

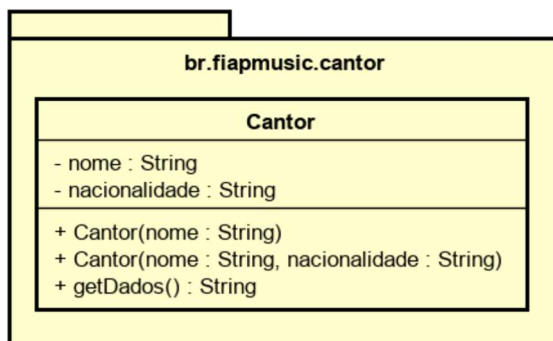
ESSE MODELO É SOMENTE PARA QUEM TEM RM TERMIMANDO COM NÚMERO ÍMPAR

Leia com atenção antes de iniciar os exercícios.

1. **Crie um projeto no eclipse.** O nome do projeto deverá ser o seu primeiro nome seguido pelo seu RM. Por exemplo, para um aluno chamado Antonio Marcos Selmini com RM 82728, o projeto deverá ter o seguinte nome → Antonio82728.
2. Faça os exercícios no mesmo projeto e, ao final, você deverá compactar (**zipar**) o projeto e enviar usando o portal da FIAP (Trabalhos Graduação).
3. Todos os projetos enviados serão submetidos ao sistema MOSS da Universidade de Stanford (<https://theory.stanford.edu/~aiken/moss/>) para detecção de similaridade. Se a porcentagem for maior que 30%, os trabalhos serão considerados cola e terão as notas zeradas.

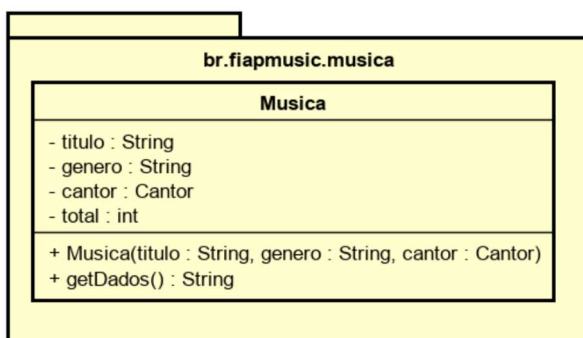
Você foi contratado(a) para desenvolver um sistema de *playlist* para um novo aparelho chamado **Fiap Music** da empresa **Fiap** que tem lançamento previsto para 2023. A equipe de engenharia de software do Renato Parducci enviou os diagramas abaixo para implementação. **Você deverá seguir rigorosamente as informações descritas nos diagramas.**

1. **(1,0)** Codificar a classe **Cantor** como definida no diagrama de classes.



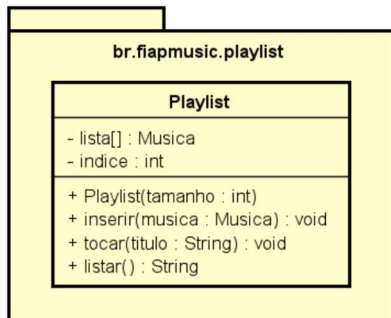
- O método **getDados()** deverá retornar todos os dados do **Cantor**.
- A codificação dos métodos **get** e **set** fica a seu critério.

2. **(1,0)** Codificar a classe **Musica** como definida no diagrama de classes.



- O **atributo total** será utilizado para armazenar o total de vezes que a música foi tocada.
- O método **getDados()** deverá retornar todos os dados da **Musica**.
- A codificação dos métodos **get** e **set** fica a seu critério.

Codificar a classe **Playlist** como definida no diagrama de classes.



- O **atributo indice** será utilizado para controlar as posições do **array lista**.
- O método construtor deverá receber o tamanho do **array lista** e instanciá-lo (**no corpo do construtor**).
- A codificação dos métodos **get** e **set** fica a seu critério.

3. **(1,0)** O método **inserir()** deverá receber um objeto do tipo **Musica** e inserir no **array lista**. O controle das posições do array fica a seu critério (**não pode gerar erro durante a execução do programa**).
4. **(1,0)** O método **tocar()** deverá receber o título de uma música e tocá-la no dispositivo (**apenas uma simulação, um faz de conta**). Ao tocar a música, você deverá incrementar (em uma unidade) o **atributo total** da música tocada.
5. **(1,0)** O método **listar()** deverá retornar os dados de todas as músicas da *playlist*.

Codificar a classe **Main** no pacote **br.fiapmusic.main** para:

6. **(2,0)** Instanciar uma *playlist* e, em seguida, gerar 4 (quatro) objetos do **tipo Musica** e inserir na *playlist*. **A forma de geração dos objetos fica a seu critério** (você pode utilizar entrada de dados via *Scanner* ou *JOptionPane* ou até mesmo definir os valores dos atributos).
7. **(2,0)** Tocar pelo menos duas músicas da *playlist* (você decide no seu código qual será tocada, fica a seu critério).
8. **(1,0)** Imprimir no vídeo os dados de todas as músicas armazenadas na *playlist*.