

TP1 : Configuration d'Eclipse et premier programme

Objectifs (Harmonisation 1/2)

1. Installer et configurer l'IDE Eclipse.
2. Écrire et exécuter des programmes simples en Java pour se familiariser avec les bases.

Étapes du TP

Partie 1 : Installation et configuration d'Eclipse

1. **Téléchargement et installation d'Eclipse :**
 - Rendez-vous sur le site officiel d'Eclipse : <https://www.eclipse.org/downloads/>.
 - Téléchargez "Eclipse IDE for Java Developers".
 - Suivez les instructions d'installation pour votre système d'exploitation (Windows, macOS, Linux).
2. **Configuration de l'environnement Java :**
 - Assurez-vous que vous avez installé le JDK (Java Development Kit).
 - Si ce n'est pas le cas, téléchargez et installez le JDK à partir d'[Oracle](#) ou utilisez [OpenJDK](#).
 - Configurez la variable d'environnement `JAVA_HOME` pour pointer vers votre installation JDK.
3. **Configuration d'Eclipse :**
 - Lancez Eclipse.
 - Créez un nouvel espace de travail (workspace).
 - Dans le menu principal, allez à `File > New > Java Project`.
 - Donnez un nom à votre projet : "**JeusIO**".
 - Cliquez sur **Finish**.

Partie 2 : Écrire et exécuter des programmes simples en Java

1. **Création de votre premier programme Java :**
 - Cliquez avec le bouton droit sur le nom de votre projet dans l'explorateur de projets (**Package Explorer**).
 - Sélectionnez **New > Package**.
 - Donnez un nom à votre package, par exemple **Test**.
 - Cliquez sur **Finish**.
 - Cliquez avec le bouton droit sur le package **Test** que vous venez de créer.
 - Sélectionnez **New > Class**.
 - Donnez un nom à votre classe, **HelloWorld**.
 - Cochez la case `public static void main(String[] args)`.
 - Cliquez sur **Finish**.

2. Écriture du programme "Hello, World!" et autres programmes simples :

Voici le fichier **HelloWorld.java** avec tous les programmes de test, complétez les trous ("_____") et faites fonctionner le programme.

```
package Test;

public class HelloWorld {

    public static void main(String[] args) {
        // Réflexion : Quelle est la fonction principale en Java pour
        afficher du texte à la console ? Programmez la ligne qui affiche Hello,
        World! dans la console java ici :



---


        // Programme pour afficher les types de données primitifs
        afficherTypesDeDonnees();

        // Programme pour démontrer les conditions (if/else)
        demontrerConditions();

        // Programme pour démontrer les boucles (for)
        demontrerBoucleFor();

        // Programme pour démontrer les boucles (while)
        demontrerBoucleWhile();

    }

    private static void afficherTypesDeDonnees() {

        // Ci-dessous, vous trouverez les types de données primitifs en Java.
        char monChar= 'A';
        boolean monBoolean = true;
        float monFloat = 5.75f;
        String monString = "Bonjour";
        int monInt= 10;

        // Réflexion : Comment afficher des variables en Java ? Complétez le
        code ci-dessous en remplaçant les "_____" :
        _____("Entier: " + mon_____);
        _____("Décimal: " + mon_____);
        _____("Caractère: " + mon_____);
        _____("Booléen: " + mon_____);
        _____("Chaîne de caractère: " + mon_____);

    }

    /* Expliquez l'utilisation du caractère "+" :


---



```

```

private static void demontrerConditions() {

    int nombre = 20;

    // Réflexion : Comment écrire une condition if/else en Java ?
    if (nombre > 0) {
        System.out.println("Le nombre est _____.");
    } else if (nombre < 0) {
        System.out.println("Le nombre est _____.");
    } else {
        System.out.println("Le nombre est égale à _____.");
    }
}

private static void demontrerBoucleFor() {

    // Réflexion : Comment initialiser et incrémenter une boucle for en
Java ?

    for (____; ____; i++) {
        System.out.println("Boucle for : i = " + i);
    }
}
/*

```

Que fait la première partie entre la parenthèse (et le premier ; dans l'expression for :

Que fait la deuxième partie entre les deux ; de l'expression for :

Expliquez l'utilisation des caractères "i++" :

*/

```
private static void demontrerBoucleWhile() {  
  
    int i = 0;  
  
    /* Réflexion : Comment structurer une boucle while en Java ? Nous  
    souhaitons que le programme s'exécute tant que la variable i est inférieur  
    ou égale à 7, Complétez le code pour que le résultat de l'exécution soit :  
    Boucle : i = 0  
    Boucle : i = 2  
    Boucle : i = 2  
    Boucle : i = 3  
    Boucle : i = 4  
    Boucle : i = 5  
    Boucle : i = 6  
    Boucle : i = 7  
    */  
  
    while (_____) {  
        System.out.println("_____");  
        i++;  
    }  
}  
  
/* Expliquez la différence entre public et private dans le programme  
réalisé.
```

```
/* Expliquez la différence entre public et private dans le programme réalisé :
```