

TP N° : 1 Préparation de l'environnement de travail et exemples de programmes simple

1- Installation du JDK :

Eclipse a besoin d'un Runtime Java (JRE) ou d'un JDK pour fonctionner. Vaut mieux installer un JDK au lieu d'un JRE pour bénéficier automatiquement de la Javadoc et du code source de l'API Java Standard. La version de la JDK à installer est la dernière version stable disponible sur le site d'Oracle.

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

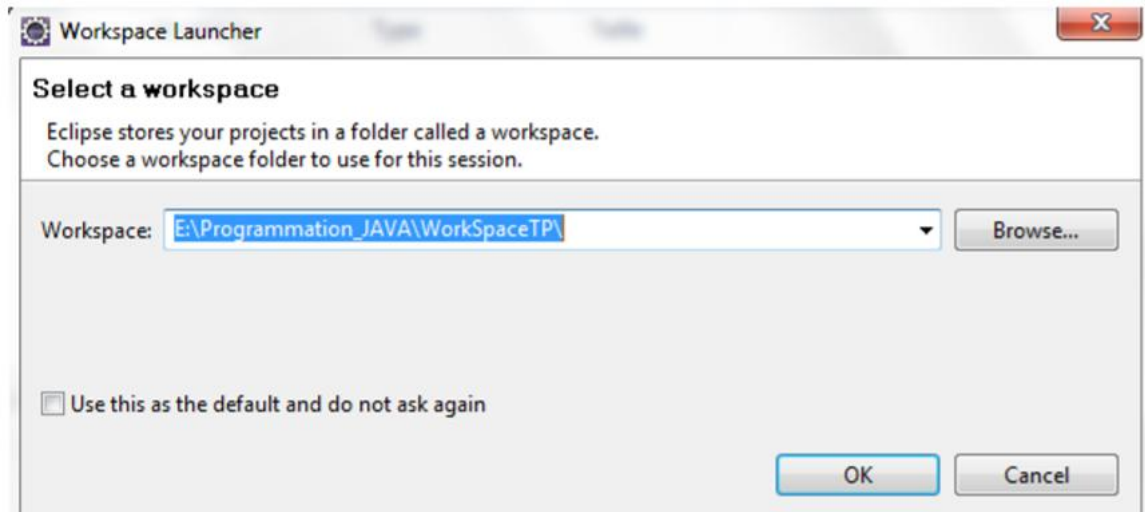
2- Java en ligne de commande :

Bien que ce ne soit pas le mode de développement le plus utilisé, Java peut être utilisé en ligne de commande. Il est toujours utile de savoir compiler et exécuter une application Java en ligne de commande. Pour compiler votre programme, vous devez utiliser le compilateur Java (**javac**). Après compilation, s'il n'y a pas eu d'erreur, un fichier .class sera créé. Pour exécuter le fichier .class, vous devez appeler l'interpréteur Java à l'aide de la commande : **java MaClasse**.

3- Installation et configuration d'Eclipse :

Eclipse est un logiciel qui simplifie la programmation en proposant un certain nombre de raccourcis et d'aides. Il est développé par IBM, il est gratuit et disponible pour la plupart des systèmes d'exploitation. Au fur et à mesure que vous programmez, Eclipse compile automatiquement le code que vous écrivez, en soulignant en rouge ou jaune les problèmes qu'il détecte. Il souligne en rouge les parties du programme qui ne compilent pas (Errors), et en jaune les parties qui compilent mais peuvent éventuellement poser problème (warning). Dans les prochains TP de Java, nous utiliserons Eclipse comme environnement de développement. Il faudrait utiliser la dernière version stable d'Eclipse qui peut être téléchargée de la page <http://www.eclipse.org/downloads>. Eclipse est utilisé pour le développement C/C++ , PHP, Java, JEE, ... pour le développement Java standard le package à télécharger est « Eclipse IDE for Java Developers » . Eclipse est un logiciel portable ainsi pour l'installer il suffit de décompresser l'archive d'Eclipse dans un répertoire de votre disque dur.

- a) Lancer eclipse.
- b) Créer votre WorkSpace dans le répertoire X:\Programmation_JAVA\WorkSpaceTP\



Qu'est-ce qu'un Workspace ?

Le workspace est le répertoire dans lequel sont stockés les différents programmes que vous allez réaliser et la configuration d'Eclipse. Ainsi si vous supprimer le workspace votre configuration Eclipse sera perdu. En toute rigueur les paramètres de configuration sont stockés dans un sous-dossier caché nommé .metadata, ainsi, effacer ce dossier ou démarrer Eclipse avec l'option -clean va supprimer ces paramètres.

Comment changer le workspace ?

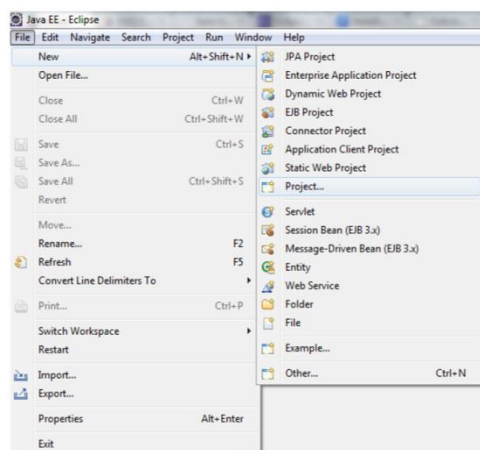
Il est possible de changer le workspace à tout moment. Le workspace en cours est alors fermé. Procédure :

- Aller dans le menu « File ».
- Sélectionner « Switch workspace ».

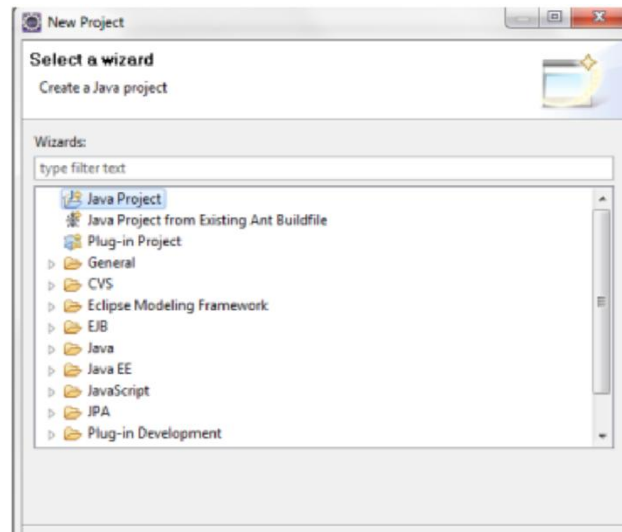
4- Création d'un projet JAVA :

- a) Pour créer un projet JAVA : File -> New -> Project.

- b) Choisir Java



Project



- c) Donner un nom à votre projet (par exemple **MyFirstProject**) et cliquer sur **Finish**.

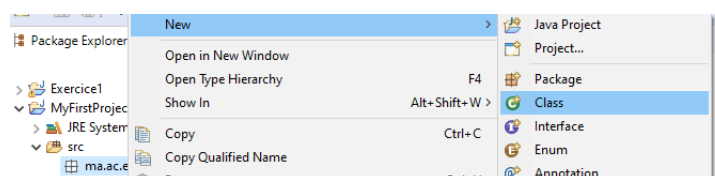
5- Création d'un package :

En Java, les classes sont organisées en package qui forment une arborescence. Un package est un ensemble de classes attachées à un même concept. Un exemple est donné par l'API Java fournie en standard on trouve le packages suivants : java.io qui regroupe toutes les classes associées aux entrées sorties. L'organisation sur disque des fichiers source doit refléter l'organisation en packages. Pour créer un package, faites clic droit sur le dossier des sources (src) , donner un nom à votre package (par exemple ma.ac.ensate.tp) et cliquer sur Finish.

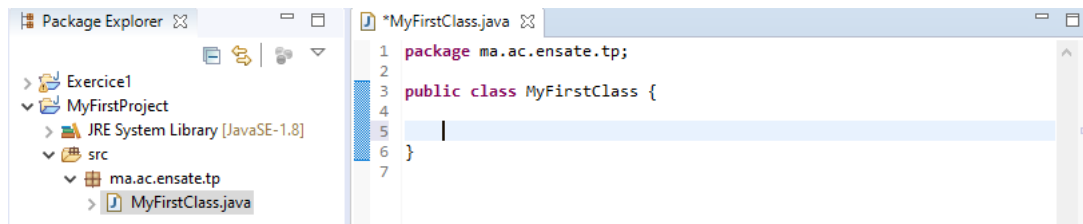
6- Création d'une classe :

Rappel sur l'architecture générale d'un programme Java : Programme source Java = ensemble de fichiers « .java » chaque fichier « .java » contient une ou plusieurs définitions de classes. Au plus une définition de classe public par fichier « .java » avec nom du fichier = nom de la classe publique.

- a) Pour créer une classe : click droite sur le package dont vous voulez ajouter la classe New ->Class.



- b) Donner un nom à votre classe et cliquer sur Finish.



7- Compilation :

Par défaut, lorsqu'un projet est créé, Eclipse active l'option de « compilation automatique ». Ainsi, votre projet est compilé en temps réel, dès que vous tapez du code et sauvegardez un fichier ou que vous en lancez l'exécution. Vous pouvez désactiver la compilation automatique dans le menu Project en décochant « **Build Automatically** ».

8- Exécution d'un programme :

- a) Ajouter la méthode main suivante dans votre classe


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

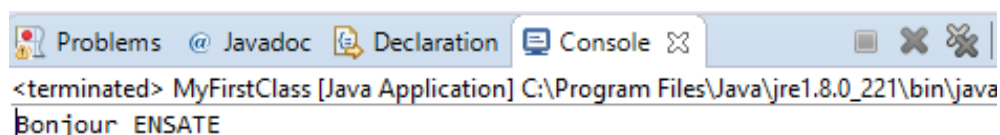
Astuce

On peut générer une méthode main avec la façon suivante :

Ecrire **main** dans le corps de la classe puis cliquer sur **CTRL+ESPACE** et puis sur la touche Entrer.

On peut générer l'instruction « **System.out.println();** » ainsi : Ecrire **Syso** puis cliquer sur **CTRL+ESPACE** . En général lorsque vous écrivez votre programme, Eclipse peut vous aider à compléter ce que vous écrivez, en utilisant le raccourci clavier **CTRL+ESPACE**

- b) Exécuter le programme en cliquant sur l'icône  de la barre d'outils ou clic droite sur la classe -> Run As -> Java Application. Le résultat d'exécution du programme s'affiche dans la console :



Raccourcis clavier indispensables:

Vous pouvez afficher la liste des raccourcis clavier pour Eclipse par CTRL + SHIFT + L

Il est possible de configurer soi-même sa combinaison pour les raccourcis clavier en allant dans le menu Window > Preferences > General > Keys.

Les raccourcis suivant sont très utiles :

CTRL + Espace : auto-complétion

CTRL + SHIFT + O : Organize Imports

CTRL + SHIFT + R : Recherche d'un fichier dans le workspace

CTRL + SHIFT + T : Rechercher une classe dans le workspace

CTRL + T : Affiche l'arborescence d'héritage de la classe courante

CTRL + SHIFT + F : Formatage du code (vous pouvez surligner une zone de code pour restreindre le formatage)

CTRL + SHIFT + I : Indentation du code (vous pouvez surligner une zone de code pour restreindre l'indentation)

CTRL + SHIFT + C : pour commenter / dé-commenter les lignes sélectionnées

CTRL + SHIFT + P : Pour se déplacer d'une accolade à l'autre

CTRL + K et CTRL + SHIFT + K : Positionne le curseur sur l'occurrence suivante ou précédente de la sélection de départ.

9- Exemples de programmes :

Réaliser sur Eclipse les programmes suivants

Exercice 1 :

Ecrire un programme qui demande à l'utilisateur de saisir deux nombre entiers et qui affiche leur somme et leur produit.

Exercice 2 :

Ecrire un algorithme qui permet de calculer la surface d'un carré.

(Rappel: Surface d'un carré = longueur*longueur).

Exercice 3 :

Ecrire un programme en java qui permet de calculer la factorielle d'un nombre saisi au clavier. Le programme doit tenir en compte que la factorielle n'est définie que sur les entiers positifs.

Exercice 4 :

Ecrire un programme qui affiche un triangle composé de caractères '*'. On demandera à l'utilisateur de rentrer le nombre de lignes. Pour chaque ligne, il faudra afficher un certain nombre d'espaces, puis un certain nombre d'étoiles. Par exemple, pour 4 lignes le résultat sera :

```
*  
  
***  
  
*****  
  
*****
```

Exercice 5 :

Ecrire un programme qui propose un menu permettant de rassembler tous les programmes précédents dans un seul programme. Le résultat sera :

- 1- Somme et produit de deux entiers
- 2- Surface d'un carré
- 3- Factorielle d'un nombre
- 4- Dessiner un triangle
- 5- Quitter le programme

Veuillez choisir un des programmes suivants :