

– Aide mémoire NetBeansIDE –

Ceci est un aide-mémoire pour quelques manipulations standard sous NetBeans. Il se peut que certains détails varient d'une version à l'autre de NetBeans. Nous aborderons d'abord les manipulations concernant subversion (puisque vous devez l'utiliser dès le premier TP), puis nous aborderons d'autres manipulation basiques.

Fenêtre principale de l'éditeur

Il est possible, à tout moment, de rétablir l'affichage des fenêtres par défaut : `Window /Reset Windows`. Les fenêtres de base de l'Environnement de Développement Intégré (EDI) NetBeans sont les suivantes :

- La fenêtre en haut à gauche a trois onglets, nous n'utiliserons pour le moment que l'onglet `Projects`. Cette fenêtre montre l'arborescence des projets ouverts : les projet, qui contiennent différent dossiers (sources, librairies), qui contiennent des packages, qui contiennent des fichier sources (fichiers `.java`). **Double cliquer** sur un fichier pour l'ouvrir.
- La fenêtre en bas a gauche, `Navigator`, décrit la classe sélectionnée dans la fenêtre précédente de manière synthétique. Double cliquer sur une méthode ou un attribut de classe pour accéder directement au code correspondant.
- La fenêtre de droite présente les fichiers en cours d'édition.

1 Subversion sous NetBeans

Vous disposez d'une archive subversion sur le serveur `gforge.cnam.fr`. Cette archive contiendra tous vos TP, et vous pourrez également vous en servir pendant l'année. Gardez en mémoire qu'elle contiendra plusieurs répertoires (un par TP + d'autres pendant l'année).

L'utilisation de subversion est obligatoire : en particulier les TPs seront rendus via subversion. **PENSEZ À COMMITER SOUVENT** (plusieurs fois par TP). Grosso modo à chaque fois que vous avez répondu à une question, committez, ça ne coûte rien et ça vous permet de ne rien perdre (fausse manip, coupure de courant, effacement du compte `fip1` par un autre utilisateur du même compte). À la fin d'un TP, committez avant de partir afin de ne pas perdre votre travail et de permettre que la correction se fasse sur votre dernière version. Une fois que vous avez commité, effacez vos données (vous n'êtes pas le seul à utiliser cette machine avec le compte `fip1`).

1.1 Connaître votre archive sur la forge cnam

Pour connaître L'URL de votre archive subversion (configurez la forge pour être en français) :

- Allez sur le site `gforge.cnam.fr` ;
- identifiez-vous avec les identifiants fournis en TP ;
- cliquez sur “mon environnement” ;
- cliquez sur l'onglet “projet” (au milieu de la page) ;
- cliquez sur votre projet ;
- cliquez sur “SVN” dans la barre à gauche ;
- cliquez sur “information d'accès” dans la barre à gauche.

l'URL est en général de la forme : `http://gforge.cnam.fr/svn/fip_votrenom`

1.2 Récupérez une copie de votre archive via NetBeans

L'idée étant de créer vos projet dans votre copie de travail, il faut avant tout récupérer une copie de travail. Si vous en avez déjà une il n'est pas nécessaire d'en avoir une autre (vous pouvez passer à l'étape suivante). Si vous ne l'avez pas encore, voici la procédure dans NetBeans :

- assurez vous qu'aucun projet n'est surligné (clic *droit* dans une zone vide de la fenêtre de sélection des projet) ;
- menu "Team" → "subversion" → "Checkout" ;
- remplissez la casse "URL" avec l'url de votre archive ;
- remplissez les cases "User" et "Password" avec vos identifiant et mot de passe sur la forge (fournis en TP)
DÉCOCHEZ LA CASE "SAVE USERNAME AND PASSWORD" (gardez à l'esprit que vous n'êtes pas le seul à utiliser cette instance de NetBeans) ;
- cliquez sur "Next", en principe vous obtenez une nouvelle fenêtre. Ne remplissez rien à part "Local folder" : mettez un répertoire "java-XXX" où XXX est votre nom (vous reconnaîtrez votre répertoire) ;
- cliquez sur "finish". Votre copie de travail est créée, vous pouvez regarder dans le répertoire choisi plus haut ("java-XXX").
- Deux cas se présentent ici :
 - si NetBeans trouve des projets dans votre copie de travail (donc pas au premier TP), NetBeans vous demande si vous voulez y créer un projet. Répondez "close", nous le créerons plus tard (voir plus loin) ;
 - si NetBeans trouve des projets dans la copie de travail il vous propose de sélectionner ceux qu'il faut ouvrir dans NetBeans.

1.3 Commit

- *N'utilisez pas le menu Team c'est source d'erreur.*
- clic droit sur le projet ou le fichier à commiter :
"Subversion" → "Commit"
- Remplissez le message de commit et vérifier que les fichiers à commiter sont les bons.

1.4 update

- *N'utilisez pas le menu Team c'est source d'erreur.*
- clic *gauche puis droit* sur le projet à commiter :
"subversion" → "Update"
- Remplissez le message de commit et cochez les fichiers à commiter sont les bons (en principe tous, sauf si vous savez ce que vous faites).

2 Opérations de base

2.1 Créer nouveau projet

1. File / New Project
2. Sélectionner la catégorie Java et le projet type Java Application puis cliquer sur suivant.
3. Nommer le projet et choisir son emplacement sur le disque. **CHOISISSEZ UN SOUS RÉPERTOIRE DE VOTRE COPIE DE TRAVAIL.**
4. Cocher Create Main Class pour créer automatiquement une classe qui possède une méthode main.
5. Valider (Finish)
6. Une fois le projet créé et après s'être assuré qu'il avait le bon nom, le bon emplacement etc, commitez-le rapidement (voir ci-dessous).

2.2 Ouvrir un projet existant

1. File / Open Project...

2.3 Créer une nouvelle classe

Une classe est dans un fichier du même nom que cette classe.

1. Clic droit sur l'emplacement (le *package*) ou l'on veut placer la classe et New puis Java Class.
2. Nommer la classe
3. Choisir éventuellement un autre package ou un autre projet, mais laisser tout par défaut si on souhaite créer la classe dans le package sur lequel on a cliqué.

2.4 Exécuter du code

Pour exécuter du code :

- cliquer sur Run puis Run Main Project,
- ou cliquer sur la flèche verte (bouton play),
- ou touche F6,
- ou clic droit sur un projet puis Run.

Lancer le code d'un projet exécute la méthode main de la classe principale.

2.5 Créer une méthode dans une classe

1. Double cliquer sur la classe qu'on souhaite modifier (la créer si besoin, voir étapes précédentes)
2. Editer le code pour placer une nouvelle méthode, par exemple la suivante :

```
public double test(double a, double b){  
    return a*b;  
}
```

3 Tests Junit

3.1 Créer un fichier de test

Pour créer un fichier de test avec JUnit :

1. Sélectionner la classe pour laquelle on souhaite générer des tests unitaires (Fenêtre Projects en haut à gauche).
2. Cliquer sur Tools puis Create JUnit Tests.
3. Tout laisser par défaut, juste valider. Une nouvelle classe de test sera créée dans le répertoire Test Packages.
4. Editer les méthodes de test en modifiant les valeurs attendues
5. Si on modifie la classe testée, refaire la manipulation pour mettre à jour la classe de test.

3.2 Lancer un test

Sélectionner le projet à tester puis :

- cliquer sur Run puis Test Project,
- ou touches Alt+F6,
- ou clic droit sur un projet puis Test.

4 Javadoc

Javadoc permet de générer une documentation pour le code, en analysant les commentaires de vos fichiers sources. Vos commentaires sont également exploités dans NetBeans, pour afficher une documentation lorsque vous utilisez l'auto-complétion (lorsque netbeans vous propose automatiquement la suite de votre code). Pour pouvoir utiliser Javadoc, il faut respecter une certaine syntaxe dans les commentaires.

- Pour générer automatiquement une partie des commentaires, se placer au dessus du code à commenter (une méthode par exemple) et taper `/**` puis `Enter`.
- Pour générer la documentation, faire `Run` puis `Generate Javadoc`, ou clic droit sur le projet puis `Generate Javadoc`.