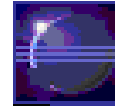




TP N°1



Objectifs	Temps alloué	Outils
<ul style="list-style-type: none"> - Se familiariser avec l'environnement de développement Java - Apprendre les commandes nécessaires à la compilation et exécution d'un programme. - S'initier aux concepts de classe, d'attributs et de méthodes 	3h00	Bloc Note/NotePade JDK 1.8 Eclipse

Compilation et exécution

Les commandes à connaître !!

- Compilation d'un fichier source
javac <nom de fichier.java>
 ex : javac Bonjour.java
- Une classe ne peut être exécutée que si elle contient une méthode **main()** correctement définie. Pour exécuter un fichier contenant du bytecode, toujours dans le même répertoire (dans lequel est créé le fichier .class), il suffit d'invoquer la commande java avec le nom du fichier source avec ou sans son extension .class
java <nom de classe>
 ex : java Bonjour
- il doit exister une classe Bonjour comportant une certaine méthode main
- Le nom du fichier doit correspondre au nom de la classe principale en respectant la casse même si le système d'exploitation n'y est pas sensible.
- Avec Eclipse tout cela devient transparent.

Activité 1 :

Tous à vos console !

- 1- Ecrire l'incontournable "Hello World" dans un fichier nommé HelloWorld.java

```
public class HelloWorld {
    public static void main(String argv[]) {
        System.out.println("Hello World\n");
    }
}
```

- 2- A l'aide de la console, compiler puis exécuter le fichier créé.
- 3- Modifier le programme pour qu'il affiche Hello <votre nom>
- 4- Modifier le programme précédent pour qu'il affiche Hello <le premier argument que vous passerez au programme>

Pour faire un affichage du premier paramètre de la ligne de commande :

```
System.out.println("Hello"+argv[0]);
```

Compiler puis exécuter en passant un paramètre à la commande d'exécution :

Ex : `java HelloWorld <votre nom>`

- 5- Modifier le programme précédent afin que pour chaque nom donné sur la ligne de commande, il affiche une ligne de salutation.

NB : L'instruction `argv.length` permet de renvoyer la taille du tableau `args`.

Activité 2 :

- 1- Copier le programme suivant et enregistrer le sous un fichier nommé

`Bienvenu.java`

```
import java.util.Scanner;

public class Bienvenu {
    public static void main(String args[]) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Quel est votre prénom ? ");
        String prenom = sc.nextLine();
        System.out.println("\nBienvenu " + prenom + " !");
    }
}
```

- 2- Compiler et exécuter le fichier, que se passe-t-il ? Déterminez ce qu'il fait.
- 3- Exécuter en entrant un nombre à la place du prénom. Que se passe-t-il ? Pourquoi ?
- 4- Dans le même fichier, ajouter des lignes de code en utilisant la méthode « `nextInt()` » de la même façon que « `nextLine()` » pour saisir votre âge dans une variable (`int age`) et l'afficher comme suit :

```
Bienvenu Toto !
Vous avez 17 ans.
```

Activité 3 :

Ecrire un programme Java permettant de saisir au clavier le prix d'un produit et le pourcentage de la remise puis d'afficher le prix avec remise sachant que :

$$\text{prix_ar} = \text{prix} - \text{prix} * \text{remise} / 100$$

Activité 4 :

- 1- Créer un fichier `MyHello.java` contenant la classe `MyHello`
- 2- Ajouter une méthode `void Hello()` qui fait l'affichage d'une chaîne de caractères
exemple : « Bienvenue dans le monde de java ! »
- 3- Ajouter la méthode `main` dans laquelle vous faite instancier la classe `MyHello` pour créer l'objet `s` comme suit :

```
MyHello s=new MyHello() ;
```

- 4- Dans la méthode `main` faite un appel à la méthode `Hello()` de l'objet `s` comme suit :
`s.Hello()`
- 5- Ajouter un attribut `String maChaine` à la classe `MyHello`
- 6- Modifier la méthode `void Hello()` pour faire l'affichage de la chaîne de caractères attribut de la classe comme suit :

```
System.out.println(this.maChaine) ;
```

- 7- Dans la méthode `main` faite affecter l'attribut de l'objet `s` comme suit :

```
s.maChaine="Salut !" ;
```

- 8- Dans la méthode `main` faite un appel à la méthode `Hello()` de l'objet `s` comme suit :
`s.Hello()` ;