

TP 0

Premiers pas en Java**1 – Présentation du TP***Objet du TP*

Ce TP consiste à écrire, compiler et exécuter des programmes simples en Java « à la main » (c'est-à-dire sans IDE), afin de comprendre ces différentes étapes et les outils associés.

Compétences travaillées dans ce TP

- Manipulation de base des commandes `javac` et `java`.
- Lecture des arguments en ligne de commande

2 – Votre premier programme*Ecriture du code*

Ouvrez l'éditeur de texte de votre choix et copiez/collez le code suivant :

```
public class Test {  
    public static void main(String[] args) {  
        System.out.println("My first program in Java !");  
    }  
}
```

Sauvegardez ce programme sous le nom `Test.java`.

Attention : le fichier doit s'appeler **exactement** comme la classe.

La méthode `main()` est le point d'entrée du programme, c'est-à-dire la méthode qui sera exécutée à son démarrage.

Compilation

Ouvrez un terminal (commande « `cmd` » sous Windows), allez dans le répertoire où se trouve le fichier `Test.java` et exécutez la commande suivante :

```
javac Test.java
```

Cette commande permet de compiler la classe Test. Le résultat de cette compilation est un fichier Test.class qui se trouve dans le même répertoire par défaut.

Vérifiez que le fichier Test.class a bien été généré. Si vous essayez de l'ouvrir dans un éditeur de texte, vous n'y comprendrez pas grand-chose car ce fichier contient du *bytecode*, c'est-à-dire des instructions destinées à la machine virtuelle Java (JVM).

Exécution

A présent, exécutez la commande suivante :

```
java Test
```

Attention : ne mettre aucune extension après « Test »

Cette commande a pour effet d'appeler la méthode main() de la classe Test. Elle devrait produire l'affichage :

```
My first program in Java !
```



Point clé

A cette étape, la JVM exécute le bytecode présent dans le fichier Test.class, et non le code source présent dans le fichier Test.java !

3 – Affichage des arguments en ligne de commande

Comme vous pouvez le voir dans le code, la méthode main() possède un paramètre `args`, qui est un tableau de chaînes de caractères.

Ce tableau contient les arguments fournis au programme en ligne de commande, c'est-à-dire écrits après le nom du programme dans la commande de lancement. Par exemple, si on exécute la commande suivante :

```
java Test some command line arguments
```

Alors le contenu du tableau `args` sera : `["some", "command", "line", "arguments"]`.

Modifiez le code de la méthode main() pour que le programme affiche le contenu du tableau « `args` ». Dans notre exemple, le programme afficherait :

```
some  
command  
line  
arguments
```

Essayez d'utiliser une boucle « classique » (avec un indice) puis une boucle de type « for each ».