

# Evaluation Java UDEV3

## 1. Instructions

Cette évaluation doit permettre de vérifier les connaissances de l'apprenant (Acquis, en cours d'acquisition et non acquis) sur les points vus pendant la formation java.

Pour cela vous trouverez 3 exercices :

1. Exercice 1 – permet d'évaluer vos compétences sur la syntaxe de base.
2. Exercice 2 – permet d'évaluer vos compétences sur les notions de la P.O.O avec java
3. Exercice 3 – permet d'évaluer vos compétences sur les notions de la P.O.O avec java

Durée estimée : 2 heures

Merci de créer un fichier zip contenant les sources. Nommer votre fichier zip avec votre nom et prénom. Par Exemple : Nom\_Prenom.zip

Envoyer ce fichier à l'adresse mail suivante : [david.jacquot@epsi.fr](mailto:david.jacquot@epsi.fr)

## 2. Exercice 1

Ecrire un programme qui affiche toutes les manières possibles d'obtenir un euro avec des pièces de 2 centimes, 5 centimes et 10 centimes.

Dire combien de possibilités ont ainsi été trouvées.

Les résultats seront présentés ainsi :

1 euros = 50 x 2 centimes

1 euros = 45 x 2 centimes + 2 x 5 centimes

1 euros = 40 x 2 centimes + 4 x 5 centimes

1 euros = 35 x 2 centimes + 6 x 5 centimes

1 euros = 30 x 2 centimes + 8 x 5 centimes

...

1 euros = 5 x 2 centimes + 9 x 10 centimes

1 euros = 2 x 5 centimes + 9 x 10 centimes

1 euros = 10 x 10 centimes

En tout, il y a ?? façons de faire 1 euro

### 3. Exercice 2

Définissez les classes Guichet, Personne et Ticket de façon à simuler le déroulement suivant :

1. Création d'un objet g de la classe Guichet : ce guichet n'a pas fabriqué de ticket.
2. Création d'un personne pp de la classe Personne : cette personne n'a pas de ticket.
3. La personne pp demande un ticket au guichet g ; celui-ci fabrique un ticket qui reste en possession du guichet.
4. Le guichet g donne le ticket à la personne pp ; pp détient alors le ticket et g n'est plus en sa possession.

### 4. Exercice 3

Définissez une application qui permette de gérer un agenda où sont répertoriées des personnes avec leur nom, leur adresse et leur numéro de téléphone. L'application doit proposer un menu qui permette à l'utilisateur de choisir un traitement parmi les suivants :

- Ajout d'une personne
- Affiche complet
- Recherche d'une personne pour afficher son adresse et son numéro de téléphone

Il est demandé de définir une classe Personne et une classe Gestion\_Agenda structurée par une variable de type List<Personne>

## 5. Exercice 4 (à rendre pour le 7 novembre)

Créer un programme qui permet de gérer une pizzeria.

Ce programme doit permettre de :

- ✓ Créer différentes pizzas à partir d'une liste d'ingrédients
- ✓ Lister les pizzas créées
- ✓ Supprimer une pizza de la liste

Un ingrédient est caractérisé par son id, son nom et son prix.

La liste des ingrédients est la suivante :

- Tomate, 2€
- Œuf, 2€
- Pate, 50 cts
- Sauce tomate 20 cts
- Champignon 1€
- Fromage de chèvre 2€
- Poivron 1€
- Mozzarella 1€
- Emmental 2€
- Roquefort 3 €
- Jambon 1€
- Olive 1,50 €
- Lardons 2€
- Oignons 1€
- Reblochon 3€
- Magret 5€

Les ingrédients doivent être stockés dans un tableau (Ne pas utiliser les listes)

L'utilisateur doit pouvoir visualiser le tableau des ingrédients par prix croissant ou par nom des ingrédients trié.

Une pizza a un id (qui doit s'auto incrémenter automatiquement lors de la création de celle-ci), un nom et un prix ainsi qu'une liste d'ingrédient.

Une pizza peut contenir de 3 à 6 ingrédients maximum.

L'utilisateur peut créer sa propre pizza à partir de la liste des ingrédients

Il peut aussi afficher la liste des pizzas créées (avec nom de la pizza, prix de la pizza et la liste des ingrédients)

Le prix d'une pizza est défini en fonction du prix de chaque ingrédient auquel il faut rajouter 3€.

Faire les contrôles nécessaires pour éviter les plantages.

Je vous conseille de créer une classe Pizzeria, une classe Pizza et une classe Ingredient