

	Institut Supérieur d'Informatique et de Multimédia de Sfax
	Année Universitaire: 2014-2015
	Module: Programmation Orientée Objet (Java)
	Auditoire: LAMW 3
	Enseignants: Mohamed TOUNSI & Mohamed Abdelhedi & Rihab Maaloul
	TP n°: 1

L'objectif de ce TP est la mise en place de l'environnement de travail en Java et l'étude des fondements du langage. Ainsi, l'étudiant doit être capable, à la fin de la séance, de compiler et d'exécuter un code Java et aussi de générer automatiquement la documentation. NB: dans ce TP, nous n'allons pas utiliser un IDE.

Exercice 1

Compiler et exécuter une application Java

1. Ajouter un répertoire **TP1_Java**. Ce répertoire sera appelé répertoire de travail ou workspace. A l'intérieur de votre répertoire de travail ajouter le répertoire **exercice1**. Ce dernier est appelé répertoire de projet.
2. A l'intérieur du répertoire **exercice1**, ajouter le fichier "Hello.java". Dans ce fichier, copier le programme ci-dessous.

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

3. A partir de votre répertoire de projet, compiler le programme avec la commande "Javac". Quel est le résultat obtenu? Corriger et recompiler le programme.
4. Exécuter le programme avec la commande "Java".

Générer la documentation

1. Ajouter des commentaires dans le programme "Hello.java".
2. Générer la documentation dans un répertoire nommé "docs". Utiliser la commande "javadoc".

Exercice 2

Lire la ligne de commande

Lire la ligne de commande s'avère beaucoup plus simple en Java qu'en C. On reçoit comme argument de main un seul tableau (habituellement nommé `args`). On obtiens le nombre d'éléments du tableau par `args.length`. Les arguments de la ligne de commande seront donc dans `args[0]...args[args.length - 1]`.

1. Modifiez le programme de l'exercice 1, pour que

```
$ java Hello Toto
```

produise l'affichage: Hello Toto!

2. Que se passe-t-il si on ne donne pas d'argument sur la ligne de commande? A l'aide d'une conditionnelle, résolvez ce problème.
3. Modifiez maintenant le programme pour que chaque nom donné sur la ligne de commande, il affiche une ligne de salutation.

Exercice 3

compiler et exécuter une applet java

1. Donner une définition de l'Applet Java.
2. A l'intérieur de votre répertoire de travail ajouter le répertoire **exercice3**. A l'intérieur de ce répertoire, ajouter le fichier "Dessin.java". Dans ce fichier, copier le programme ci-dessous.

```
import java.awt.Graphics ;
import java.applet.Applet;
public class Dessin
    extends Applet {
    public void paint(Graphics g) {
        g.drawLine(10, 30, 200, 30);
        g.drawRect(20, 40, 100, 50);
        g.drawOval(140, 40, 50, 50);
        g.drawString("texte", 10, 120);
    }
}
```

3. Dans le même répertoire, créer un fichier "Dessin.html". Dans ce fichier, copier le programme ci-dessous.

```
<html>
<head>
  <title>applet Dessin</title>
</head>
```

```
<body>  
  <applet code="Dessin.class" width=400 height=300>  
  </applet>  
</body>  
</html>
```

4. Exécuter l'applet dans un navigateur.
5. Exécuter l'applet avec la commande "AppletViewer".