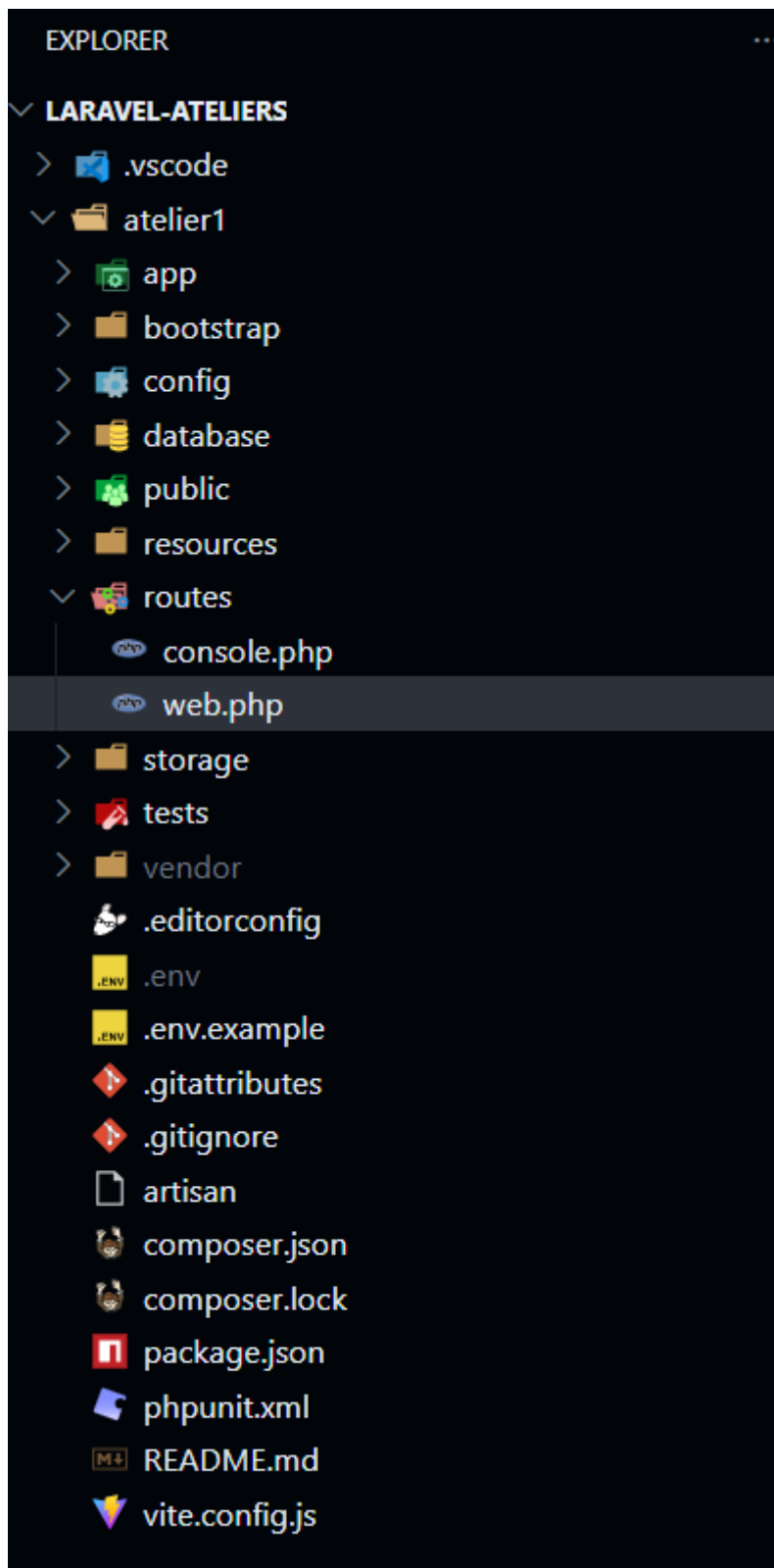
The screenshot shows a terminal window with the command `php artisan serve` executed. The output shows the server running on `http://127.0.0.1:8000`. A message at the bottom indicates that pressing `Ctrl+C` will stop the server.

```
PS C:\Users\Dell PC\Documents\laravel-ateliers\atelier1> php artisan serve

INFO  Server running on [http://127.0.0.1:8000].
INFO  Server running on [http://127.0.0.1:8000].

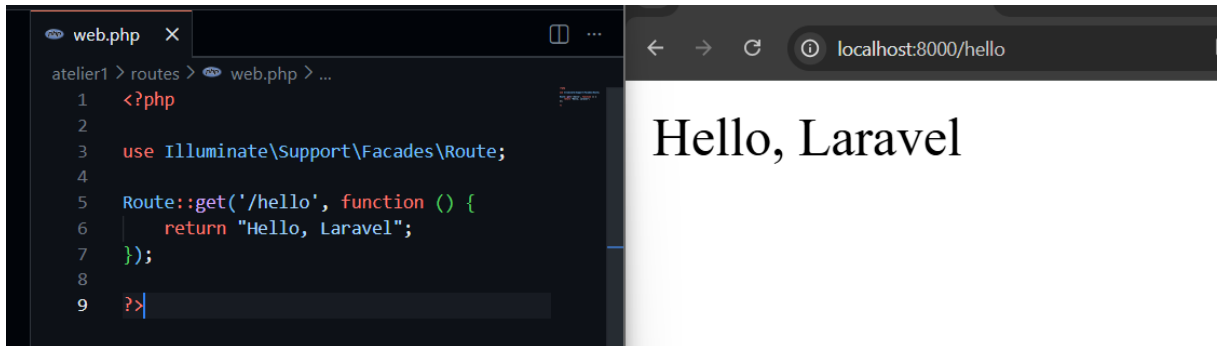
Press Ctrl+C to stop the server
```

2. Architecture du projet Laravel et exploration des dossiers

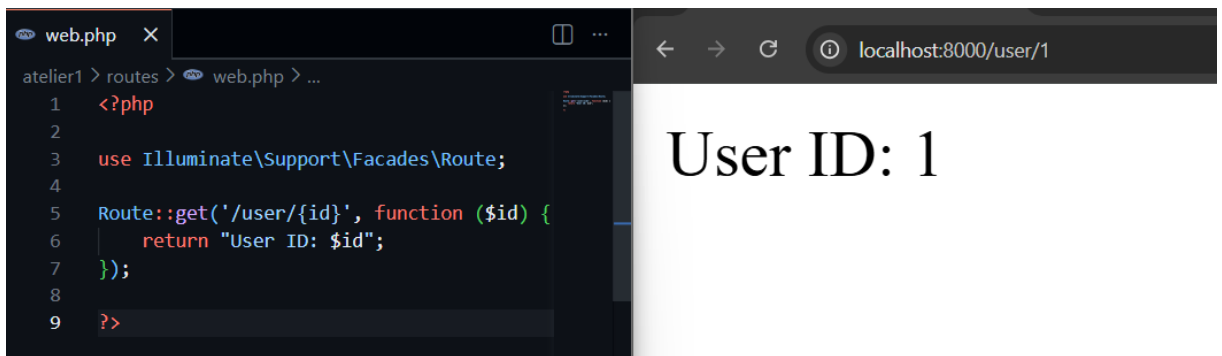


3. Créer et tester des routes dans Laravel

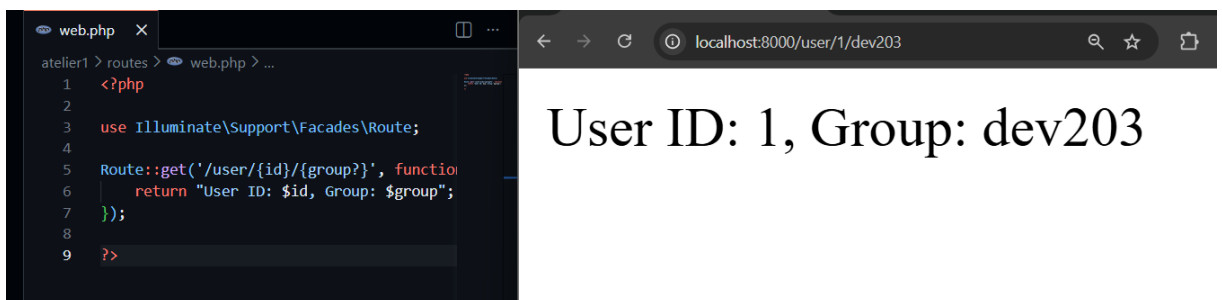
3.1 Routes de base



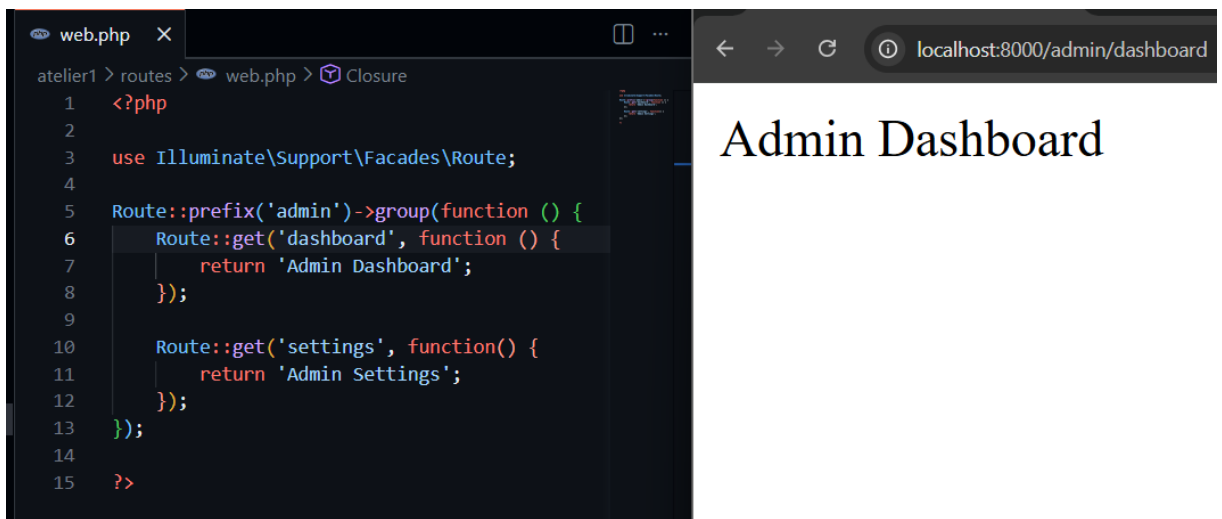
3.2 Route dynamique avec des paramètres



3.3 Route avec un paramètre optionnel

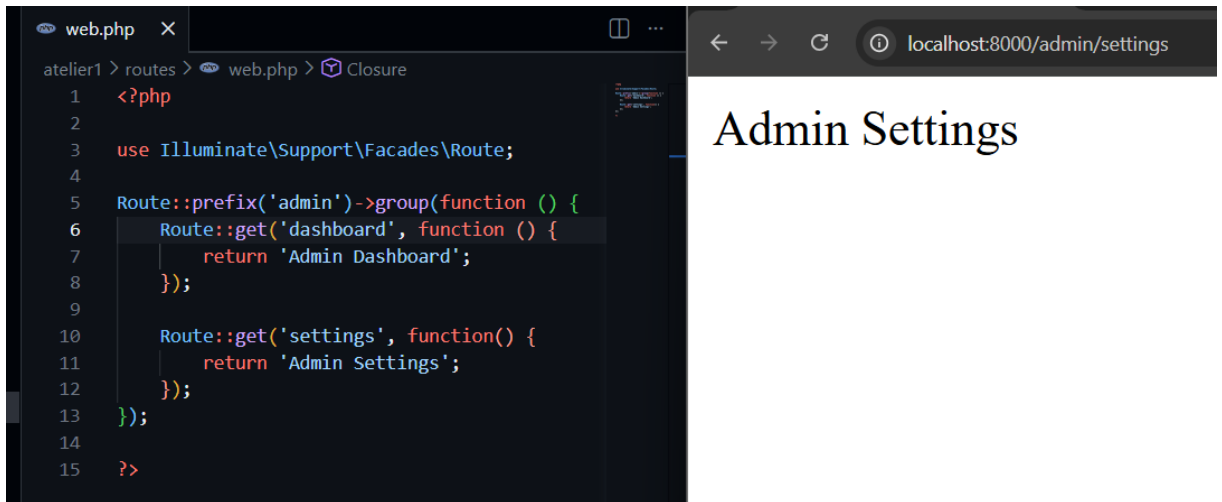


3.4 Route préfixée



The screenshot shows a web browser at `localhost:8000/admin/dashboard` displaying the text "Admin Dashboard". To the left, a code editor shows the following PHP code:

```
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4
5 Route::prefix('admin')->group(function () {
6     Route::get('dashboard', function () {
7         return 'Admin Dashboard';
8     });
9
10    Route::get('settings', function() {
11        return 'Admin Settings';
12    });
13 });
14
15 ?>
```

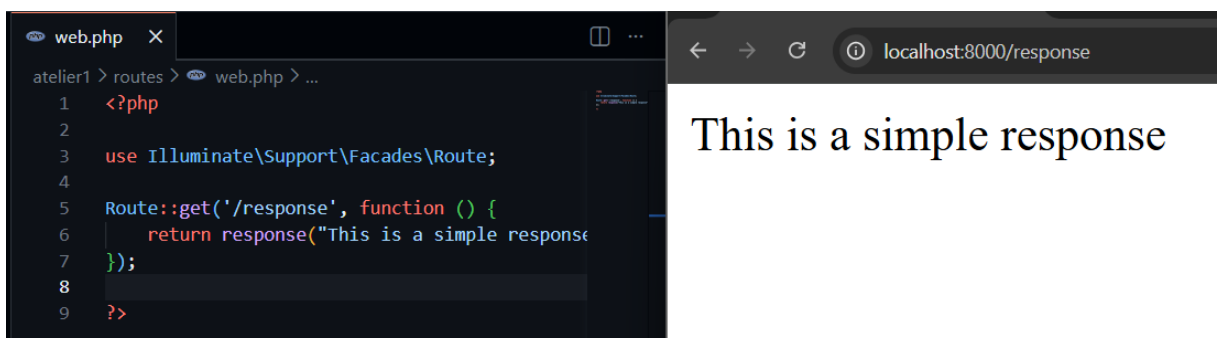


The screenshot shows a web browser at `localhost:8000/admin/settings` displaying the text "Admin Settings". To the left, a code editor shows the same PHP code as the previous block:

```
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4
5 Route::prefix('admin')->group(function () {
6     Route::get('dashboard', function () {
7         return 'Admin Dashboard';
8     });
9
10    Route::get('settings', function() {
11        return 'Admin Settings';
12    });
13 });
14
15 ?>
```

4. Gérer les réponses et redirections

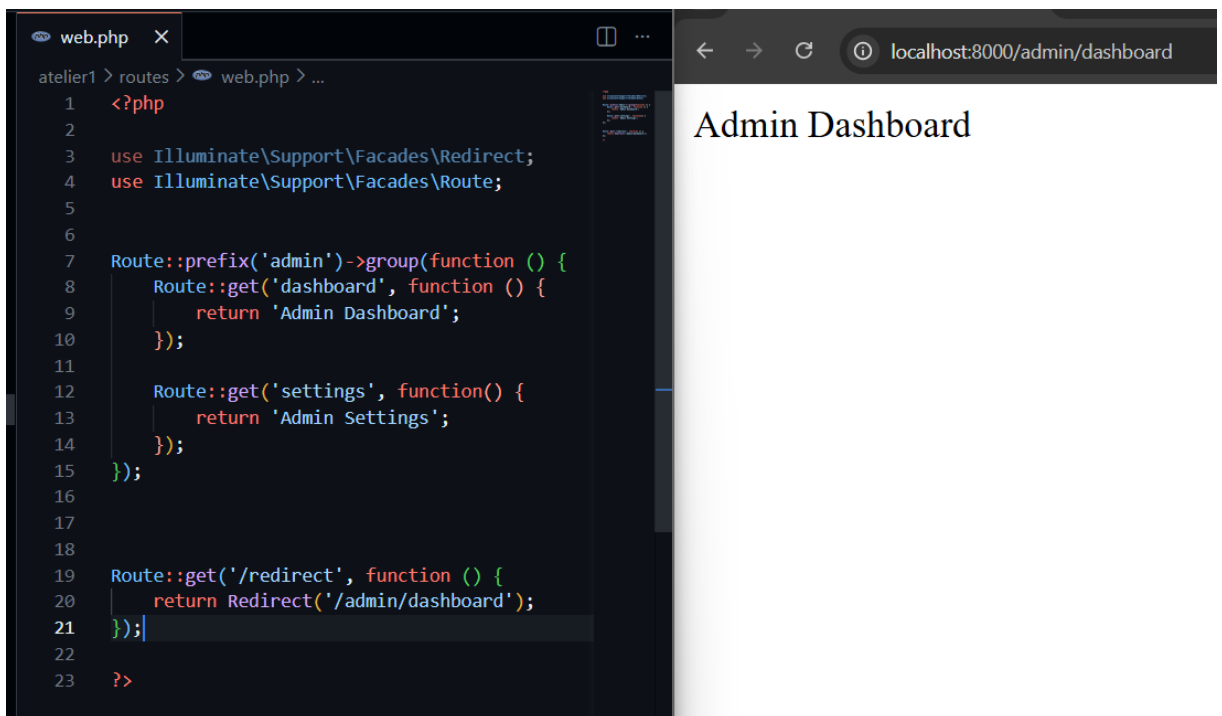
4.1 Retourner une réponse simple



The screenshot shows a web browser at `localhost:8000/response` displaying the text "This is a simple response". To the left, a code editor shows the following PHP code:

```
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4
5 Route::get('/response', function () {
6     return response("This is a simple response");
7 });
8
9 ?>
```

4.2 Redirection vers une autre route

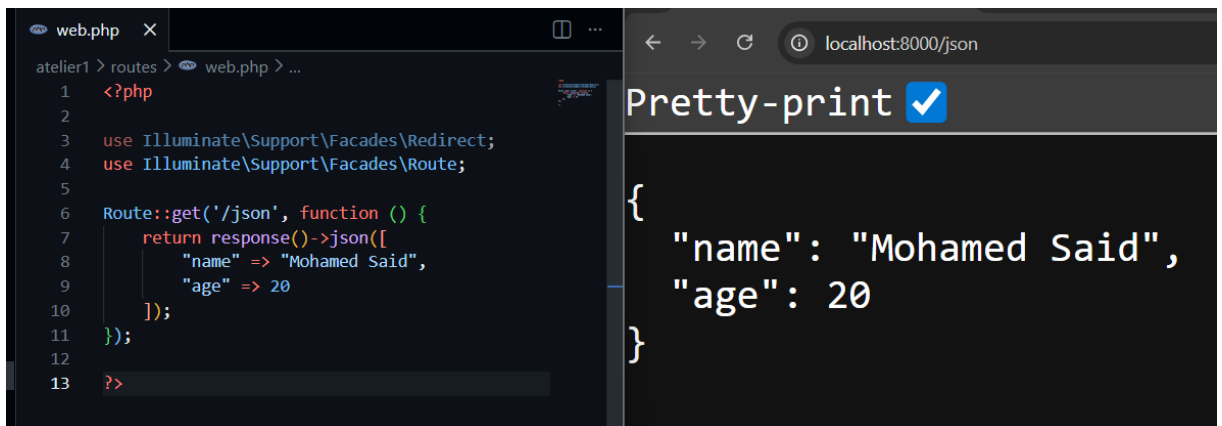


The screenshot shows a code editor on the left and a web browser on the right. The code editor displays a PHP file named `web.php` with the following content:

```
1 <?php
2
3 use Illuminate\Support\Facades\Redirect;
4 use Illuminate\Support\Facades\Route;
5
6
7 Route::prefix('admin')->group(function () {
8     Route::get('dashboard', function () {
9         return 'Admin Dashboard';
10    });
11
12    Route::get('settings', function() {
13        return 'Admin Settings';
14    });
15 });
16
17
18
19 Route::get('/redirect', function () {
20     return Redirect('/admin/dashboard');
21 });
22
23 ?>
```

The web browser on the right shows the URL `localhost:8000/admin/dashboard` and the page content "Admin Dashboard".

4.3 Retourner une réponse JSON



The screenshot shows a code editor on the left and a web browser on the right. The code editor displays a PHP file named `web.php` with the following content:

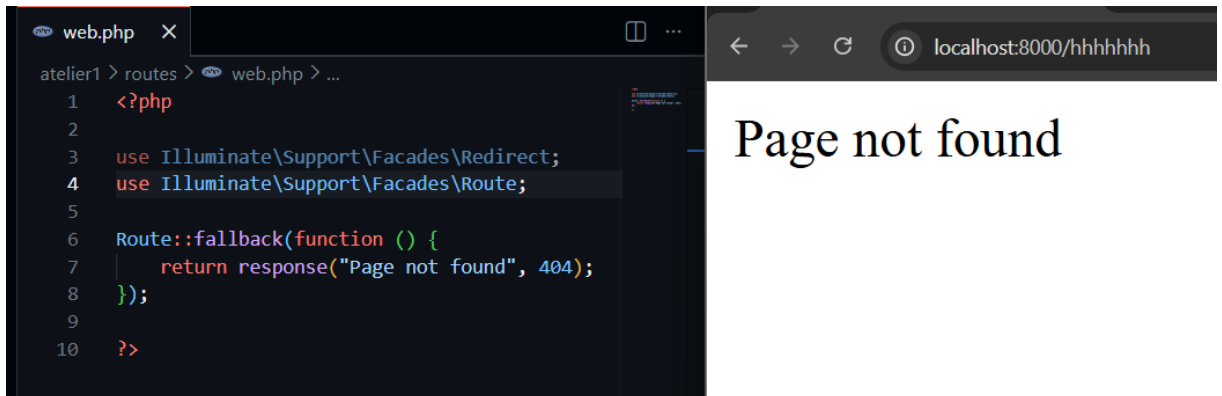
```
1 <?php
2
3 use Illuminate\Support\Facades\Redirect;
4 use Illuminate\Support\Facades\Route;
5
6 Route::get('/json', function () {
7     return response()->json([
8         "name" => "Mohamed Said",
9         "age" => 20
10    ]);
11 });
12
13 ?>
```

The web browser on the right shows the URL `localhost:8000/json` and the page content "Pretty-print" with a checkmark icon. Below this, the JSON response is displayed in a dark theme:

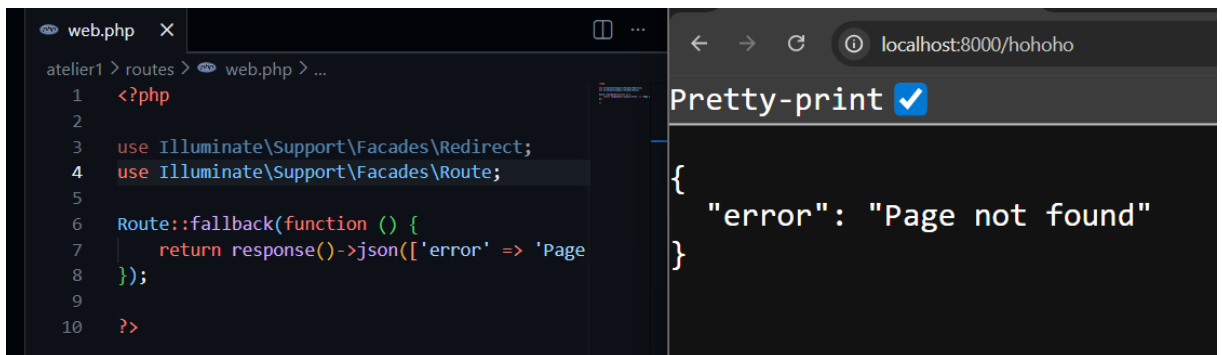
```
{
  "name": "Mohamed Said",
  "age": 20
}
```

5. Gérer les erreurs 404

5.1 Route Fallback

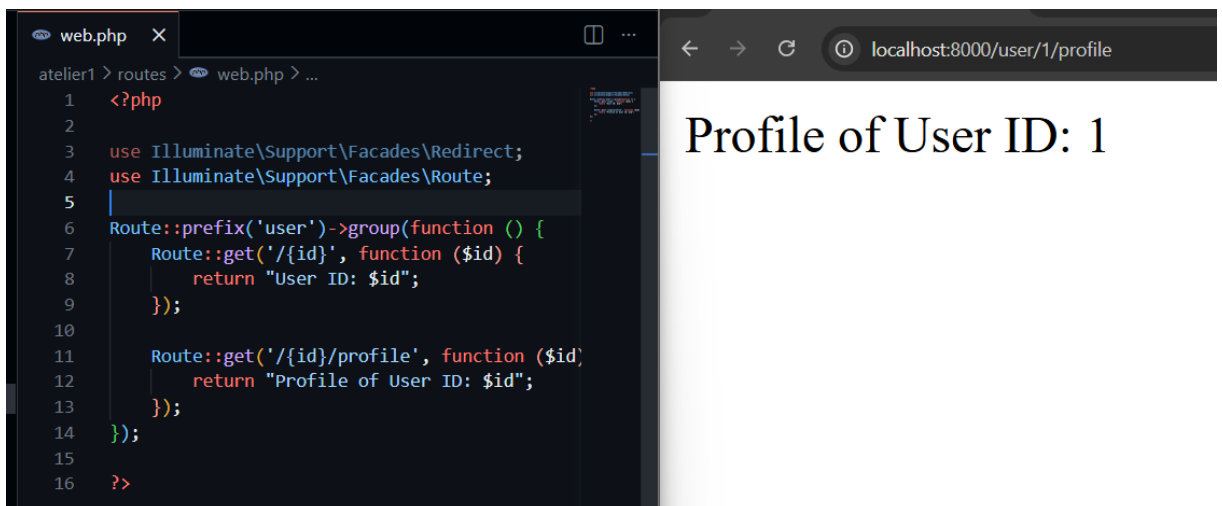


5.2 Personnalisation des erreurs 404



6. Bonnes pratiques en matière de routing

6.1 Utilisation des groupes de routes



6.2 Routes avec des paramètres optionnels



The screenshot shows a web browser window with the address bar displaying `localhost:8000/user/1`. To the left, a code editor shows the following PHP code:

```
1 <?php
2
3 use Illuminate\Support\Facades\Redirect;
4 use Illuminate\Support\Facades\Route;
5
6 Route::get('/user/{id}/{group?}', function ($id, $group = 'default') {
7     return "User ID: $id, Group: $group";
8 });
9
10 ?>
```

The browser displays the output of the route: **User ID: 1, Group: dev203**.

6.3 Routes avec des conditions



The screenshot shows a web browser window with the address bar displaying `localhost:8000/user/11`. To the left, a code editor shows the following PHP code:

```
1 <?php
2
3 use Illuminate\Support\Facades\Redirect;
4 use Illuminate\Support\Facades\Route;
5
6 Route::get('/user/{id}', function ($id) {
7     return "User ID: $id";
8 }->where('id', '[0-9]+'))
9
10 ?>
```

The browser displays the output of the route: **User ID: 11**.