
UE Conception Orientée Objet

TP 0

Les manipulations à réaliser, sans difficulté de conception objet ou de langage Java, ont pour objectif de valider l'usage des outils utilisés en COO, gitlab et maven. Ces outils sont présentés dans le document fourni. Vous validez ainsi votre bonne compréhension de ce document.

Vous pouvez en plus utiliser Eclipse pour réaliser ce travail et ainsi vous familiariser avec cet IDE.

Il est rappelé que pour accéder au gitlab depuis l'extérieur du campus vous devez avoir configuré le VPN sur votre machine. Voir <http://fil.univ-lille1.fr/documentation-vpn>

Vous devez dans cet exercice :

1. créer un dépôt sur gitlab pour vos projets de COO, vous y placerez les différents travaux de COO qui seront à rendre au cours du semestre.

Vous ne devez créer qu'un dépôt pour les deux membres du binôme. N'oubliez pas d'ajouter votre enseignant en tant que `Developer` à votre projet.

N'oubliez pas de créer à la racine de ce dépôt un fichier `readme.md` qui précisera les noms des deux binômes ainsi que d'autres informations que vous jugerez utiles (liste des projets par exemple).

Pensez à configurer le fichier `.gitignore` pour votre dépôt.

2. utiliser Maven pour créer dans le dossier qui correspond à votre dépôt un projet Java que vous appellerez `TPO` (vous pouvez ou non conserver le `groupId` à `fil.coo`).

N'oubliez pas de placer un fichier `readme.md` à la racine de ce projet et de le rédiger pour qu'il décrive ce projet "TPO".

3. coder dans votre projet une classe `Student.java`. Un objet `Student` aura deux attributs : un nom et un numéro d'étudiant (deux chaînes de caractères) et des getter et setters adaptés, ainsi qu'une méthode `equals`.

Evidemment vous rédigerez la javadoc de cette classe.

Pensez à faire des *commits* réguliers pour les différentes étapes de votre projet.

4. coder une classe de test `JUnit4` pour cette classe `Student`.
5. coder une classe `MainStudent` qui possède une méthode `main` qui crée deux objets étudiants et affiche leurs noms et numéro d'étudiant.
6. configurer Maven pour qu'à l'exécution de `mvn package` les tests soient exécutés, une archive exécutable soit produite avec pour point d'entrée `MainStudent` et la javadoc soit générée.
7. déposer l'ensemble de votre travail sur gitlab avant la date d'échéance communiquée par votre enseignant.

Pour rappel votre enseignant exécutera les commandes :

```
git pull
```

suivie de

```
mvn package
```

Tout doit se dérouler sans erreur. On doit pouvoir ensuite consulter la documentation et utiliser le jar produit.