



L'Environnement de développement **Eclipse**



1

Qu'est ce qu'Eclipse ?



Eclipse est :

- ❖ Une plate-forme universelle pour des environnements de développement intégrés
- ❖ Fondée sur une architecture ouverte et extensible

Environnement de
développement plug-in

Outils de développement
Java

Plate-forme Eclipse

Machine virtuelle Java2



2

Propriétés de la plate-forme

C'est un logiciel

- ❖ Gratuit : licence OpenSource
- ❖ Portable : Écrit en Java

C'est un IDE

- ❖ Générique: Indépendant de tout langage
- ❖ Extensible: Par l'ajout de plug-in
- ❖ Convivial : Assistance au développeur(s)



3

Composants de la plate-forme

L'interface principale (*workbench*) manipule différents composants

1. Editeurs

- ❖ Pour des fichiers texte ou autre
- ❖ Souvent assorties de fonctions d'assistance
 - Ex: fichier source Java, diagramme de classes UML,

2. Vues

- ❖ Fournit une information sur un élément précis
- ❖ Ex: résultat de compilation, contenu d'une classe, ..

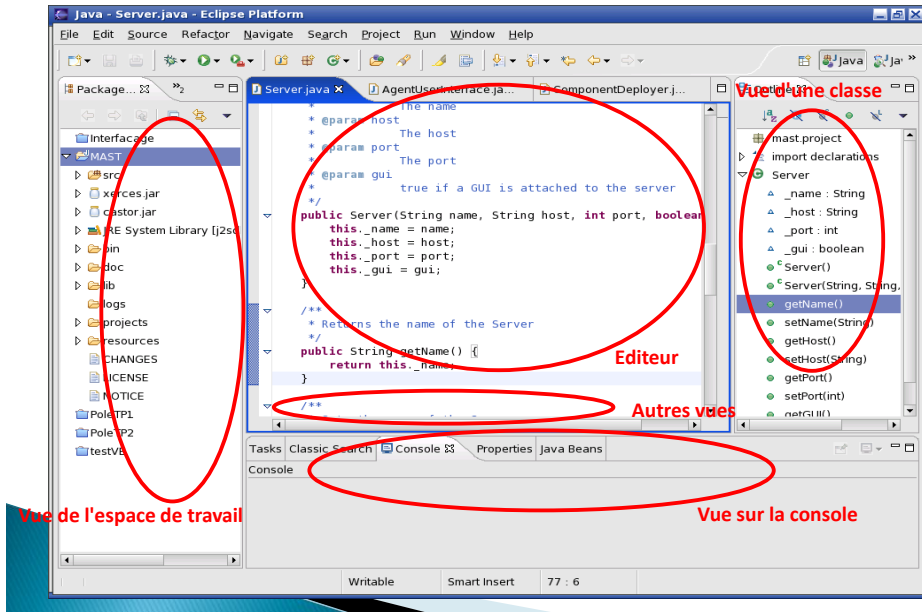
3. Perspectives

- ❖ Ensemble de vues et d'éditeurs Agencées d'une certaines manières
- ❖ Quelques perspectives standards sont fournies



4

Perspective Java

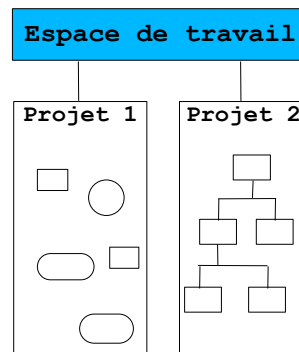


5

L'espace de travail

L'espace de travail (workspace)

- ❖ contient tous les fichiers manipulés
- ❖ autorise la création, sauvegarde, modification ou suppression de ces ressources
- ❖ est organisé en un ensemble de projets
- ❖ Correspond à un répertoire précis du disque



6

Projet



Un projet est un regroupement de ressources (fichiers, répertoires, projets)

Un projet peut être :

- ❖ Créé
- ❖ Ouvert : il est pris en compte
- ❖ Fermé : il est ignoré
- ❖ Détruit : avec ou sans ses ressources



7

Outils de développement Java



Compilateur

- ❖ compilation incrémentale de tous les projets ouverts (build)

Exécution

- ❖ différents types (application, applet, bean, ...)
- ❖ configurable (classe exécutable, classpath, ...)

Débugueur (**exemple**)

- ❖ exécution pas-à-pas ou avec des points d'arrêt (breakpoint)
- ❖ visualisation de la valeur des variables



8

Assistance de l'éditeur Java



- ❖ Complète automatiquement des identificateurs
- ❖ Détecte certaines erreurs et avertissements avant la compilation
- ❖ Propose des corrections d'erreur
- ❖ Ajoute les importations nécessaires
- ❖ Outils de travail coopératif intégré



9

Archives JAR



Java Archive

- ❖ Extension du format ZIP

Avantages

- ❖ Archivage
- ❖ Production d'exécutables portables (à condition d'avoir une JVM)
- ❖ Alternative au classique .exe
- ❖ Lancement aussi simple (double-clic, ou ligne de commande)
- ❖ Partage, ré-utilisation



10

Produire un JAR depuis Eclipse



- ❖ Click droit sur le projet, puis sur "export", et ensuite choisir "Java -> JAR file" (ou "Runnable...")
- ❖ Choisir les ressources à mettre dans le JAR
- ❖ **Bien penser à intégrer les sources si besoin !!!(cocher "Export java source files and resource")**
- ❖ Choisir les options, puis entrer l'adresse de destination du JAR et cliquer sur "Next"
- ❖ Cliquer sur "Next", puis préciser la classe contenant la méthode main
- ❖ Cliquer sur "Finish"

11

Plug-ins



- Un plug-in étend la plate-forme Eclipse pour certains types de développement
 - ❖ Téléchargeables et à installer par le menu *Help -> Software Updates -> Find and Install...*
 - ❖ Un plug-in peut en nécessiter d'autres
 - ❖ Possibilité de développer de nouveaux plug-ins
 - ❖ Un site répertorie les principaux plug-ins disponibles :

<http://eclipse-plugins.2y.net/eclipse/index.jsp>

12

Quelques raccourcis

- Code Assist (Ctrl+space)
- Quick Fix (Ctrl+1)
- Refactoring (Alt+Shift+T)
- Auto-Import (Ctrl+Shift+O)
- Surround With (Alt+Shift+Z)
- Call Hierarchy (Ctrl+Alt+H)
- Quick Type Hierarchy (Ctrl+T)
- Quick Outline (Ctrl+O)
- Show all Shortcuts (Ctrl+Shift+L)

13

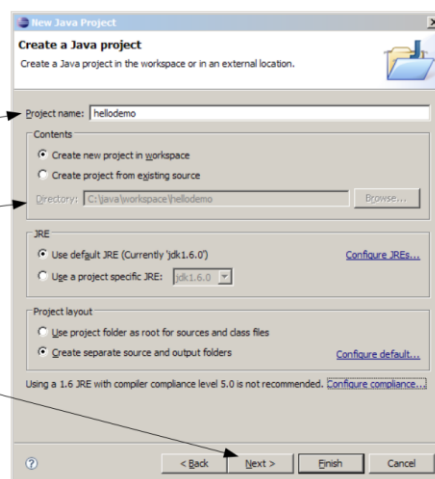
Créer un nouveau projet

- Indiquer le nom du projet

Nom du projet

Du répertoire dans le workspace

Puis **next**

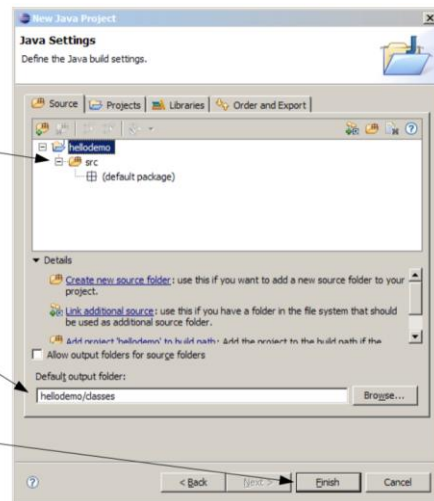


14

Créer un nouveau projet

- Les sources dans **src**
- Les classes dans **classes**

Puis **finish**



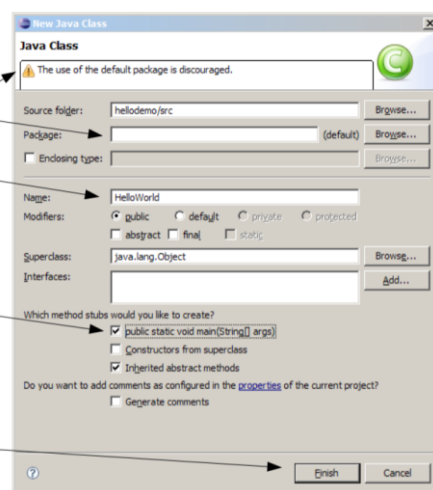
15

Nouvelle Classe

- File > New > Class
- Package aucun (mal)
- Nom de la classe

Je veux un main()

Puis **finish**

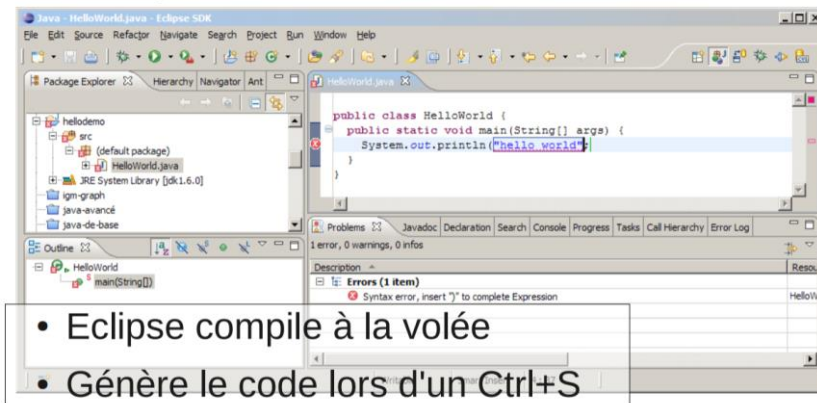


16

Et on commence à développer



- On tape le code de la classe HelloWorld



- Eclipse compile à la volée
- Génère le code lors d'un Ctrl+S

17

6) CONCLUSION



18

Merci de votre attention!

