

Lista 10 - Generics e Introdução a Collections

Prazo de entrega: 28/06/2024 até 23:59 (Sexta)

Instruções de entrega: Você deve ter um repositório em seu github para a disciplina de POO chamado **Programacao-Orientado-Objetos**.

Crie uma pasta dentro dele chamada Listas e dentro dela crie uma pasta chamada Lista10.

Para cada exercício da lista deve ser criado um projeto java.

Ao terminar a lista, suba seus exercícios no github e envie o link no formulário abaixo.

Link para o formulário de envio: <https://forms.gle/GFh28cP36bW7XmDZ7>

MONITORIAS: Serão realizados quatro dias de monitorias para auxiliar com a resolução dos exercícios. As datas e horários das monitorias são:

Dia	Horários		
Segunda-Feira	19h até 21h		
Terça-Feira	19h até 21h	-	-
Quarta-Feira	09h até 11h	19h até 21h	-
Quinta-Feira	19h até 21h	-	-
Sexta-Feira	09h até 11h	19h até 21h	22h até 23:59

OBS: Se você não conseguir participar das monitorias nos dias e horários propostos, você pode solicitar uma monitoria em uma data e horário diferente, desde que seja comunicado com pelo menos um dia de antecedência 😊

1. Crie um pacote chamado `locadora`.
2. Dentro do pacote `locadora`, crie a interface `AudioVisual`.
Interface `AudioVisual`:
 - Métodos:
 - `void exibirInfo()`: Método abstrato que imprime as informações do item audiovisual.
3. Desenvolva as classes `Filme` e `Jogo` que implementam a interface `AudioVisual`.

Classe Filme:

- Métodos:
 - `void exibirInfo()`: Método abstrato que imprime as informações do item audiovisual.
- Desenvolva as classes `Filme` e `Jogo` que implementam a interface `AudioVisual`.
- **Classe Filme:**
 - Atributos:
 - `titulo: String` - Representa o título do filme.
 - `diretor: String` - Representa o diretor do filme.
 - `atores: List<String>` - Representa a lista de atores principais do filme.
 - Construtor:
 - `Filme(String titulo, String diretor, List<String> atores)`: Inicializa os atributos com os valores fornecidos.
 - Métodos:
 - `void exibirInfo()`: Sobrescreve o método da interface `AudioVisual` para exibir as informações do filme.

- Crie getters e setters para todos os atributos.
- 4. **Classe Jogo:**
 - Atributos:
 - `titulo: String` - Representa o título do jogo.
 - `plataforma: String` - Representa a plataforma em que o jogo está disponível.
 - Construtor:
 - `Jogo(String titulo, String plataforma)`: Inicializa os atributos com os valores fornecidos.
 - Métodos:
 - `void exibirInfo()`: Sobrescreve o método da interface `AudioVisual` para exibir as informações do jogo.
 - Crie getters e setters para todos os atributos.
- 5. Desenvolva a classe genérica `Locadora<T extends AudioVisual>` que gerencia a locadora de itens audiovisuais.

Classe Locadora<T extends AudioVisual>:

- Atributos:
 - `acervo: List<T>` - Lista que armazena os itens audiovisuais disponíveis na locadora (Deve ser iniciado na declaração do atributo).
- Métodos:
 - `void adicionarItem(T item)`: Adiciona um item audiovisual ao acervo.
 - `void listarItens()`: Lista todos os itens audiovisuais do acervo.
 - `T buscarItem(String titulo)`: Busca um item audiovisual pelo título e retorna o item encontrado. (**É necessário realizar cast para utilizar o método `getTitulo()` para comparar o título recebido com o do Filme/Jogo**)

Classe App:

- Método `main`:
 - Crie instâncias de `Filme` e `Jogo` utilizando os construtores com parâmetros.
 - Crie uma instância de `Locadora<AudioVisual>` e adicione os itens criados ao acervo.
 - Utilize o método `listarItens()` para exibir as informações dos itens disponíveis na locadora.
 - Utilize o método `buscarItem(String titulo)` para buscar um item pelo título e exibir suas informações.