

TP1 : Initiation à Java**Enseignants : Madhi Kamel, Majdoub Manel****Groupes : Prepa A02****Objectifs :**

- Savoir compiler et exécuter un code Java
- Se familiariser avec l'environnement de développement Eclipse.
- Comprendre la syntaxe et les éléments de bases du langage Java.

1 Compilation et exécution

1.1 Compilation et exécution à l'aide de l'invite de commandes

1.1.1 Exemple 1

- Ouvrez un éditeur de texte et saisissez ce programme

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

- Enregistrez le fichier sous `Bonjour.java`
- Pour la compilation, ouvrez l'invite de commande (CMD) sous Windows. Puis, tapez la commande suivante et appuyez sur Entrée : `javac Bonjour.java`
- Après la compilation, le fichier `.java` est traduit vers le fichier `.class` (code octet). Nous pouvons maintenant exécuter le programme. Pour exécuter le programme, tapez la commande suivante et appuyez sur Entrée : `java Bonjour`

1.1.2 Exemple 2

- Ouvrez un éditeur de texte et saisissez ce programme

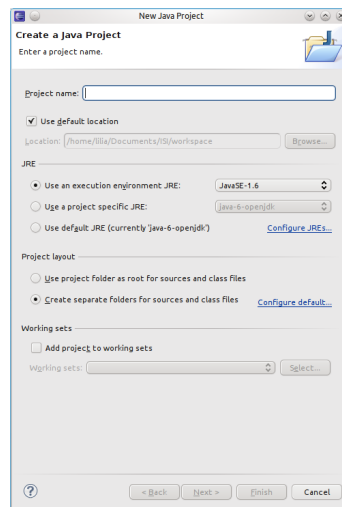
```
public class Hello {  
    public static void main(String[] args) {  
        if (args.length == 0)  
            System.out.println("Bonjour inconnu(e)");  
        else  
            System.out.println("Bonjour " + args[0]);  
    }  
}
```

- Compilez ce programme et exécutez avec la commande : `java Hello Mohamed`

1.2 Environnement de Développement : Eclipse

1.2.1 Création d'un projet

- Ouvrez l'IDE Eclipse, on vous demandera de choisir l'espace de travail dans lequel vous allez trouver vos projets. Tapez le chemin du répertoire que vous avez créé sur le bureau. Si vous n'avez pas encore créé d'espace de travail, Eclipse le créera pour vous.
- Fermez la fenêtre de bienvenue qui apparaît.
- Vous vous trouvez actuellement dans votre espace de travail.
- Créez un nouveau projet Java. Pour cela, appuyez sur File-> New -> Java Project. La fenêtre suivante apparaîtra



- Tapez comme indiqué le nom de votre projet. Conservez les réglages par défaut, et appuyez sur Finish.
- Dans l'espace de travail, sous l'onglet Packages, vous verrez que le projet est créé, et qu'il contient déjà un répertoire src (qui doit contenir tous les fichiers source que vous créez), et JRE System Library, qui est utilisée pour compiler votre code.

1.2.2 Création d'une classe

- Cliquez sur le package qui doit contenir la classe puis clic->droit sur le package, et choisir New -> Class. Dans la fenêtre qui apparaît, choisissez le nom de la classe.
- Réécrivez, compilez et exécutez le programme ci-dessus.

Exercices

Exercice 1 :

Ecrire un programme java qui permet de saisir deux entiers A et B et de vérifier s'ils sont des nombres cousins.

Deux nombres sont cousins si le produit des chiffres non nuls du premier est égal au produit des chiffres non nuls du second. Exemples :

1221 et 140 sont cousins : $1 * 2 * 2 * 1 = 1 * 4$

6002 et 143 sont cousins : $6 * 2 = 1 * 4 * 3$

105 et 150 sont cousins : $1 * 5 = 1 * 5$

Pour effectuer des lectures au clavier de valeurs, inspirez vous de l'exemple suivant :

```
import java.util.Scanner; // importer le package
public class Lectures
public static void main(String[] args)
Scanner clavier = new Scanner(System.in);
int i, x;
System.out.print("donnez un nombre entier ");
i = clavier.nextInt();
System.out.println("valeur lue : " + i);
System.out.print("donnez un nombre entier ");
x = clavier.nextInt();
System.out.println("valeur lue : " + x);
```

Exercice 2 :

Écrire un programme qui réalise la décomposition binaire d'un entier positif. Exemple : la décomposition binaire de 26 est 11010.

On pourra utiliser une chaîne de caractères pour stocker la décomposition binaire

Exercice 3 :

1. Écrivez une classe `Article` pour représenter les articles vendus dans un supermarché. Chacun comporte quatre attributs : un numéro de référence, une description textuelle de l'article, un prix hors taxe, et une quantité disponible en stock.

2. Ajouter un constructeur initialisant les propriétés.

3. Ajouter une méthode `void provision(int quantity)` permettant d'approvisionner les stocks de cet article.

4. Ajouter une méthode `boolean sell(int quantity)` permettant de vendre l'article. Si l'article n'a pas suffisamment de disponibilité, la vente est annulée et `false` est retourné.

5. Ajouter une méthode `double price()` retournant le prix TTC de l'article (appliquer une TVA de 20%).

6. Ajouter une méthode `main` qui crée deux articles et qui essaie toutes ces méthodes.