

# TD n°1 : Introduction à Java

## Informatique

### M2TTT

#### Exercice 1 - Hello World

On rappelle qu'en Java chaque classe publique doit être définie dans un fichier qui lui est propre. Le nom du fichier doit être le nom de la classe qu'il contient, auquel on ajoute le suffixe **.java**. Dans un premier temps nous allons écrire un petit programme permettant de se familiariser avec le compilateur, la machine virtuelle et les méthodes.

1. Dans un nouveau répertoire (par exemple le sous-répertoire TD1 du répertoire Java), écrire le programme suivant :

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

le sauvegarder sous le nom HelloWorld.java

2. Compiler ce programme en utilisant la commande javac puis vérifier que le fichier .class correspondant existe bien.

```
TD1$ javac HelloWorld.java
```

3. Exécuter le programme avec la commande java

```
TD1$ java HelloWorld
```

On ne spécifie pas l'extension ".class": la machine virtuelle, démarrée par la commande java, cherche d'elle même un fichier dont le nom est celui qui est fourni en lui ajoutant le suffixe .class.

#### Exercice 2

Le but de cet exercice est de se familiariser avec la programmation java dans un environnement de développement intégré. On utilisera comme environnement de développement eclipse et on créera un projet de type « hello world » appelé test

1. Ouvrir une console et lancer eclipse en tapant à partir du prompt :

```
$eclipse &
```

& permet de lancer eclipse en tache de fond et donc de pouvoir continuer à utiliser la console.

2. Créer un nouveau projet :

```
file/new/JavaProject
```

Une boîte de dialogue apparait alors.

- préciser le nom du projet : test
- choisir la localisation ou utiliser la localisation par défaut
- cliquer sur finish

Votre projet est alors créé, il porte le nom de test.

3. Il reste alors à ajouter une classe permettant d'afficher « bonjour tout le monde » par exemple. Ouvrir le projet, un répertoire src apparait. Avec un clic droit sur le répertoire src, choisir :  
New/Class.  
Une boîte de dialogue apparait alors.
  - préciser le nom de la classe : Bonjour
  - à la question which method stubs would you like to create ?  
répondre : public static void main (String[] args)
  - Cliquer sur finish
4. Un fichier Bonjour.java apparait alors. Le modifier pour écrire : « bonjour a tous »
5. Compiler en cliquant sur la flèche verte et en sélectionnant :  
Run As > Java Application  
Vous devez alors voir apparaitre le message « bonjour a tous » dans la fenêtre output.

### Exercice 3

L'objectif de cet exercice est d'utiliser la classe Scanner qui permet de lire des variables à partir du clavier.

Attention, chaque exercice aura son projet associé.

1. Créer un nouveau projet
2. En prenant exemple sur la structure suivante :  

```
public class .....//Donner le nom de la classe
{
    Public static void main(String [] args)
    {
        //Declaration des variables
        //lecture des deux entiers...
    }
}
```

Ecrire un programme qui lit successivement 2 entiers et renvoie leur somme.

3. Modifier le programme pour 2 flottants, puis pour un entier et un flottant (deux solutions)

### Exercice 4

Soit la méthode :

```
int f(int x)
{
    int res;
    res = -x*x+3*x-2;
    return res;
}
```

1. Inclure la fonction f dans une classe et écrire la fonction main() qui affiche le résultat de la fonction f(x) pour x=0.
2. Transformer la fonction main() précédente de façon à calculer et afficher les résultat de la fonction f(x) pour x entier variant de -5 à 5.

#### **Exercice 4 :Le temps qu'il fait...**

Soit la température T. Ecrire une classe qui affiche :

- froid      si  $T < 8$
- frais      si  $8 < T < 17$
- bon        si  $17 < T < 25$
- chaud     sinon

#### **Exercice 5 (facultatif)**

Le but de cet exercice est de créer une classe Complexe ( $z = a+ib$ ). Cette classe va nous permettre de manipuler des nombres complexes.

1. Déclarer la classe Complexe dans un fichier constituée de ses paramètres a et b, ainsi que de son constructeur par défaut.
2. Implémentez le constructeur par défaut en attribuant à a et b la valeur 0.
3. Ajouter un constructeur permettant d'initialiser les valeurs de a et b (pouvant être différents de 0) en les passant comme paramètres.
4. Créez une méthode permettant de saisir au clavier un nombre complexe sous la forme partie imaginaire et partie réelle.
5. Créer une méthode qui affiche à l'écran un nombre complexe sous la forme partie imaginaire et partie réelle.
6. Définir une fonction qui calcule la somme de deux nombres complexes.
7. Définir une fonction qui détermine le produit de deux nombres complexes.