

## TP

### Prise en main d'Eclipse

#### Partie 1 : Lancement

► Double-cliquez sur le fichier exécutable eclipse.exe présent dans le répertoire 'eclipse'.  
Eclipse se lance...



#### eclipse-lancement

Un écran apparaît ensuite pour vous demander le '**workspace**' (Répertoire de travail). C'est dans ce répertoire que eclipse stockera les programmes (\*.java) et fichiers de configuration (\*.xml, \*.properties, ...) de vos projets.

► Acceptez le répertoire par défaut, ou bien proposez **c :/JAVA/workspace**. Puis cliquez sur le bouton OK

La page d'accueil d'éclipse s'affiche.

QUESTION (Optionnel) : à quoi sert chacune des icônes de la page d'accueil ?

A NOTER : Vous pourrez à tout moment revenir sur la page d'accueil via menu help/Welcome ça y est, Eclipse est lancé. Vous êtes prêt à travailler dans de bonnes conditions.

Il vous faut cependant comprendre 2 notions : Perspective et vue

#### Partie 2 : Vues et perspectives

INFO : Une fois Eclipse installé et lancé, vous vous trouvez dans la 'perspective' JAVA.

INFO : Une perspective est en ensemble de 'vues'.

INFO : une vue est une petite fenêtre possédant des informations particulières.

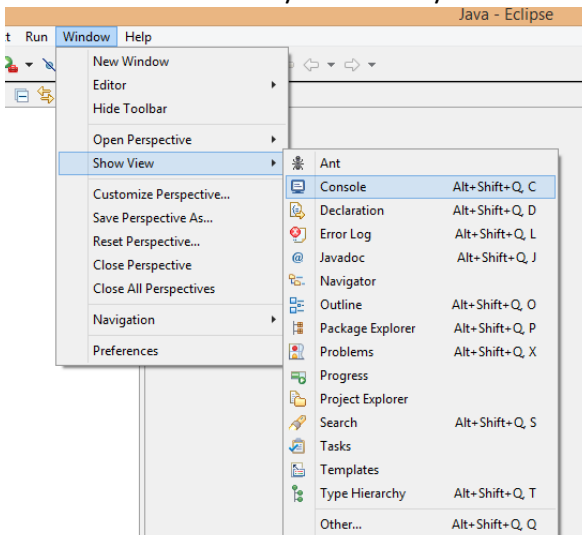
Par exemple vous pouvez voir les vues 'Problems', 'Javadoc' et 'Declaration'

Problems			
0 items			
Description	Resource	Path	Location

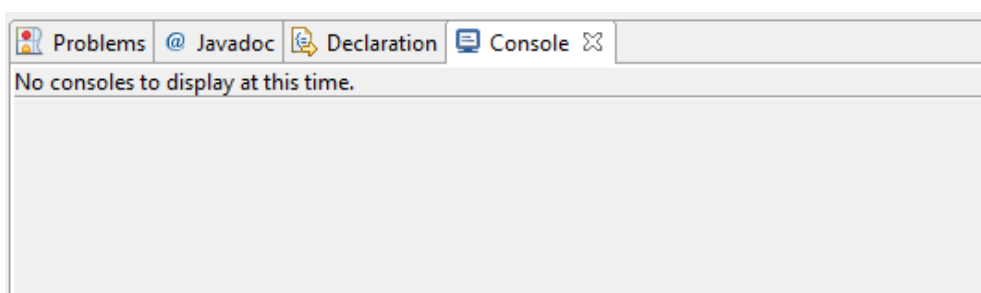
## eclipse-vues

### Vue console

► Faire menu window/show view/console. La vue console s'affiche à la suite des autres.



## eclipse-show-vues

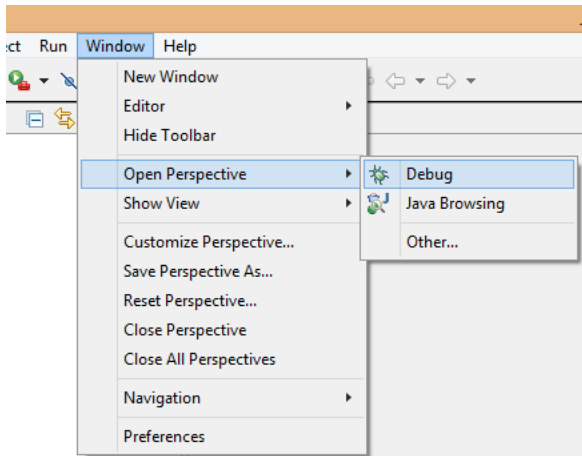


## eclipse-show-vues-console

### Perspective 'Debug'

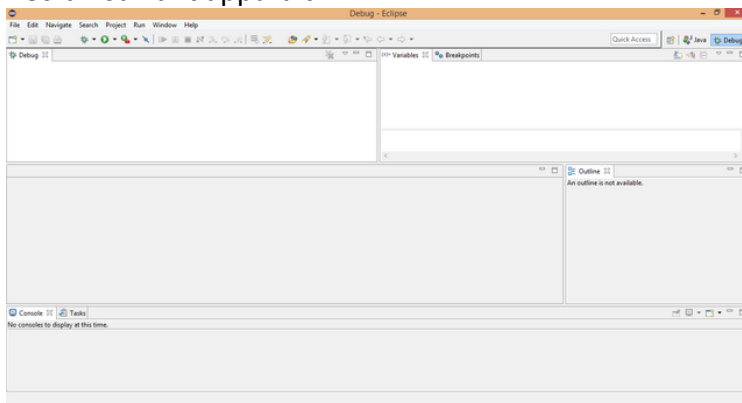
Tôt ou tard vous aurez des problèmes qui nécessiteront que vous exécutiez votre code java pas à pas. La perspective DEBUG s'y prête très bien.

► Menu Window/Open perspective/Debug .



eclipse-show-perspective-debug

L'écran suivant apparaît



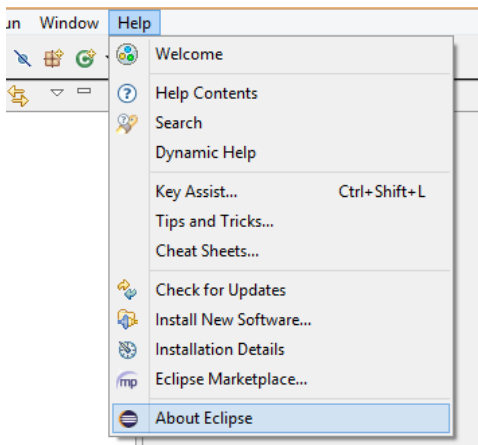
eclipse-show-perspective-debug-1

Vous remarquez alors que toute l'environnement Eclipse change de peau. Plusieurs 'vues' apparaissent, comme la vue 'debug', la vue 'variables' ou encore la vue 'Breakpoint' (points d'arrêt).

- Revenez à la perspective 'JAVA' cliquez en haut à droite sur JAVA, ou alors Menu Window/Show perspective/Java

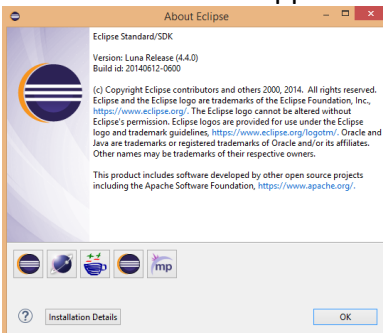
### Connaître la version d'éclipse installée

- A partir de votre interface Eclipse, faites Menu **Help/About Eclipse** (A propos d'Eclipse)



### eclipse-about

La fenêtre suivante apparaît.

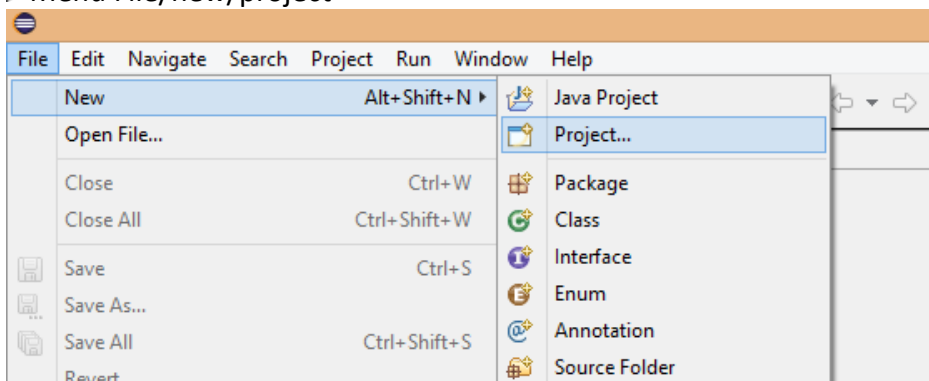


### eclipse-about-1

Comme vous le constatez, la version d'Eclipse (ici Neon.2 Release (4.6.2)) apparaît.

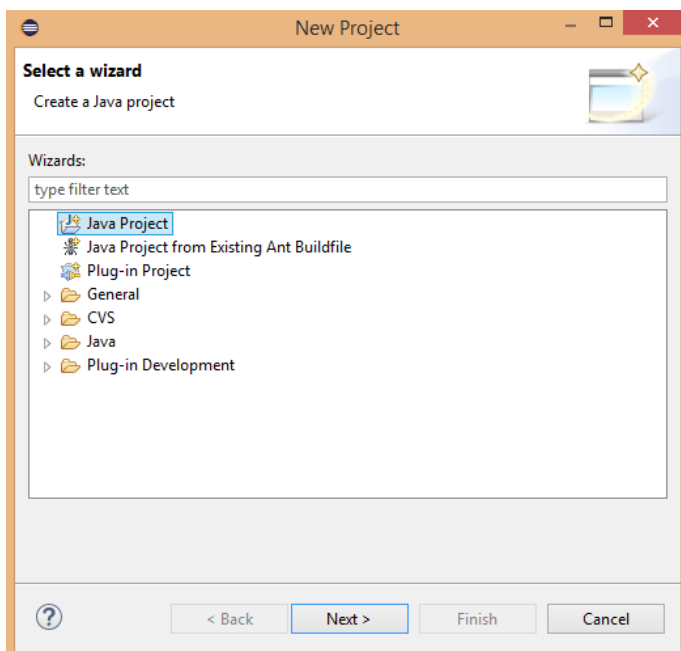
## Création projet Java

### ► Menu File/new/project



### eclipse-new-project

### ► Dans l'écran qui s'affiche, choisir Java Project



### eclipse-new-project-1

ensuite le bouton 'Next' .

► Entrez le nom du projet dans le champ 'Project name' . Ici '**TpJAVA**' .

Sélectionnez 'Use Default JRE' , pour les autres options vous pouvez ne pas toucher

REMARQUE : Eclipse par défaut va créer un répertoire spécifique (src) pour les fichiers .java, et un autre répertoire spécifique (bin) ou il mettra les fichiers compilés (\*.class)

► Cliquez sur Next

Les boutons « Add Jars... » et « Add External Jars... » permettent de référencer des librairies qui n'ont pas encore été définies dans l'environnement d'Eclipse.

Le bouton « Add Library... » permet d'ajouter une librairie déjà référencée dans l'environnement Eclipse. Lorsque l'on clique sur ce bouton, la liste des librairies disponibles apparaît. Il suffit de choisir dans la liste et de cliquer sur « Next ». Dans le cas où plusieurs versions ou options de la librairie sont disponibles, on peut choisir laquelle utiliser. Enfin, lorsque l'on clique sur « Finish », la librairie est ajoutée aux dépendances et l'on revient aux options du projet.

► Cliquez sur Finish

Enfin, lorsque l'on clique sur « Finish », le projet est effectivement créé dans le workspace et l'on revient à la vue globale d'Eclipse. Il vous faut désormais créer un 'package' Java dans lequel vous allez ranger/créer la classe principale de l'application.

► Configurer

Vous pouvez accéder aux options de configuration de votre projet par le menu « Project/Properties » ou par un clic droit sur votre projet dans l'onglet « Package Explorer ». Cette boîte de configuration vous permet de paramétrer de nombreuses options (comme la

version de Java à utiliser, les bibliothèques à inclure, etc.) que nous ne détaillerons pas ici. A vous de les découvrir au fur et à mesure de votre utilisation d'Eclipse.

### Création d'un package

- ▶ Cliquez droit sur le projet (TpJAVA) dans la vue 'project Explorer' puis new/package
- Entrez 'lanceur' comme nom de package. Puis cliquez sur le bouton Finish
- Eclipse crée alors un package
- Il vous faut désormais créer une classe principale.

### Création d'une classe

- ▶ Cliquez droit sur le package (lanceur) , puis new/class
  - ▶ Ecrire le nom de la classe , ici '**HelloMasterIT**'
  - ▶ Cochez la case 'public static void main(Strings [] args)'
- Cela aura pour effet de générer pour vous la méthode principale main() que doit avoir toute Application java

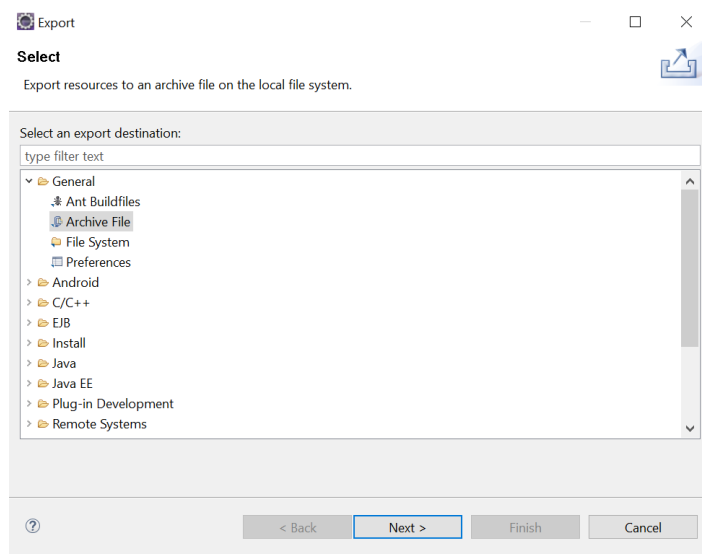
- ▶ Cliquez sur le bouton Finish
- Notez la ligne //TODO , elle permet d'écrire des tâches que vous devez faire. Le contenu texte après le 'TODO' apparaît dans la vue 'Tâches' (Window/Show view/Other/Tasks)

### Ajout de code

- ▶ Ajoutez la ligne suivante juste en dessous de la ligne // TODO
- System.out.println("Bonjour Master IT, !!!") ;**

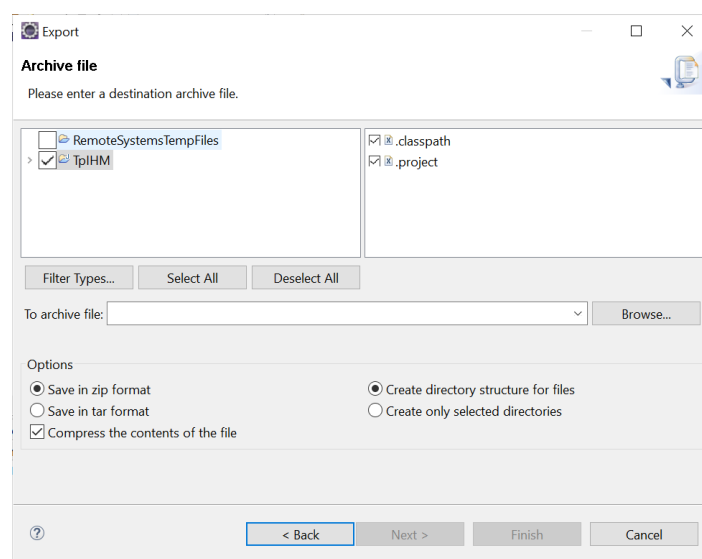
### Lancement de l'application

- ▶ cliquez-droit sur la classe **HelloMasterIT** , puis Run As / Java Application
  - ▶ Exporter/Importer un projet
- Eclipse permet aussi d'exporter et d'importer des projets complets sous la forme de fichiers d'archives, de manière à les transférer dans d'autres workspaces ou sur d'autres machine. Nous utiliserons souvent cette possibilité au cours des TP de JAVA. Vous pourrez aussi l'utiliser pour vous passer simplement vos projets entre binômes.
- Pour exporter un projet, il suffit de sélectionner le projet et de choisir le menu de la figure suivante :



apparaît alors et permet de choisir quel type d'export l'on veut faire.

Ici, nous choisissons « General/Archive File ». Après avoir cliqué sur « Next », la fenêtre de la figure suivante :



permet de choisir les options d'exportation (fichiers à exporter, nom du fichier d'export, format de l'archive). Par défaut, tout est sélectionné et c'est ce qu'il faut faire pour exporter un projet complet de manière à le ré-importer dans un autre workspace. Lorsque tout est renseigné, il suffit de cliquer sur « Finish » pour créer le fichier archive.

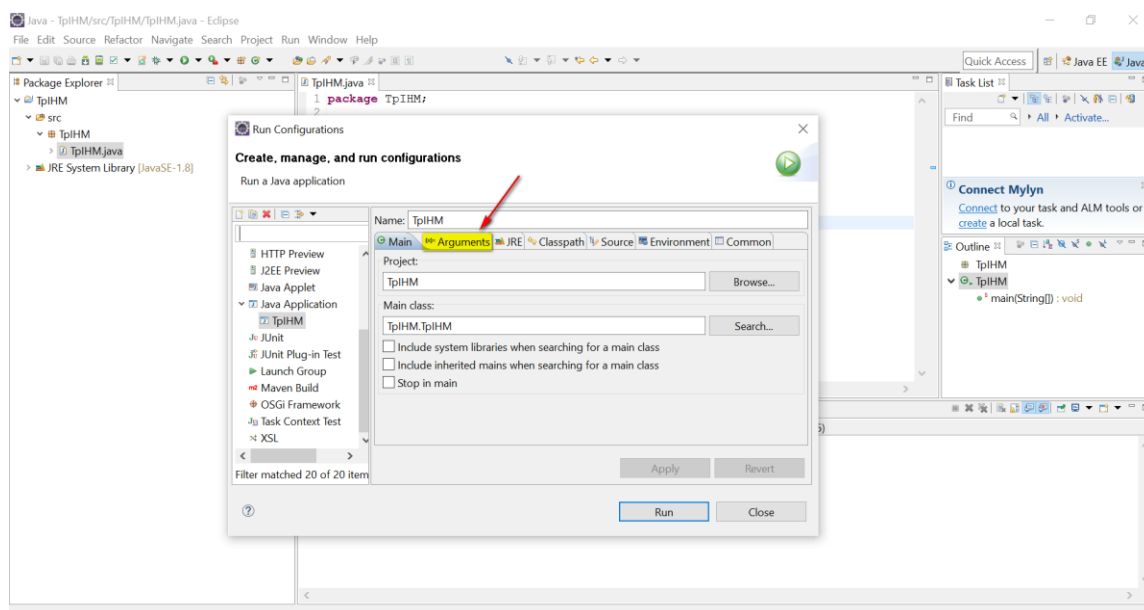
Exportez la totalité du projet créé précédemment dans un fichier « zip » nommé « TpJAVA.zip ». Ouvrez cette archive avec le navigateur de fichiers de windows pour examiner son contenu.


L'importation d'un projet s'effectue à partir du menu « File/Import... » ou du menu contextuel de l'onglet « Package Explorer ». Une fenêtre similaire à la fenêtre ci-dessus apparaît et permet de choisir le type d'import (Eclipse permet d'importer des projets entiers, mais aussi des fichiers individuels, compressés ou non, etc.). Ici, nous choisirons par contre « Existing Projects into Workspace ». Il suffit de spécifier le fichier « zip » contenant le projet à importer dans le champ

« Select archive file » et de cocher les projets à importer une fois qu'Eclipse a analysé l'archive. Si le fichier spécifié ne contient pas de projet, il n'est évidemment pas possible d'importer de projet...

Vous devrez réaliser ces opérations pour importer le projet dans votre workspace personnel.

► Pour définir des arguments à la classe, cliquer sur la flèche à côté de l'icône d'exécution et sélectionner **Run Configurations**, ou cliquer sur la classe que vous voulez exécuter, et aller à Run -> Run Configurations... La fenêtre suivante va apparaître.



► Dans la partie de gauche, sélectionner Java Application, puis cliquer sur l'icône  (en haut, à gauche), pour ajouter une nouvelle configuration. Vous verrez que votre classe HelloMasterIT a été ajoutée sous Java Application.

► Sélectionner l'onglet Arguments (ci-dessus coloré en jaune) et, dans le cadre Program Arguments, tapez simplement votre nom.

► Cliquer ensuite sur Run. Vous verrez dans la console l'affichage "**Bonjour Master IT, !!!!**".

► Modifier le programme afin d'avoir à la console Bonjour **votre\_nom** . votre\_nom est le nom écrit dans les arguments.