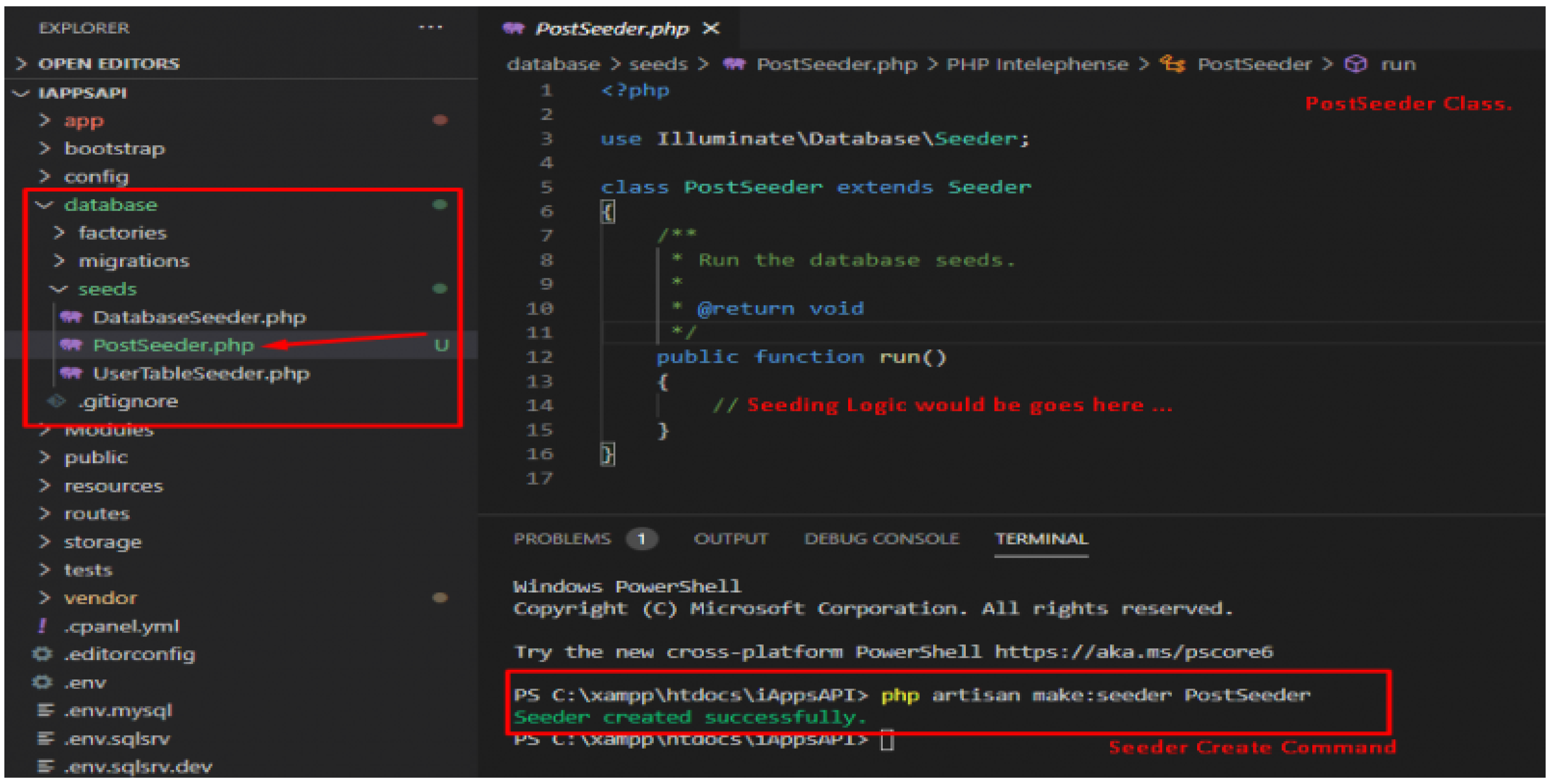
DEVS



```php
<?php

use Illuminate\Database\Seeder;

class PostSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        // Seeding Logic would be goes here ...
    }
}
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\xampp\htdocs\iAppsAPI> php artisan make:seeder PostSeeder
Seeder created successfully.
PS C:\xampp\htdocs\iAppsAPI>
```

Categories - Laravel PHP Framework Tags - Laravel Maniruzzaman Akash 2 years ago 4383 3 minutes 0

What is Seeder in Laravel and Why we need it?

What is Seeder in Laravel

Seeder or called Database Seeder is a feature of Laravel Php framework, which is used to auto generate data when an initial version is instantiated.

Why Use Seeder in Laravel

We use a seeder to test our laravel project quickly and efficiently. Like in a scenerio in a Laravel project, where we have to make `migrate:refresh` command which drops the old migrations and write new migrations in database.

So, what If our projects has 25 blog posts by which our site is visible fully. Ok, now if make that command, datbase actually clear, right? So, will we put 25 posts again in database and check if everythings right.

Or in another example would be we had two test users and check site by that's user ID, now create again that two test users in database and check project ?

Ok, so think once that how much painful would be that for you. Actually you may doing this currently right or not?

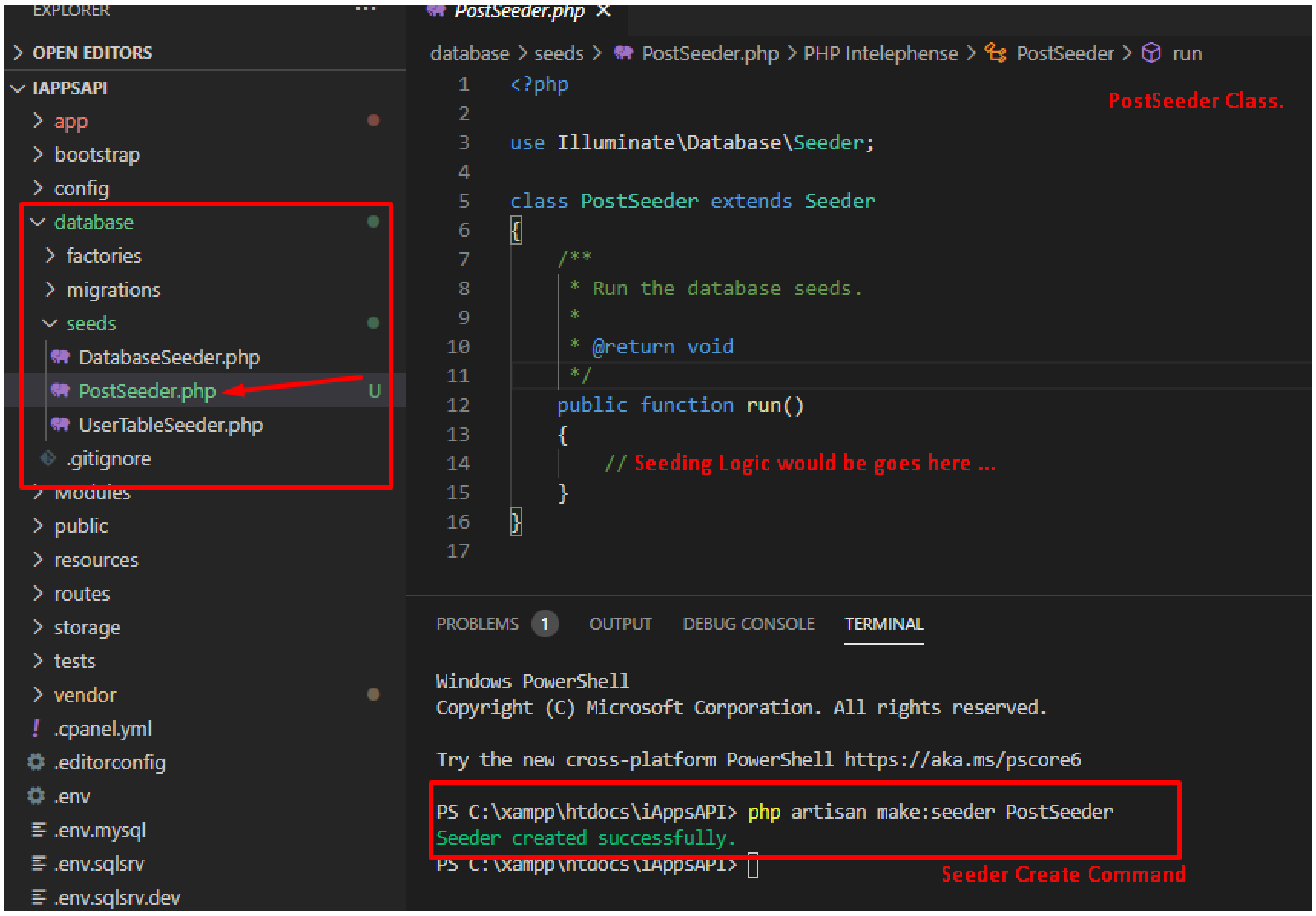
So, we'll make this process simple using the Laravel Database Seeder.

How to Create Seeder in Laravel

Ok, let's start to create a seeder in Laravel.

By default Laravel comes with a Seeder pattern in database/seeder folder. To Create a Seeder in that folder, we can use Laravel Seeder Command -

```
php artisan make:seeder PostSeeder
```



PostSeeder Class.

```php
<?php

use Illuminate\Database\Seeder;

class PostSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        // Seeding Logic would be goes here ...
    }
}

```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\xampp\htdocs\iAppsAPI> php artisan make:seeder PostSeeder
Seeder created successfully.
PS C:\xampp\htdocs\iAppsAPI>
```

Seeder Create Command

Assume, We want to create 10 dummy post in `PostSeeder` .

Steps to make dummu posts using Laravel Database Seeder.

Step 1:

Create a database migration for Post Model. We'll create a post model and with a migration

```
php artisan make:model Post -m
```

So, put the below migrations in `2020_07_20_133415_create_posts_table.php`

```
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreatePostsTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('posts', function (Blueprint $table) {
            $table->id();
            $table->string('title');
            $table->string('slug')->index();
            $table->text('description')->nullable();
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('posts');
    }
}
```

We've just created a post model with title, slug and description.

Step 2:

Now add fake 1000 posts in PostSeeder using also [Laravel Faker](). So, `PostSeeder.php` file would be,

```
use App\Post;
use Illuminate\Database\Seeder;
use Faker\Factory as Faker;

class PostSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        $faker = Faker::create();
        for ($i = 0; $i < 100; $i++) {
            Post::create([
                'title' => $faker->jobTitle,
                'slug' => $faker->unique()->slug(),
                'description' => $faker->text()
            ]);
        }
    }
}
```

So, in our `run()` ,

1. we've first instantiated the Faker instance.
2. Then loop throw 100 to generate 100 posts.
3. Inside loop, we've create Post, using Laravel `create()` function.
4. We've used th title, slug, description data of the faker class data. You could also use as your purpose data here.

Step 3:

Add this PostSeeder class in `DatabaseSeeder.php`

```
<?php

use Illuminate\Database\Seeder;

class DatabaseSeeder extends Seeder
{
    /**
     * Seed the application's database.
     *
     * @return void
     */
    public function run()
    {
        $this->call(PostSeeder::class);
    }
}
```

```
php artisan db:seed
```

Any Problem ?

Probably there could be need to autoload seeder class and run the following commands -

```
composer dump-autoload

php artisan optimize

php artisan db:seed
```

So, after this in your database there would be 100 posts stored and boom. Your site has now a full visibility and load testing also..

Step 5: More

You could also run a specific seeder like this,, we'll only seed the PostSeeder.

```
php artisan db:seed --class=PostSeeder
```

To refresh migrations and run seeder, there is also a command like this -

```
php artisan migrate:fresh --seed
```

To forcely Seed of the seeder -

```
php artisan db:seed --force
```

That is a basic and real life example of Laravel Database Seeder.

Official DOcumentation of Laravel about Database Seeder is here - https://laravel.com/docs/7.x/seeding#introduction

← Previous

PHP If-else-elseif and Switch-case

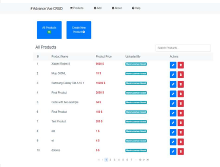
Next →

PHP String Functions - All necessary String functions in PHP to manage strings better.

Add a comment...

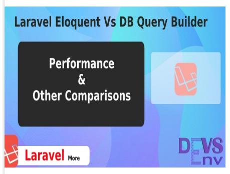
Facebook Comments plugin

Popular Tutorials



[Vue JS 3 - Complete Advance CRUD Example with Vuex, Vue-Validate, Pagination, Searching and Everything](#)

2 years ago


Laravel Eloquent Vs DB Query Builder
Performance & Other Comparisons
Laravel
DEVS Env

[Laravel Eloquent Vs DB Query Builder [Performance and other statistics]](#)

2 years ago

[Laravel Role Permission Management System Full Example with Source Code](#)

2 years ago

Laravel Workshop
Bangla
Day1
Laravel Membership Management Project
Installation & Setup & Basic File Structure
Laravel 8.x

[Laravel Membership Management Full Project Demo with Source Codes](#)

2 years ago

[Laravel Ecommerce Complete Web Application with Admin Panel](#)

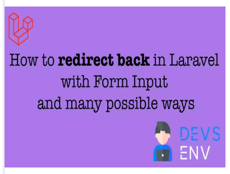
2 years ago

[Connect your React app with WebSocket to get real time Data](#)

2 years ago

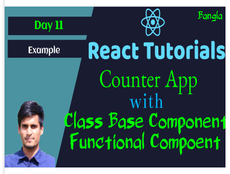
[#3 Basic about Laravel Routing](#)

2 years ago


How to redirect back in Laravel with Form Input and many possible ways
DEVS ENV

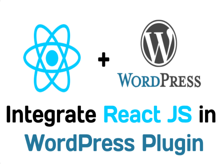
[How to redirect back in Laravel with Form Input and many possible ways](#)

1 year ago



[#11 - Simple Counter App in React JS - with Functional and Class Based Component](#)

2 years ago



[Start WordPress Plugin Development with React JS easily in just few steps](#)

10 months ago

Categories

Bash Scripting	1

C Programming	8
C#	0
ChatGPT	1
Code Editor	2
CSS	2
Design Pattern in PHP	2
E-Book	1
Git Commands	1
HTML	3
Interview Prepration	2
Java Programming	0
JavaScript	11
Laravel PHP Framework	36
Mysql	1
Node JS	1
Online Business	0
PHP	28
Programming	7
Python	1
React Js	19
React Native	1
Redux	2
Tailwind CSS	1
Uncategorized	0
Vue JS	1

Woocommerce	1
WordPress Development	2

Tags

Bash Scripting Business C C-sharp programming C++ Code Editor CSS Database Design pattern Express JS git Git Commands github HTML Java JavaScript Laravel Mathematics MongoDB Mysql Node JS PHP Programming Python React Js Redux Vue JS Windows terminal Woocommerce WordPress WordPress Plugin Development

Quick links

Home

Our Portfolios

Blog

Tutorials

Contact Us

Contribute Guest Article

Services

Android App Developement

Website Design & Developement

Desktop Software Developement

Search Engine Optimization

Advanced HTML Meta Tag Generator

Newsletter

Stay update with our latest tutorials

Enter Email Address →