**Trabalho Turma TPSI-CAS-0919**

**Formadora: Fátima Pereira**

**Nome do Projeto: PlayList**

**Ano: 2019**

Crie um programa em JAVA para implementar uma playlist cuja capacidade deverá ser solicitada e definida pelo utilizador. A sua playlist deverá ser composta por músicas.

**O seu programa deverá permitir as seguintes funcionalidades:**

1. Criar, editar a playlist
2. Criar, editar e apagar músicas de uma playlist
3. Tocar uma música
4. Listar todos os atributos de uma música especifica
5. Listar todas as músicas da Playlist (título, cantar/grupo, estilo musical e duração)
6. Listar todas as músicas de um mesmo autor/grupo.

**Requisitos Técnicos Obrigatórios:**

* **Interface IPlay**Esta interface deverá conter apenas o método com a seguinte assinatura:

void play()

* **Enumerado EstiloMusical**  
  O estilo musical deverá ser implementado através de um tipo enumerado de JAVA. Insira as constantes que lhe fizeram sentido. Exemplo: ROCK, POP, CLÁSSICA, RAP, METAL, JAZZ, etc
* **Classe Musica**  
  Deverá implementar o interface IPlay. Contem os seguintes atributos:
  + Título
  + Ano de Criação
  + Estilo musical
  + Cantor/Grupo
  + Duração
  + Localização do ficheiro mp3 ou mp4  
      
    Poderá implementar o método play() recorrendo a uma chamada ao exec do sistema operativo, da seguinte forma:  
      
    public void play() throws IOException {

Runtime.getRuntime().exec("cmd /c start " + localizacao + "\\" + titulo);

}

* **Classe Playlist**Deverá conter os seguintes atributos:
  + Nome
  + Capacidade (nº de músicas que é capaz de suportar)
  + Data de criação (String)
  + Uma array de músicas
  + O nº de músicas existentes atualmente na PlayList (quando a playlist é criada, este número deverá ser zero)
* **Classe do Main**Deverá conter o método main. Esta classe é a responsável por implementar o menu de opções da aplicação e pela interação com o utilizador.
* **Encapsulamento**  
  Deve assegurar que as suas classes satisfazem o princípio básico do encapsulamento. Para isso para além de selecionar corretamente os modificadores de visibilidade (private, public, etc), deve também assegurar que não existem operações de I/O (entrada e saída de dados) nas suas classes (exceto no main), de forma a torná-las independentes do contexto.
* **Composição**

As suas classes devem recorrer ao princípio de composição, por isso é solicitado um array de Musica na classe Playlist.

* **Tratamento de Exceções**

O seu programa deverá fazer o devido tratamento de exceções. Por exemplo: se o utilizador tentar inserir uma música, numa playlist que já não tem espaço, o programa deverá gerar e tratar a exceção que deverá ser tratada de forma a que o programa não termine. Exemplo de linha de código gerar uma exceção caso o utilizador pretenda inserir uma nova música numa playlist já cheia:  
  
throw new IllegalArgumentException("ERRO: Impossivel inserir música. Playlist cheia!");

* **Legibilidade do código**

Dê nomes às variáveis e métodos consistentes com os respetivos propósitos.   
Siga as convenções das várias nomenclaturas em JAVA.  
Os métodos devem ser pequenos. Número máximo de linhas: 25.

Bom Trabalho!