

# Audiothèque Java

## Analyse, conception et implémentation

### 1 But

Analyser, concevoir et implémenter une audiothèque en Java.

### 2 Pré-requis

### 3 Consignes

#### 3.1 Généralités

Vous devez implémenter une application en Java qui gère une audiothèque, c'est-à-dire une liste de morceaux de musique.

Le but de cette audiothèque n'est pas de concurrence iTunes : on souhaite rester simple !

Chaque morceau est caractérisé par son titre, sa durée (en secondes) son (ou ses) compositeur(s), son (ou ses) interprète(s) ainsi que par son classement (0 à 5 étoiles).

Un artiste peut être à la fois compositeur et interprète.

L'audiothèque gère également des listes de lecture (playlist). Celles-ci sont identifiées par leur nom et contiennent un ou plusieurs morceaux de musique.

Il faut pouvoir connaître la durée totale d'une liste de lecture et le nombre de morceaux qui la composent.

Il y a deux types de listes de lecture : la liste de lecture classique et la liste de lecture intelligente.

#### Liste de lecture classique

La liste mémorise un ou plusieurs morceaux.

Le même morceau peut faire partie plusieurs fois de la même playlist ou de plusieurs playlists.

#### Liste de lecture intelligente

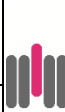
La liste des morceaux dépend de quelques critères simples :

- Une liste d'artistes, permettant de ne garder que les morceaux pour lesquels un artiste donné est soit le compositeur, soit l'interprète.
- Le classement minimum, ce qui permet de ne garder que les meilleurs morceaux.

#### 3.2 Fonctionnalités

Il faut pouvoir ajouter et retirer des morceaux de l'audiothèque.

Il faut pouvoir obtenir la liste des morceaux de l'audiothèque, ainsi que des tous les compositeurs et interprètes.

EXERCICES		INFORMATIQUE D'ENTREPRISE		 <b>CEJEF</b> DIVISION TECHNIQUE <b>ÉCOLE PROFESSIONNELLE          TECHNIQUE</b>
<b>M226 / Analyse et conception : Audiothèque</b>		EPT – INF3 – ED 04	JCO	

Il faut pouvoir créer des listes de lectures et les modifier.

Il faut pouvoir ajouter et retirer des compositeurs à un morceau.

Il faut pouvoir ajouter et retirer des interprètes à un morceau.

Le titre, la durée et le classement d'un morceau doivent pouvoir être changés.

Il faut pouvoir comparer deux morceaux afin de savoir s'ils sont identiques.

## 4 Travail à effectuer

### 4.1 Modélisation

À partir des informations données ci-dessus, réalisez un diagramme de classes complet de l'application.

Les différentes méthodes et propriétés de chaque classe doivent être représentées. Les liens entre les classes et les cardinalités doivent également être présentes.

Il n'est pas nécessaire de représenter en détail tous les constructeurs, ni les accesseurs/mutateurs, à moins que cela soit utile à la compréhension du diagramme.

### 4.2 Implémentation

Une fois que la modélisation a été validée par votre enseignant, vous pouvez implémenter votre diagramme de classes en Java.

Pour chaque classe, il faut implémenter la méthode `toString()`.

La méthode `equals()` doit être implémentée dans les classes pour lesquelles cela serait utile.

Pour tester votre application, voici quelques exemples de morceaux que vous pouvez instancier :

**Brûleur de Frontières** (3'59), compositeur : *Lino*, interprètes : *Lino* et *Corneille*

**Choc funèbre** (4'05), compositeur/interprète : *Lino*

**Le flingue à Renaud** (4'53), compositeur/interprète : *Lino*

**Suicide commercial** (5'30), compositeur/interprète : *Lino*

**Madame l'existence** (3'56), compositeur/interprète : *Jacques Dutronc*

**La Vie en live** (3'45), compositeur/interprète : *Jean-Jacques Goldman*

**Ensemble** (3'59), compositeur/interprète : *Jean-Jacques Goldman*

**Tournent les violons** (4'38), compositeur/interprète : *Jean-Jacques Goldman*

**Un goût sur tes lèvres** (4'29), compositeur/interprète : *Jean-Jacques Goldman*

**J'oublierai ton nom** (4'42), compositeur : *Jean-Jacques Goldman*, interprètes : *Johnny Hallyday*, *Carmel*

**Tu peux chercher** (4'55), compositeur : *Jean-Jacques Goldman*, interprète : *Johnny Hallyday*

**Sad Situation** (3'37), compositeur/interprète : *Carmel*

**Rome** (3'18), compositeur/interprète : *Corneille*