

Rattrapage, Travaux dirigés.

Benoît Valiron
benoit.valiron@lipn.univ-paris13.fr

8 décembre 2010

Exercice 1.

1. Dans la classe suivante, quels sont les champs, constructeurs et méthodes, et quels sont leurs types.

```
public class Personne {
    String nom;
    int age;

    Personne(String s, int i) {
        nom = s;
        age = i;
    }

    public String retourneNom() {
        return(nom);
    }

    public void changeAge(int i) {
        age += i;
    }

    public int retourneAge(){
        return(age);
    }

    public static void main(String[] args) {
        // DU CODE
        // ENCORE DU CODE
    }
}
```

2. À la place des commentaires de la méthode main, comment feriez-vous les choses suivantes :
 - (a) Allouez un nouvel objet `p` de type `Personne`, avec nom Bob et age 99.
 - (b) Ajoutez 12 à son age.

- (c) Dans une nouvelle variable `s` de type `String`, placez son nom et son age, séparés par un espace, à l'aide des méthodes correspondantes.
3. Ajoutez une méthode `toString` à la classe qui fait la même chose que dans la question 2(c).

Exercice 2.

Considérez les classes suivantes :

```
public class A {
    int a;
    int b;

    // constructeurs

    int getX(int i){
        // retourne a si i=0, b si i=1, et 0 dans tous les autres cas.
    }
}
et
public class B {
    public static void main(String[] args) {
        // DU CODE
        // DU CODE
    }
}
```

1. Placez les constructeurs de la classe `A` : il y en a trois : un sans arguments, un avec 1 argument, et un dernier avec 2 arguments. Celui sans argument initialise `a` et `b` à 4, le deuxième met son argument dans `a` et met `b` à 0, le troisième initialise `a` et `b` avec ses deux arguments.
2. Quel est le code de la méthode `getX` ?
3. Dans la méthode `main` de la classe `B`, initialisez trois objets de type `A` en utilisant les trois constructeurs. Affichez la valeur du champ `a` de la première variable en utilisant sa méthode `getX`.

Exercice 3.

Créez une classe `tamponEntiers` avec

1. deux champs :
 - un tableau d'entiers `tab`,
 - un entier `int`.
2. un constructeur qui attend un argument de type `int` pour initialiser le tableau à la taille de l'argument. Initialisez `index` à 0.
3. deux méthodes :
 - `public void agrandir()` qui multiplie par deux la taille du tableau
 - `public void ajouter(int e)` qui ajoute un élément dans le tableau l'indice `index`. La méthode incrémente `index` après ça.

Exercice 4.

1. Créez une sous-classe `Etudiant` à la classe `Personne`. Cette sous-classe comporte un champs `numEtudiant` de type `long`. Le constructeur attend les mêmes arguments que celui de la classe `Personne`, plus un, pour le numéro d'étudiant.
2. Sur le même principe, créez une classe `Prof`, sous-classe de `Personne`. Elle possède un champ `bureau` de type `String`. Le constructeur doit aussi initialiser ce champ.
3. Redéfinissez la méthode `toString` de `Personne` pour prendre en compte les nouveaux champs.
4. À la place des commentaires, dans la classe suivante :

```
public class Test {
    public static void main(String[] args) {
        // DU CODE
        // DU CODE
    }
}
```

créez deux nouvelles variables de type `Personne` : un nouvel étudiant `Bob` et un nouveau prof `Ben`.

Exercice 5.

1. Comment feriez-vous pour que la classe `A` implémente l'interface `Runnable` ?
2. Dans la classe `Test`, créez une variable `r` de type `Runnable` et placez un nouvel objet de type `A` dedans.

Exercice 6.

Commencez l'exercice 2 de la série 2.