

MAVEN – TP 02

OBJECTIF

- Se familiariser avec l'utilisation des plugins Maven

REECRITURE DES METHODES EQUALS ET HASHCODE

- La librairie "commons-lang" offre des outils pour redéfinir plus simplement les méthodes equals et hashCode.
- Voir :
 - <https://commons.apache.org/proper/commons-lang/apidocs/org/apache/commons/lang3/builder/EqualsBuilder.html>
 - <https://commons.apache.org/proper/commons-lang/apidocs/org/apache/commons/lang3/builder/HashCodeBuilder.html>
- Réécrire les les méthodes equals et hashCode avec cette librairie.

CREER UN JAR EXECUTABLE

- Générer un package avec Maven
 - mvn clean package
- Exécuter le JAR avec la commande
 - java -jar target/pizzeria-console.jar
 - une erreur s'affiche, le JAR généré n'est pas exécutable
- Configurer le plugin « maven-jar-plugin » pour qu'il mette à jour l'attribut « Main-Class » du fichier META-INF/MANIFEST.MF généré pour le JAR.
- Exécuter le JAR avec la commande
 - java -jar target/pizzeria-console.jar
 - Le programme se lance mais il y a une erreur NoClassDefFoundError
- Utiliser le plugin « maven-shade-plugin » pour créer un « uber-jar » qui contient toutes les classes compilées du projet (y compris celles des dépendances).
- Exécuter le JAR avec la commande
 - java -jar target/pizzeria-console.jar
 - Le programme fonctionne correctement.

GENERER UN JAR SOURCES

- Utiliser le plugin “maven-source-plugin” pour qu’un JAR de sources du projet soit généré lors de la commande “mvn package”.
- Documentation du plugin : <https://maven.apache.org/plugins/maven-source-plugin/>

[OPTIONNEL] CONSTRUCTION AUTOMATISEE DU MENU

- Créer une annotation @OptionsMenu qui sera positionnée sur les options de menu à afficher.
- Modifier le projet “pizzeria-console” pour que le menu soit automatiquement construit en parcourant les classes filles de la classe “AbstractOptionsMenu” et en prenant en compte uniquement celles qui sont annotées avec @OptionsMenu.
- Utiliser la librairie “Reflections” (<https://github.com/ronmamo/reflections>)