

Java - Initiation

Évaluation finale

EVALUATION FINALE (1/5)

Chaque question compte pour 4 points.

Tous les fichiers de ces exercices seront créés dans le même projet « evaluation ». Dans ce projet, il faudra créer une classe « Main » avec la méthode « main » (le point d'entrée du programme). Dans les exercices, les codes indiqués comme devant être dans la méthode « main » seront placés les uns à la suite des autres.

Documents autorisés (le cours, Internet...) mais TRAVAIL DE GROUPE INTERDIT (il sera pénalisé s'il est détecté à la correction)

1. Dans la classe « Main », écrire une méthode « tableDe7(int n) » qui :
 - a) prend en paramètre un entier n et affiche les n premiers termes de la table de multiplication par 7
 - b) affiche une étoile « * » à côté des termes qui sont aussi divisibles par 3 (càd : `terme % 3 == 0`)

Dans la méthode « main », écrire la ligne « `tableDe7(8)` ». L'exécution devra afficher :

7 14 21* 28 35 42* 49 56

SUITE PAGE SUIVANTE...

EVALUATION FINALE (2/5)

...

2. En respectant les bonnes pratiques de **l'encapsulation**, coder une classe « Chien » possédant les attributs « nom », « poids » et « age » (à vous de trouver des types pertinents) ainsi qu'une méthode « aboie » qui **affiche** « WOUF ! » et une méthode « toString » qui **retourne** un texte correspondant aux exemples ci-dessous.

Dans la **méthode « main »**, créer deux instances :

- le chien « Medor », 8 ans, qui pèse 12,5 kg
- le chien « Rex », 2 ans, qui pèse 3,8 kg

Ensuite, toujours dans la **méthode « main »**, en utilisant les « toString » de « Medor » et de « Rex », afficher :

Le chien Medor, age de 8 ans, pese 12.5 kilos

Le chien Rex, age de 2 ans, pese 3.8 kilos

SUITE PAGE SUIVANTE...

EVALUATION FINALE (3/5)

...

3. Créer deux classes publiques :

- Une classe « Etudiant » avec les attributs « nom », « prenom » et « idEtudiant » ainsi qu'un attribut de classe « promotion » (tous ces attributs seront de type texte)
- Une classe « Module » avec l'attribut « matiere », l'attribut « coefficient » et l'attribut « étudiants » (liste des étudiants inscrits à cette matière)

Dans la **méthode « main »**, écrire le code qui initialise les instances d'« Etudiant » et de « Module » et qui affiche :

Liste des etudiants de la promotion 2022 inscrits en Java (coef. 2) :

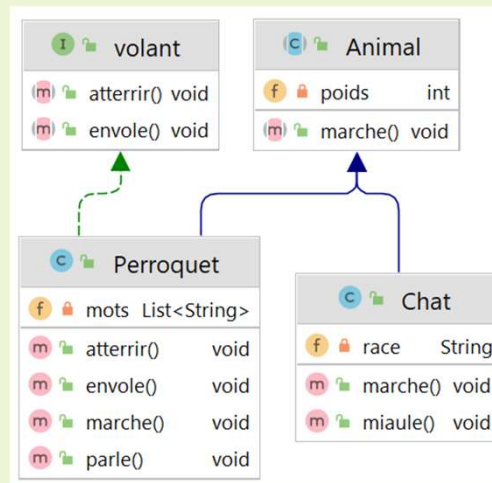
- Marcel Dupont
- Jean Durant
- Marc Duchemin

SUITE PAGE SUIVANTE...

EVALUATION FINALE (4/5)

...

4. Donner le code Java correspondant au **diagramme UML** suivant :



Remarques :

- il faut juste donner le code des classes (pas de code à rajouter dans la classe « Main »)
- pas besoin de mettre de code entre les « { » et « } » des méthodes

SUITE PAGE SUIVANTE...

EVALUATION FINALE (5/5)

...

5. Mettre en place le **polymorphisme** de manière à ce que le code suivant, ajouté à la **méthode « main »**...

```
List<BebeAnimal> animalerie = new ArrayList<>();
animalerie.add(new Chaton());
animalerie.add(new Canari());
animalerie.add(new Chiot());
for (BebeAnimal elem : animalerie) {
    elem.crie();
}
```

... affiche le résultat suivant :

MIAOU !
CUI CUI !
WOUF !