

## UE Conception Orientée Objet

---

### Préparation au TP Plugins

#### Préliminaire

L'objectif de ce TP est la mise en place *progressive* d'une application qui s'adapte dynamiquement en fonction de "plugins" installés dans un répertoire.

Ce TP sera l'occasion de voir la mise en place du design-pattern observer et d'utiliser certains aspects de la réflexivité. La mise en place du design pattern *Observer* est détaillée étape par étape dans le poly de cours ! Relisez donc cette partie avant le TP !

Un exemple de l'application que vous l'on va réaliser au cours de ce TP est disponible. Pour cela récupérez l'archive `fichiers-plugins.zip` disponible sur le portail et décompressez la dans votre espace de travail. Vous devez avoir maintenant un répertoire `plugins` contenant les répertoires `classes`, `disponibles`, `extensions` et `src`. Placez vous dans le répertoire `plugins` et exécutez la commande :

```
java -classpath .:classes:extensions edit extensions &
```

La fenêtre qui apparaît contient une barre de menu. L'item du menu **Outils** propose un certain nombre de fonctionnalités (une pour l'instant). Ces outils correspondent à des opérations qui s'appliquent au texte contenu dans la zone de texte constituant la partie inférieure de l'application.

Vous pouvez tester cette application en appliquant l'outil proposé sur un texte (que vous saisissez au clavier ou chargez depuis le menu **Fichier**).

Cette application a la particularité qu'il est possible de l'enrichir en lui ajoutant dynamiquement, c'est-à-dire pendant qu'elle est en cours d'exécution, de nouveaux outils. Ce sont ces outils qui vont constituer les plugins évoqués ci-dessus.

Un certain nombre de plugins sont proposés, il se trouve dans le dossier `disponibles`.

Copiez un par un les `.class` situés dans de répertoire vers `extensions`, observez ce qui se passe entre chaque copie dans le menu **Outils** et tetez chacun des nouveaux outils proposés. Vous pouvez placer dans le même dossier un autre fichier (y compris d'extension `.class`) quelconque et, normalement, rien ne se passe au niveau de l'application. Pour être reconnu comme étant un plugin un fichier doit en effet satisfaire un certain nombre de contraintes.

Il est également possible de supprimer un outil, il vous suffit pour cela d'effacer le fichier `.class` correspondant du répertoire `extensions`.

L'objectif de ce TP sera l'implémentation de cette application.

Avez-vous une idée de comment la réaliser ? Réfléchissez-y...

#### Travail préparatoire

1. testez l'application proposée
2. relire dans les notes de cours la mise en œuvre étape par étape de *Observer* proposée.
3. lire la javadoc de `javax.swing.Timer`
4. lire la javadoc de `java.io.FileNameFilter`
5. lire la javadoc de la méthode `public String[] list(FileNameFilter filter)` de la classe `java.io.File` (lire le texte d'introduction de la javadoc de cette classe est certainement utile également).
6. Etudier les documentations des classes `JMenu`, `JMenuItem` et `JMenuBar` du paquetage `javax.swing`. Vous pouvez notamment jeter un œil au lien "*How to Use Menus*".