|  |  |  |
| --- | --- | --- |
|  | **INSTITUT SUPERIEUR DES ETUDES TECHNOLOGIQUES DE NABEUL** | |
|  | Travaux Pratiques N° 1 : Débuter avec java | |
| **Classe : DSI** | | **Enseignante : Sana REFAI** |

## 1. Objectifs

Ce TP permet de se familiariser avec la programmation en java :

* Installation de la JVM ( Java Virtual Machine)
* Exécuter des programmes simples sous console.
* Manipuler l’outil de développement Eclipse (téléchargement, lancement, configuration et création de projets)
* Créer et exécuter un premier programme java en utilisant Eclipse

## 2. Introduction

Java est un langage de programmation orienté objet. Il a été créé en 1995 par James Gosling un ingénieur dans Sun Microsystems qui a été rachetée par la société Oracle en 2009.



Java possède plusieurs avantages:

* **Simple** : Sa syntaxe est très proche des langages C et C++
* **Orienté objet :** Il utilise les notions de classes et objets.
* **Interprété :** Un fichier source est simplement précompilé pour générer un pseudo code qui sera interprété par une machine virtuelle lors de l’exécution.



*Fichier source*

*Pseudo-code/bytecode*

Pré-compilation

Exécution (sur une JVM)

Fichier.java

Fichier.class

− **Dynamique et performant** : Le chargement des classes se fait en cours d’exécution et à la demande. Sans oublier que Java n’est pas gourmant en mémoire, la JVM a besoin de quelques centaines de kilo octets pour s’exécuter.

− **Portable** : Java est indépendant du matériel ou de la plateforme car dès qu’on a un pseudo code (.class) il est exécutable sur n’importe quel système d’exploitation disposant d’une JVM (Java Virtual Machine)



Exécution du même fichier (.class) sur

Windows

Fichier source Pseudo-code

Pré-compilation

Linux

Fichier.java

Fichier.class

Solaris

− **Robuste et sécurisé** : Il intègre plusieurs mécanismes de sécurité et d’autres qui rendent les applications plus fiables tels qu’un gestionnaire d’exception et un ramasse miette. En plus, il ne permet pas d’accès direct à la mémoire par des pointeurs (donc moins d’erreurs).

− **Riche et distribué** : Il contient plusieurs bibliothèques qui couvrent tous les domaines d’utilisation (Réseau, Système, Graphique, …) et en particulier, il a été conçu pour être exploité dans des environnements serveur et distribués (Java RMI, Socket, Servelet, …).

− **Multithread** : Java a la capacité d'effectuer plusieurs tâches simultanément. La programmation multithread est bien intégrée.

− **Intégré au Web** : Il est possible d’ajouter des applets Java dans des pages HTML.

Il existe de nombreux frameworks et API permettant l’utilisation de Java dans des domaines diversifiés. On distingue essentiellement 3 grandes éditions :

− Java SE (Standard Edition) : Destiné aux applications pour poste de travail.

− Java EE (Entreprise Edition) : Spécialisé dans les applications serveurs. Il contient un plus grand nombre d’API et d’extensions.

− Java ME (Mobile Edition) : Dédié pour les applications mobiles.

Pour coder en Java, il faut disposer d’un environnement de développement composé d’un ***JDK*** (*Java Development Kit*) et d’un ***JRE*** (*Java Runtime Environment*). Le JDK offre un ensemble d’outils y compris le compilateur et interpréteur Java. Le JRE ou environnement d’exécution contient la ***JVM*** ou *machine virtuelle Java*.

## 3. Installation JDK

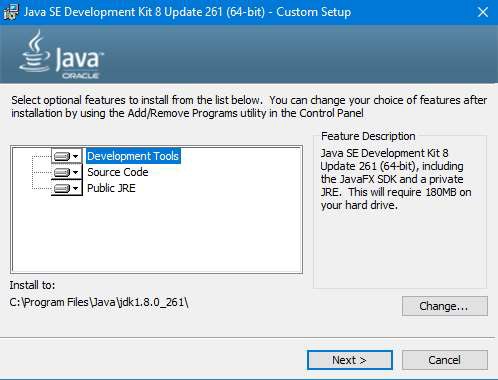
**Etape 1 : Téléchargement du JDK**

Le JDK est téléchargeable gratuitement sur le site officiel d’Oracle. La dernière version en date est le JDK 17 qui est disponible sur diverses plateformes (Linux, MacOs, Solaris et Windows). Lien : https://[www.oracle.com/java/technologies/downloads/](http://www.oracle.com/java/technologies/downloads/)

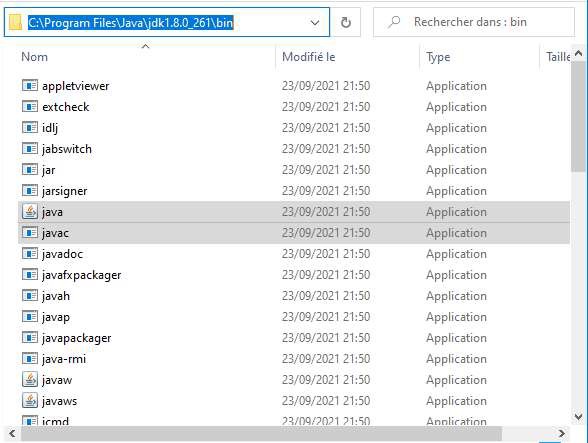
**Etape 2 : Installation du JDK**

Nous avons choisi d’installer la version 8 du JDK étant données nos besoins minimes et à cause de la taille du fichier à télécharger (seulement 170Mo).

Il suffit donc de double cliquer sur le fichier « jdk-8u301-windows-x64.exe » et suivre l’assistant d’installation.

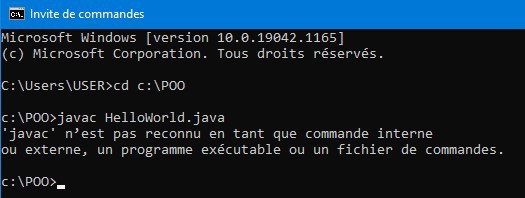


A la fin, il est possible de vérifier les différents outils disponibles dans le JDK dans le dossier « C:\Program Files\Java\jdk1.8.0\_261\bin ». Celles dont on aura le plus besoin sont « javac » pour la compilation et « java » pour l’exécution.

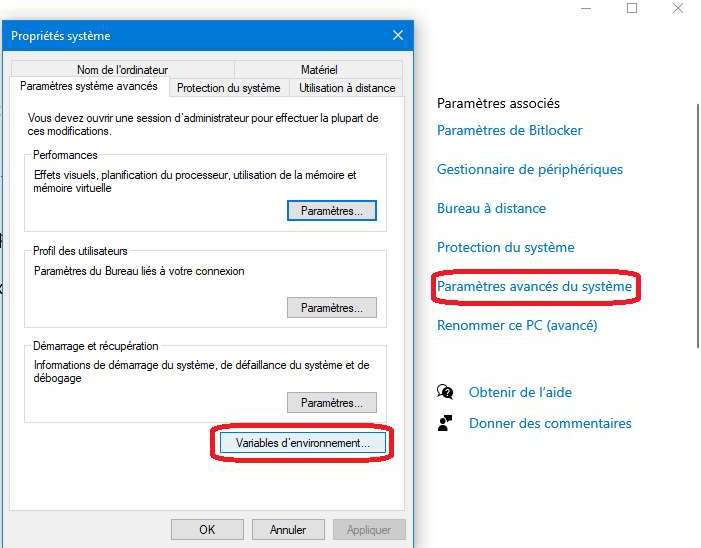


**Etape 3 : Configuration des variables d’environnement**

Lorsqu’on écrit un programme en java et qu’on désire le compiler en mode commande, l’éditeur de commande ne reconnait pas les commandes javac et java. Il faut donc configurer la variable d’environnement « Path ».

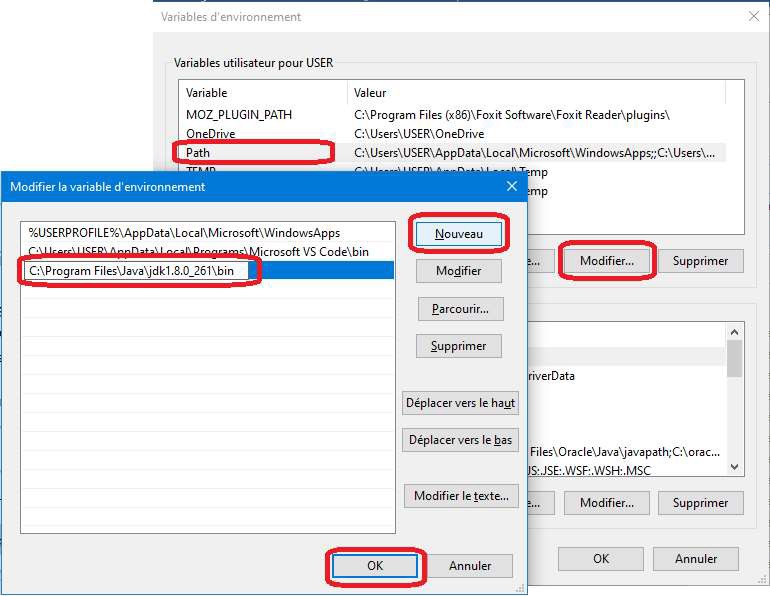


Pour accéder aux variables d’environnement sur un ordinateur, il suffit de faire un clic droit sur l’icone « My PC » puis choisir « Propriétés ». Dans la fenêtre qui s’ouvre, choisissez « Paramètre avancés du système ». Dans la fenêtre suivante, cliquez sur le bouton « Variables d’environnement ».



pour ajouter un nouveau chemin de commandes, celui des outils fournis par le JDK :

« C:\Program Files\Java\jdk1.8.0\_261\bin ». Validez toutes les fenêtres (Bouton OK).



**NB:** Cette opération est réalisée une seule fois et pour prendre en considération ces modifications, il faut fermer l’éditeur de commande et le rouvrir de nouveau.

## 4. Les Exercices

### 4.1. Exercice 1

1. Ouvrez un éditeur de texte (Bloc Note, Edit Plus ou Sublime Text) et créez sous "**c:\votre\_classe**", un nouveau fichier intitulé **Bonjour.java**
2. Copiez dedans le programme suivant :



**public class Bonjour {**

**public static void main (String args[]) {**

**//premier programme java**

**System.out.println ("Bonjour ! Voici mon 1er programme Java") ;**

**}**

**}**

1. Lancez une session DOS et allez au répertoire "c:\votre\_classe" pour pouvoir compiler le fichier à l'aide de la commande suivante : **javac Bonjour.java**
2. Maintenant, exécutez le programme en tapant la commande : **java Bonjour** Qu’affiche-t-il ? Identifier ses différentes parties.
3. Remplacer "main" par "Main", recompilez et exécutez. Que se passe-t-il ? Pourquoi ?
4. A présent, modifiez le code précédent pour ajouter des arguments au main en remplaçant l'instruction d'affichage par :

**System.out.println ("Bonjour "+args[0]+"! Voici mon 1er programme Java") ;**

1. Exécutez en tapant la commande : **java Bonjour votre\_nom**.

### 4.2. Exercice 2

1. Copiez le programme suivant et enregistrez le sous "c:\votre\_classe\Bienvenu.java"



**import java.util.Scanner;**

**public classe Bienvenu {**

**public static void main (String args[]) { Scanner sc = new Scanner(System.in);**

**System.out.println ("Quel est votre prénom ? ") ; String prenom = sc.nextLine() ;**

**System.out.println ("\nBienvenu " + prenom + " !") ;**

**}**

**}**

1. Compilez et exécuter le fichier, que se passe-t-il ? Déterminez ce qu’il fait.
2. Exécutez en entrant un nombre à la place du prénom. Que se passe-t-il ? Pourquoi ?
3. Dans le même fichier, ajoutez des lignes de code en utilisant la méthode « nextInt() » de la même façon que « nextLine() » pour saisir votre âge dans une variable (int age) et l'afficher comme suit :

Bienvenu Toto !

Vous avez 17 ans.

### 4.3. Exercice 3

Ecrire un programme Java permettant de saisir au clavier le **prix** d’un produit et le pourcentage de la **remise** puis d’afficher le **prix avec remise** sachant que :

prix\_ar = prix - prix \* remise /100

### 4.4. Exercice 4

1. Recopiez le programme de l'exercice 2 et enregistrez le dans un nouveau fichier intitulé "MaClasse.java". Puis organisez le programme comme suit, en ajoutant une méthode afficher () :

**import java.util.Scanner;**

**public class MaClasse { public void afficher() {**

**// A compléter : saisir le nom et l'âge puis afficher le message de bienvenu**

**… }**

**public static void main (String args[]) {**

**MaClasse m = new MaClasse(); // créer un objet de la classe m.afficher() // faire appel à la méthode afficher**

**}**

**}**

1. Compilez et exécuter le fichier.
2. A présent, modifiez la méthode afficher() comme suit pour accepter des arguments :

− Remplacez l'entête de la méthode ainsi : **public void afficher(int age)**.

− Supprimez, dans le corps de la méthode, la partie saisie au clavier uniquement.

1. Maintenant ajouter juste avant la méthode afficher, une ligne de code permettant de déclarer un attribut **prenom** de type entier initialisé par votre prénom.
2. Compilez et exécuter le programme obtenu. Qu'affiche-t-il ?