

Migrations

IMPORTANTE: Salvar um backup no banco de dados antes de fazer os exercícios. Se alguma coisa der errado, reverter para esse backup e tentar novamente.

1. [Criar uma migration](#) que adicione a coluna “revenue” à tabela **movies**.
2. Criar uma migration que modifique a coluna “length” à tabela **movies** para que ela tenha um comprimento de 5 números inteiros (Não esqueça de utilizar: **composer require doctrine/dbal**).
3. Criar uma migration que elimine a coluna “rating” à tabela **movies**.
4. Criar uma migration que crie a tabela **directors** com “id”, “first_name”, “last_name” e “birthday”.
5. Criar uma migration que relacione **movies** a **directors**
 - a. Adicionar a coluna “director_id” à tabela **movies**.
 - b. Adicionar uma chave estrangeira à tabela **movies** para que a coluna “director_id” faça referência à primary key da tabela **directors**.**\$table->dropForeign(['director_id']).**

Factories / Seeders

1. Gerar um [Model Factory](#) para a classe Movie. Testar o funcionamento através do **tinker**.
2. Gerar o modelo de Director e um **Model Factory** para ele. Testar o funcionamento através do **tinker**.
3. Editar o **DatabaseSeeder** e inserir 10 diretores [utilizando o Model Factory](#).
4. Editar o **DatabaseSeeder** e inserir 50 filmes (5 por diretor) utilizando Model Factory.
5. Gerar um **Model Factory** para a classe Actor.
6. Editar o **DatabaseSeeder** e inserir 50 atores utilizando Model Factory.
7. Editar o **DatabaseSeeder** para que, a cada filme inserido, sejam atribuídos 3 atores de forma aleatória.

```
$movie->actors()->sync($actors->random(3));
```