

EXERCICE 1 : L'environnement de travail

Le but de cet exercice est de vous familiariser avec l'environnement de travail pour LOG121 :

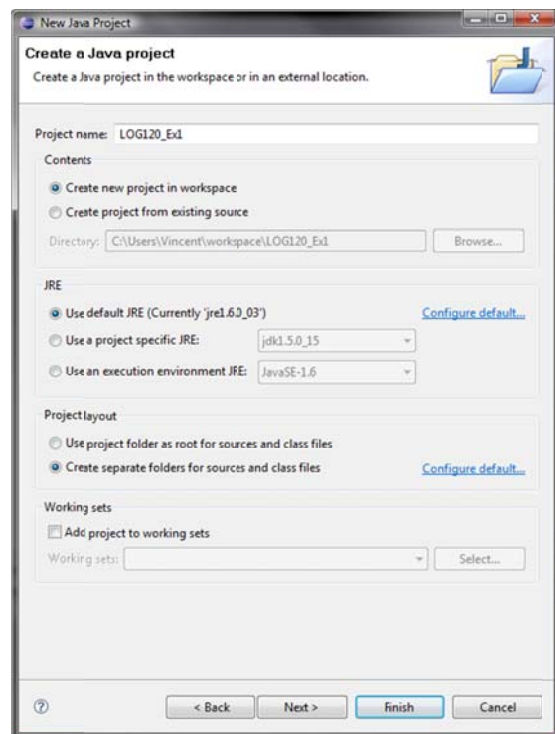
- L'environnement Eclipse pour Java;
- Le style de commentaire en-tête obligatoire pour tout fichier source;
- Le compilateur Java;
- La création d'une archive java (.jar) exécutable avec Éclipse;
- Le gabarit pour les rapports de laboratoire.

Étape 1 : Création d'un nouveau projet dans l'IDE Eclipse

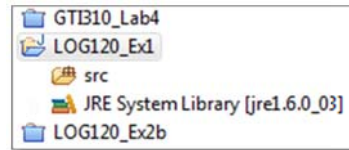
- a. Démarrez le logiciel Eclipse.

Sur les ordinateurs des laboratoires du département de Log/TI, Eclipse vous demandera l'emplacement de votre espace de travail. Conservez l'espace par défaut, sur le lecteur J: (votre espace personnel). Si votre lecteur J: est inaccessible, demandez l'aide du chargé de laboratoire ou d'un technicien.

- b. Sélectionnez le menu File / New / Project
- c. Choisissez « Java Project » dans la liste, puis cliquez sur « Next »
- d. Saisissez le nom de votre projet dans le champ « Project Name ». Dans cet exemple, le nom sera « LOG121_Ex1 »
- e. Assurez-vous que sont sélectionnées les options « Create new project in workspace », « Use default JRE » et « Create separate folders for sources and class files ».
- f. Cliquez sur « Finish »



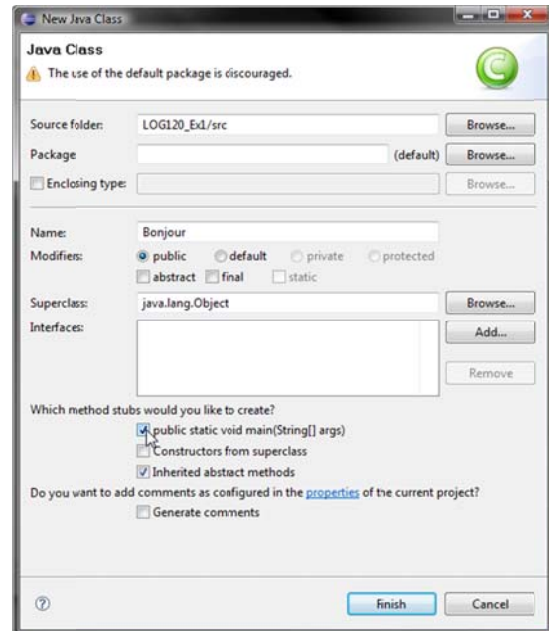
Dans le panneau de gauche, « Package », de l'environnement Eclipse, vous trouverez le projet que vous venez de créer. Si vous avez plusieurs projets dans votre espace de travail (workspace), vous les verrez tous dans ce panneau.



En suivant les étapes précédentes, vous devriez voir une structure similaire à celle-ci. Dans le répertoire src se trouveront vos fichiers source Java, classés hiérarchiquement selon la structure de packages de votre projet. Nous reviendrons sur la notion de packages au cours d'un projet de laboratoire ultérieur. On voit également un élément du projet appelé « JRE System Library ». Il s'agit de l'ensemble des classes qui composent Java. Vous avez accès à toutes ces classes dans votre projet. Les autres bibliothèques d'objets que vous importez dans votre projet, le cas échéant, seront également représentées dans cette arborescence avec la même icône.

Étape 2 : Création d'une classe dans un fichier source Java

- Cliquez sur votre projet dans le panneau Package, à gauche de l'environnement Eclipse, et sélectionnez le menu « New » / « Class ».
- Saisissez le nom de votre classe dans le champ « Name », et assurez-vous que le chemin indiqué dans le champ « Source folder » correspond à « LOG121_Ex1/src ».
- Cochez « public static void main(String[] args) ».
- Cliquez sur « Finish »



Remarquez qu'un fichier source Bonjour.java a été créé dans le répertoire « src », et du code a été généré automatiquement. En Java, le fichier qui contient le code source d'une classe doit obligatoirement porter le nom de cette classe, suivi de l'extension « .java ».

- e. Modifiez le code pour qu'il ressemble à l'exemple ci-dessous. **Attention, il faut respecter la casse, car en Java, les majuscules et les minuscules ne s'équivalent pas.**

```

/*****
Cours :          LOG121
Session :        Automne 2014
Groupe :         03
Projet :         Exercice 1
Étudiant(e)(s) : Nom, Prénom
                  Nom, Prénom
Code(s) perm. :  AAAA11111111
                  AAAA11111111
Chargé de cours : Dominic St-Jacques
Chargés de labo : Alvine Boaye Belle et Jean-Nichola Blanchet
Nom du fichier :  Bonjour.java
Date crée :       aaaa-mm-jj
Date dern. modif. aaaa-mm-jj
****
Historique des modifications
****
aaaa-mm-jj      Version initiale
*****/

/**
 * @author Nom de l'étudiant
 */
public class Bonjour {

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

        System.out.println("Bonjour tout le monde !");

    }

}

```

Cet exemple représente un fichier typique d'un projet de génie logiciel ou de génie des technologies de l'information à l'ÉTS. Il commence par un en-tête obligatoire. Tous les fichiers sources remis dans le cadre d'un travail de laboratoire dans un cours de baccalauréat du département de génie logiciel et des TI doivent obligatoirement porter cet en-tête descriptif, sans quoi il sera refusé.

Notez également les en-têtes des classes et des méthodes. Ce sont des commentaires Javadoc. Ils peuvent être générés rapidement à l'aide du raccourci clavier ALT+SHIFT+J, et permettent de générer rapidement de la documentation pour votre code. Ces commentaires font partie des exigences de tous les travaux de laboratoire du cours de LOG121.

- f. Sauvegardez ce fichier par le menu File / Save, en cliquant sur la disquette de la barre d'outils d'Eclipse (en haut), ou simplement en appuyant sur les touches CTRL+S du clavier.

Étape 3 : Compilation et exécution d'un fichier source Java

Avec l'environnement Eclipse

Sous Eclipse, par défaut, la compilation du projet se fait automatiquement. Si ce n'est pas le cas, déroulez le menu « Project » et cliquez sur « Build Project ».

Les résultats de la compilation s'afficheront dans le panneau Tasks, qui est accessible via le menu Window / Show View / Tasks.

De plus, les erreurs de syntaxe sont soulignées en rouge dans le code, et une petite ampoule apparaît dans la marge de gauche. En cliquant sur cette icône, une description de l'erreur apparaît, ainsi que des suggestions d'actions à poser pour résoudre le problème.

Pour exécuter votre programme, déroulez le menu Run de l'environnement Eclipse, puis faites Run As / Java Application. Vous devriez voir la phrase « Bonjour tout le monde ! » s'écrire dans le panneau Console, au bas de la fenêtre principale d'Eclipse.

Avec le JDK, par la ligne de commande

Bien que l'outil Eclipse facilite énormément le développement Java, il est quand même essentiel de savoir utiliser les commandes de base pour compiler un fichier Java avec le JDK de Sun. En effet, il se peut fort bien que votre programme doive être exécuté sur le poste de travail d'un développeur qui utilise un autre environnement de développement, ou même sur le poste de quelqu'un qui n'est pas un développeur. C'est cette méthode qui sera préconisée lorsque votre programme quittera l'environnement de développement, par exemple lors d'un livrable. Alors nous vous donnons des instructions ici pour ce faire.

- a. Ouvrez une fenêtre de commande MS-DOS (Démarrer / Exécuter..., entrez « CMD », puis faites la touche « entrée »)
- b. Placez-vous dans le répertoire contenant vos fichiers sources. (si vous n'en connaissez pas l'emplacement, cliquez sur votre projet dans Eclipse avec le bouton droit de la souris, cliquez sur « Properties », puis prenez en note le chemin identifié par « Location »). Placez-vous dans le répertoire src à l'aide de la commande DOS CD.
- c. Saisissez la commande `more Bonjour.java` pour visualiser le fichier que vous avez créé. (La commande `more` sert à afficher à l'écran le contenu d'un fichier texte). Si vous obtenez une erreur, vous êtes sans doute dans un mauvais répertoire.
- d. Saisissez la commande `javac Bonjour.java`. Si vous avez des erreurs de compilation, corrigez-les en utilisant un éditeur de texte ou Eclipse. Si vous avez des erreurs bizarres ou que vous ne comprenez pas, n'hésitez pas à demander conseil au chargé de laboratoire.

- e. Dès que vous avez compilé correctement votre fichier, vous devriez voir un nouveau fichier, `Bonjour.class`, dans le même répertoire. Saisissez la commande `DIR` pour confirmer que ce fichier existe.

N.B. : Il est suggéré de ne pas stocker les fichiers `.class` dans le même répertoire que les fichiers source. Typiquement, on sauvegarde les fichiers source dans un répertoire `src`, et les fichiers `.class` dans un répertoire `bin` ou `build`. Eclipse gère ceci automatiquement, mais pour le faire sous la ligne de commande, il faut utiliser un paramètre supplémentaire. On pourrait, par exemple, utiliser la syntaxe : `javac -d ..\bin Bonjour.class`, en ayant pris soin de créer à l'avance un dossier `bin` du même niveau que le dossier `src`. Pour plus d'informations sur l'utilisation de `javac`, une documentation de cet outil est disponible sur le site Web de Sun :

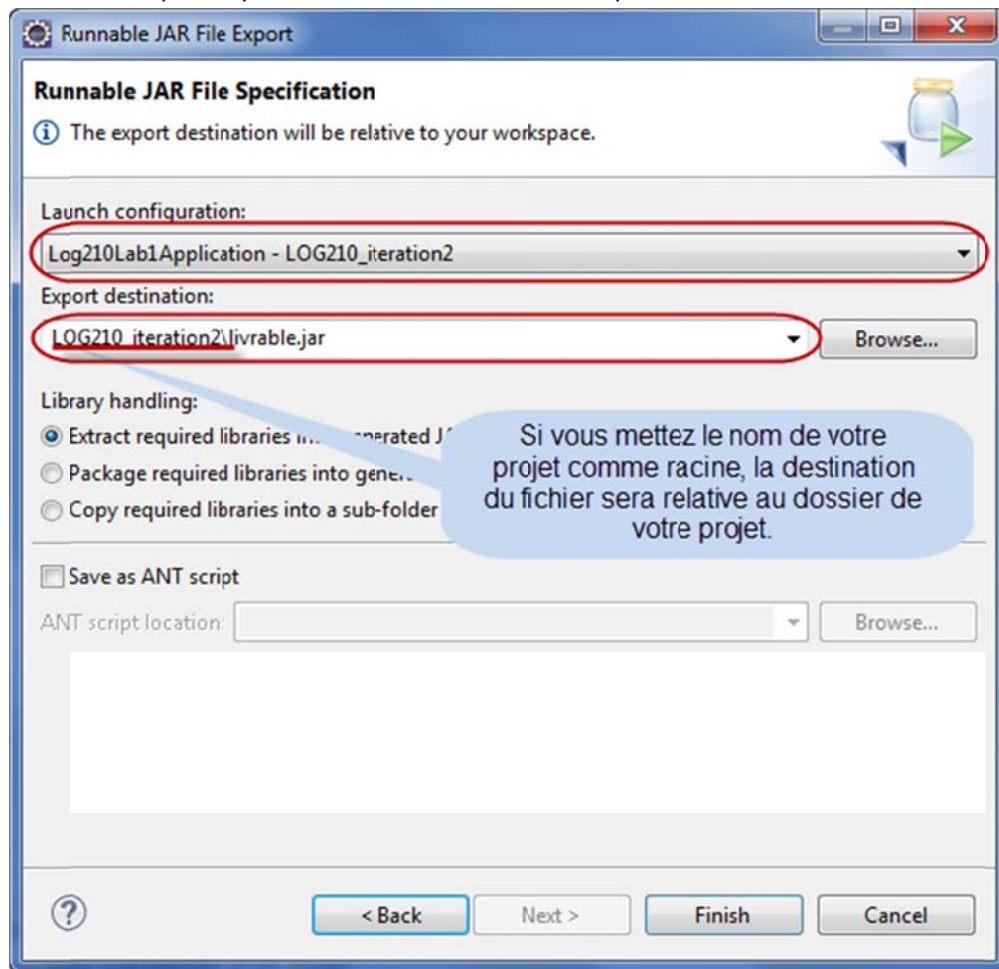
<http://download.oracle.com/javase/6/docs/technotes/tools/windows/javac.html>.

- f. À l'invite de commande, déplacez-vous maintenant vers l'emplacement de vos fichiers `.class`, si nécessaire.
- g. Saisissez la commande « `java Bonjour` » (sans extension). Vous verrez les sorties de votre programme dans la fenêtre de console.

Pour plus d'informations, sur l'utilisation de l'outil d'exécution `Java.exe` via l'invite de commande, visitez <http://java.sun.com/j2se/1.5.0/docs/tooldocs/windows/java.html>.

Étape 4 : Création d'une archive java (.jar) exécutable avec Eclipse

- a. Vous devez tout d'abord avoir une configuration d'exécution (Run configuration) pour votre projet. Il suffit d'exécuter une fois l'application pour générer une configuration.
- b. Dans le menu contextuel de votre projet (click droit sur le projet dans le navigateur d'Eclipse), vous devez choisir l'item « Export ». Vous devez ensuite choisir l'élément « Runnable Jar file » dans la section « Java ».
- c. Dans la fenêtre de configuration du fichier Jar, vous devez choisir la configuration d'exécution de votre projet. Vous devez aussi choisir la destination de votre fichier. Les autres options peuvent être laissées aux valeurs par défaut.



Étape 5 : Quelques expérimentations supplémentaires...

Avec Eclipse

- a. Si vous n'avez pas encore fait de fautes dans l'un des fichiers source Java, essayez maintenant d'introduire une erreur de syntaxe dans le programme. Par exemple, changez le mot « println » par « prntln » en supprimant le « i ».
- b. Une petite icône représentant une ampoule électrique marquée d'un « X » rouge apparaît dans la marge de gauche, et le mot incorrect est souligné en rouge.
- c. Cliquez sur l'icône d'ampoule dans la marge de gauche.

Lorsqu'on clique sur l'ampoule, Eclipse tentera de proposer une solution. On n'a qu'à cliquer sur la solution désirée, et Eclipse fait le changement automatiquement.

Eclipse vous fournit également d'autres raccourcis. Par exemple, si on tape « sysout » suivi de la combinaison de touches CTRL+ESPACE, Eclipse insère « System.out.println() » automatiquement.

Étape 6 : Documentation et remise électronique

Pour tous les laboratoires, il faut écrire un rapport. Vous devez respecter le format décrit dans le gabarit de rapport :

http://profs.etsmtl.ca/gelboussaidi/cours/LOG121/laboratoires/Gabarit_Rapport_de_laboratoire.dot

- a. Ouvrez le gabarit avec le logiciel Microsoft Word, remplissez la première page, puis sauvegardez votre rapport dans le même dossier que votre code source.
- b. Faites une remise électronique de votre rapport en suivant la procédure de remise disponible sur le site du cours.

Résumé de l'exercice

Dans cet exercice, vous avez utilisé les outils suivants :

- **Eclipse** : pour développer une application Java (création de classe en Java, rédaction de fichier source, compilation).
- **javac** MaClasse.java : pour compiler le fichier LaClasse.java -- cela produit le fichier MaClasse.class
- **java** MaClasse : pour exécuter le programme principal du fichier LaClasse.class

Vous avez également vu le gabarit pour les rapports de laboratoire, et vous avez fait une remise électronique de votre travail.

Vous êtes maintenant prêts à vous attaquer au premier laboratoire!