

Examen #1 pratique

Créez un projet « Examen1 ». Dans le dossier « src » de ce projet créez un dossier

« documents ».

Lire attentivement les consignes de remise données en annexe.

Vous disposez du fichier **radio.txt** (qu'est dans LEA) et le déposer dans le dossier **donnees** du

dossier **src**, où chaque ligne contient les informations relatives à une radio d'auto en vente dans

un magasin :

```
Sony WM323 0 1 0 199.99
Kenwood KW1000 1 1 0 249.99
Alpine A200 0 0 0 49.75
Koss KS600 1 1 1 219.99
Sony WM823 0 1 0 99.99
Koss KS1111 1 0 1 79.98
Kenwood KW1090 1 0 0 149.99
etc.
```

CD Cassettes MP3

- La première colonne, représente le fabricant du radio (Sony, Kenwood, Alpine, ...)

- La deuxième colonne, représente le numéro du modèle.

- Les colonnes 3, 4 et 5, correspond aux options disponible sur la radio, soit : un lecteur de

CD, un lecteur de cassettes et un décodeur MP3. Si la valeur est 1 alors l'option est présente, 0 si elle ne l'est pas.

- Par exemple, pour les options 1 1 0, la radio possèderait un lecteur CD et un lecteur de

cassettes, mais pas de décodeur MP3.

- La dernière colonne (le restant de la ligne), est un réel représentant le prix de l'appareil.

Notez qu'il y a, au maximum, 15 radios dans le fichier (si vous utilisez un tableau classique).

Vous devez :

Dans la classe **Radio**

1) Concevoir la classe **Radio** en respectant le principe d'encapsulation :

A. Tous les attributs sont des « String » sauf le prix qu'est un « double ». Vous pouvez

aussi déclarer les colonnes 3, 4 et 5 comme des int ou boolean.

B. Prévoyez deux constructeurs, un par défaut et un autre paramétré.

C. Prévoyez les **méthodes d'accès (get)** et de **modifications (set)** nécessaires.

D. Prévoyez une méthode **toString** pour afficher les attributs d'un objet Radio selon le

format du fichier (voir plus haut).

NOTA : 5% par minute de retard sur l'heure prévue de remise dans LEA. Vous inspirer de la

classe **Livre** développée en cours ainsi que d'autres exemples que vous avez dans LEA. Tout y

est pour faire votre examen.

2

E. Prévoyez une méthode pour déterminer l'égalité entre 2 radios (selon la marque et le

modèle). Deux radios (deux instances (objets) de la classe Radio) sont égales si elles

ont la même marque et le même modèle.

F. Prévoyez d'autres méthodes si vous avez besoin et qui ont un rapport avec une Radio.

Dans la classe GestionRadios

2) Concevoir la classe **GestionRadios** qui contiendra le « main » et d'autres méthodes tel que

chargerRadios et autres selon les besoins.

3) Lire le fichier de données et créer un conteneur par un tableau classique ou ArrayList

listeRadios de **Radios** (instances de la classe **Radio**) via la méthode **chargerRadios**.

À partir du conteneur listeRadios

4) Options de traitement via un menu de la méthode **menu()** (vous avez déjà le code d'un des

cours précédents) :

***** Gestion des Radios *****

1 Option A

2 Option B

3 Option C

4 Option D

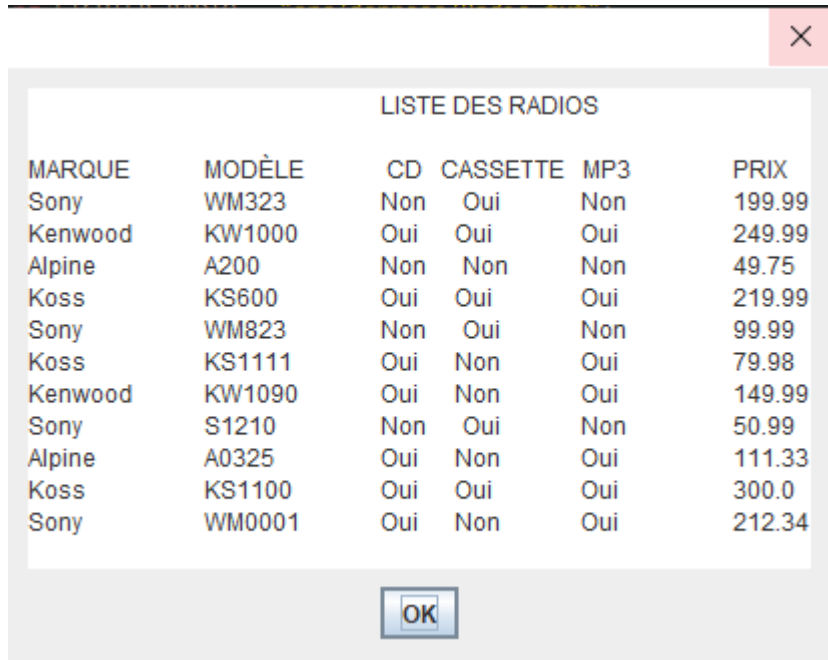
5 Option E

6 Terminer

Entrez votre choix [1-7] :

Ici CE signifie Copie Écran

A. Créer une méthode **ListerRadios()** qui va afficher toutes les radios dans le format donné par le fichier en y ajoutant le nom des colonnes. Utilisez un **ShowMessageDialog** et par l'utilisation de la méthode **toString** de la classe **Radio**. **CE**



MARQUE	MODÈLE	CD	CASSETTE	MP3	PRIX
Sony	WM323	Non	Oui	Non	199.99
Kenwood	KW1000	Oui	Oui	Oui	249.99
Alpine	A200	Non	Non	Non	49.75
Koss	KS600	Oui	Oui	Oui	219.99
Sony	WM823	Non	Oui	Non	99.99
Koss	KS1111	Oui	Non	Oui	79.98
Kenwood	KW1090	Oui	Non	Oui	149.99
Sony	S1210	Non	Oui	Non	50.99
Alpine	A0325	Oui	Non	Oui	111.33
Koss	KS1100	Oui	Oui	Oui	300.0
Sony	WM0001	Oui	Non	Oui	212.34

B. Créer une méthode **ListerRadiosParMarque(String uneRadio)** qui va afficher (comme en A.) toutes les radios du conteneur qui sont d'une marque donnée. Demander à l'utilisateur de taper la marque de radio par un **ShowInputDialog** et après appeler la fonction **ListerRadiosParMarque(String uneRadio)**. Exemple pour les **copies écran** :

- Afficher seulement les radios de marque **Sony** **CE**
- Afficher seulement les radios de marque **Koss** **CE**

3

Ne fonctionne pas!

C. Créer une méthode **modifierPrixRadio()** qui va changer les prix des radios. Dans

cette méthode vous allez demander à l'utilisateur via des **ShowInputDialog** de taper

la **marque** de radio et après le **taux** de changement. Par exemple 0.15 qui signifie une

augmentation de 15% ou -0.07 qui signifie une diminution de 7%. Pour les **copies écran** :

Ne fonctionne pas!

- **Augmenter** le prix de tous les radios de marque **Koss** de 15% **CE**
- **Diminuer** le prix de tous les radios de marque **Sony** de 7% **CE**

Faire appel à la méthode **ListerRadios()** pour afficher les informations de radios.

D. Créer une méthode **ajouterRadio()** qui va demander via des **ShowInputDialog** (via des ayant les caractéristiques suivantes (appel au constructeur avec ces valeurs pas besoin de les lire à l'écran) :

- **Sanyo**, modèle **S5210**, avec **CD** et **MP3** mais pas de cassettes, **299.99**

CE appel option A

×

LISTE DES RADIOS					
MARQUE	MODÈLE	CD	CASSETTE	MP3	PRIX
Sony	WM323	Non	Oui	Non	199.99
Kenwood	KW1000	Oui	Oui	Oui	249.99
Alpine	A200	Non	Non	Non	49.75
Koss	KS600	Oui	Oui	Oui	219.99
Sony	WM823	Non	Oui	Non	99.99
Koss	KS1111	Oui	Non	Oui	79.98
Kenwood	KW1090	Oui	Non	Oui	149.99
Sony	S1210	Non	Oui	Non	50.99
Alpine	A0325	Oui	Non	Oui	111.33
Koss	KS1100	Oui	Oui	Oui	300.0
Sony	WM0001	Oui	Non	Oui	212.34
Sanyo	s5210	Oui	Non	Oui	299.99

OK

E. Créer une méthode **chercherRadio()**, qui va chercher la radio Sony, modèle WM823, puis modifier ses informations comme suit (seulement pour cette radio, pas besoin de lire à l'écran):

- Ajouter un décodeur MP3 et modifier son prix à 169.99. **CE appel option A**

Ne fonctionne pas!

Chaque méthode peut appeler d'autres méthodes auxiliaires (déclarées comme «private») si nécessaire.

Évaluation :

Chaque option vaut 17% pour un total de 85%, si non fonctionnelle ne pourra pas dépasser 8%. Ces 8% seront le maximum pour le code développé dont celui-ci pourra aller de 0% jusqu'à 8%.

Les 15% restants seront distribuées comme suite :

10% pour la qualité de l'utilisation de la POO.

5% pour l'ergonomie au niveau de la présentation des résultats et du menu.

NOTA

Vous pouvez utiliser toutes les notes de cours mais pas de « cell » ni de courriel.

4

ANNEXE

Créez un document Word nommé **testsExamen1**. Vous devrez y mettre pour chaque option du menu le titre suivant (comme exemple) **Option A** en caractères gras et taille 18 points et après les copies écran. Chaque écran doit comporter **les informations lues à l'écran** ainsi que **le résultat**

via l'affichage du conteneur **listeRadios** donnée par **l'option A du menu**. Par exemple dans

l'option B faudra lire la marque de la radio et donc faudra afficher le message avec la marque que vous avez tapé ainsi que le résultat.

Faites des copies écran de : votre menu, résultat des Option A jusqu'à Option E. Indiquez dans

ce même document *toute question qui ne fonctionne pas et donner une idée du provient le*

problème. Je regarderai le code pour pouvoir vous donner quelques points au lieu de zéro. Si

pas de copie écran -50% de la question. Si une question ne fonctionne pas et vous n'avez pas

indiqué dans votre rapport alors vous aurez zéro pour la question.

Respectez ces règles, ce sera compliqué après pour apporter des correctifs à vos notes. Les

excuses du non-respect de ces consignes ne seront pas considérées.

Déposez « testExamen1 » (convertit en format pdf) dans le dossier « documents »

absolument. Les pénalités de -20% si cette consigne n'est pas respectée et -10% pour toute autre consigne donnée dans cet énoncé et qui n'a pas été respectée.

NOTA : Placer votre dossier **Examen1** dans **LEA**. Ne pas placer le dossier

Examen1 dans un autre dossier. Regarder l'heure de remise dans **LEA**.