

Programmation - Java - Fiches TD

Marie Beurton-Aimar

1 Prise en main

1. Programme Addition

- Taper le programme suivant sous `emacs`.

```
public class Addition
{
    /**
     * Classe sans reel intérêt
     * @author Qui vous voulez
     */

    public static void main ( String [] arg )
    {
        // ceci est une simple addition pour voir si tout marche bien
        System.out.print ( " 123 + 321 font " ) ;
        System.out.println ( 123 + 321 ) ;
    }
    /* Commentaire qui n'ira pas dans la documentation */
}
```

- Compiler le programme (`javac Addition.java`) puis exécuter le (`java Addition`). Comment fonctionne t-il ? Définir le rôle des symboles '+' contenus dans ce programme.
- Générer une documentation de la classe `Addition` en tapant `javadoc Addition.java`. Regarder la documentation ainsi générée avec `netscape` en lui donnant le chemin absolu du fichier généré.

2. Version suivante:

- Compiler et exécuter le programme `Variables.java` suivant.

```
/**
 * Petit exemple de carre
 *
 * @author Master Bioinfo
 */

public class Variables
{
    public static void main ( String [] arg )
    {
        /** affiche le carre de l'entier suivant**/
        int number = 25;
        int carre_number = number * number;

        System.out.println ( number + " * " + number + " = " + carre_number ) ;
    }
}
```

- Définir le rôle des symboles '+' ?
- Générer une documentation commune aux classes `Addition` et `Variables` avec le nom des auteurs (`javadoc -author Addition.java Variables.java`).

3. Encore un exemple :

- Taper et compiler le programme `BogusPocus.java`.

```
public class BogusPocus
{
    /** Utilisation de plusieurs types de variables **/

    public static void main(String[] arg)
    {
        int $n = 45 % 8;
        float 1er = 8.5 + 8f;
        String _S = " Test de chaine " + '!' ;
        boolean flag-1 = true ;

        System.out.println($n + 1er + _S + " " + flag-1);
    }
}
```

- Qu'observez-vous ? Corriger toutes les erreurs.

2 Respectons la tradition

1. Ecrire un programme Java qui correspond à l'exécution suivante :

```
> java Bonjour
Bonjour, X!
> java Bonjour Pierre
Bonjour, Pierre!
```

2. Modifier ce programme pour qu'il accepte plusieurs noms en paramètres. On aura l'exécution suivante :

```
> java BonjourPlus Pedro Antonio Ramirez
Bonjour, Pedro Antonio Ramirez Gomez!
```

3. Ecrire différentes versions de ce programme utilisant les structures de boucles `while` `for`.

3 Boucle et tableaux

- Ecrire un programme que stocke dans un tableau un triangle de Pascal i.e dont la valeur de la case de ligne `i` et de colonne `j` est égale à la somme des valeurs des cases `[i-1][j-1]` et `[i-1][j]`.

4 Exercice Molécule.0

- Ecrire un programme qui permet la saisie des noms de `n` molécules, `n` donné par l'utilisateur. Vous stockerez ces noms dans un tableau.
- Ecrire une fonction qui permet d'afficher le nom de toutes les molécules.
- Faire un petit menu qui demande à l'utilisateur ce qu'il veut faire.