

ACTOR SEEDER

```
1 <?php
2
3 use Illuminate\Database\Seeder;
4 use Faker\Factory as Faker;
5 use App\Actor;
6
7 class ActorTableSeeder extends Seeder
8 {
9     /**
10      * Run the database seeds.
11      *
12      * @return void
13      */
14     public function run()
15     {
16         $faker = Faker::create();
17         $faker->addProvider(new \Xylis\FakerCinema\Provider\Person($faker));
18         $actor=$faker->actors($gender = null, $count = 100, $duplicates = false);
19         foreach($actor as $index){
20             Actor::create([
21                 'name' => $index,
22                 'note' => $faker->word(),
23             ]);
24         }
25     }
26 }
27
```

FILM SEEDER

```
1 |<?php
2
3 use Illuminate\Database\Seeder;
4 use Faker\Factory as Faker;
5 use App\Film;
6
7 class FilmTableSeeder extends Seeder
8 {
9     /**
10      * Run the database seeds.
11      *
12      * @return void
13      */
14     public function run()
15     {
16         $faker = Faker::create();
17         $faker->addProvider(new \Xylis\FakerCinema\Provider\Movie($faker));
18         $title=$faker->movies(100, $duplicates=FALSE);
19         //var_dump($title);
20         foreach ($title as $index){
21             Film::create([
22                 'title' => $index,
23                 'story' => $faker->overview(),
24                 'released_at' => $faker->date($format = 'Y-m-d', $max = 'now'),
25                 'duration' => $faker->time($time = '1'|'2'|'3'),
26                 'info' => $faker->paragraph($nbSentences = 3, $variableNbSentences = true),
27             ]);
28         }
29     }
30 }
31
```

GENRE SEEDER

```
1 |<?php
2
3 use Illuminate\Database\Seeder;
4 use Faker\Factory as Faker;
5 use App\Genre;
6
7 class GenreTableSeeder extends Seeder
8 {
9     /**
10      * Run the database seeds.
11      *
12      * @return void
13      */
14     public function run()
15     {
16         $faker = Faker::create();
17         $faker->addProvider(new \Xylis\FakerCinema\Provider\Movie($faker));
18         $genre=$faker->movieGenres(20, $duplicates=FALSE);
19         foreach($genre as $index){
20             Genre::create([
21                 'genre' => $index,
22             ]);
23         }
24     }
25 }
26
```

PRODUCER SEEDER

```
1 <?php
2
3 use Illuminate\Database\Seeder;
4 use Faker\Factory as Faker;
5 use App\Producer;
6
7 class ProducerTableSeeder extends Seeder
8 {
9     /**
10      * Run the database seeds.
11      *
12      * @return void
13      */
14     public function run()
15     {
16         $faker = Faker::create();
17         $faker->addProvider(new \Xylis\FakerCinema\Provider\Person($faker));
18         $producer = $faker->directors($gender = null, $count = 20, $duplicates = false);
19         foreach($producer as $index){
20             Producer::create([
21                 'name' => $index,
22                 'email' => $faker->email(),
23                 'website' => $faker->domainName(),
24             ]);
25         }
26     }
27 }
28
```

ROLE SEEDER

```
index.blade.php  web.php  ×  create.blade.php — actor  ×  create.blade.php — film  ×  RoleTableSeeder.php  ×

1  <?php
2
3  use Illuminate\Database\Seeder;
4  use Faker\Factory as Faker;
5  use App\Role;
6
7  class RoleTableSeeder extends Seeder
8  {
9      /**
10       * Run the database seeds.
11       *
12       * @return void
13       */
14     public function run()
15     {
16         DB::table('roles') -> insert([
17             'role' => 'Protagonist',
18         ]);
19
20         DB::table('roles') -> insert([
21             'role' => 'Antagonist',
22         ]);
23     }
24 }
25
26
```

SEED

```
1 |<?php
2
3 use Illuminate\Database\Seeder;
4
5 class DatabaseSeeder extends Seeder
6 {
7     /**
8      * Seed the application's database.
9      *
10     * @return void
11     */
12     public function run()
13     {
14         $this->call(ActorTableSeeder::class);
15         $this->call(FilmTableSeeder::class);
16         $this->call(GenreTableSeeder::class);
17         $this->call(ProducerTableSeeder::class);
18         $this->call(RoleTableSeeder::class);
19     }
20 }
21
```