



Guilherme Bedin <guilherme0bedin@gmail.com>

#7DaysOfCode - Java 3/7: Modelando um domínio

1 mensagem

Paulo Silveira <paulo@7daysofcode.io>

9 de junho de 2022 08:03

Responder a: reply-caelum57945.activehosted.1325.8168.46734@d32a.emsend3.com

Para: Guilherme Bedin <guilherme0bedin@gmail.com>



Olá **Guilherme Bedin**, tudo bem? Fico muito feliz que você esteja aqui para mais um dia de desafios!

O desafio de ontem foi mais trabalhoso, mas como prometido, hoje também vou colocar a minha solução lá no final deste e-mail para você comparar com a sua.

Você já chegou no terceiro dia do #7DaysOfCode com Java!

Recapitulando, no primeiro dia você fez uma chamada à API do IMDB para buscar um JSON com os top 250 filmes. No segundo dia, você extraiu algumas informações desse JSON, como o título dos filmes e a URL dos pôsteres.

Como resultado, você provavelmente obteve algumas listas no seu código para guardar esses atributos, algo como:

```
//chamada da API omitida
```

```
List<String> titles = parseTitles(moviesArray);
```

```
titles.forEach(System.out::println);

List<String> urlImages =
    parseUrlImages(moviesArray);
urlImages.forEach(System.out::println);

//outras listas para anos e notas
```

No desafio de hoje, a ideia será **modelar, ou pelo menos iniciar uma modelagem melhor do seu código**.

Pensando um pouco sobre **Orientação a Objetos**, uma pergunta simples pode ajudar: no caso do seu projeto, o título e o pôster se referem a que tipo de objeto? A um filme, claro. Mas você ainda não tem nenhuma estrutura que defina o que é um filme.

Qual vai ser a cara disso? Um filme (`Movie`) deve ter os seguintes atributos:

- título (`title`)
- URL da imagem do pôster (`urlImage`)
- nota (`rating`)
- ano (`year`)

Em outras palavras, em vez de ter várias listas diferentes, uma para cada atributo do filme, é bem melhor organizar isso em uma única `List<Movie>`, onde **cada filme encapsula os seus próprios dados**. Bora implementar essa classe?

DICA

Algumas reflexões: você acha que faz sentido ter setters ou um construtor padrão? Um filme deve ser interfaceado? Deve ser

imutável? Justifique a sua decisão no seu repositório.

EXTRA

Para quem usa uma versão mais recente do Java, faz sentido usar Records aqui?

RESPOSTA DO DESAFIO DO DIA 2

- Solução do Dia 2

Não se esqueça de compartilhar o seu código no seu GitHub e nas suas redes sociais com a hashtag **#7DaysOfCode**, e também com **#feedback7DoC** caso você queira alguma ajuda.

Bons estudos e até amanhã!

Paulo Silveira

CEO e fundador da Alura



Enviado para: guilherme0bedin@gmail.com

[Cancelar a inscrição](#)

Alura, Rua Vergueiro 3185, São Paulo - SP, 04101-300, Brasil