1. Qu’est-ce que Maven ?

Maven est un outil de gestion pour les dépendances de projets Java. Il permet notamment d’inclure facilement des bibliothèques à un projet Java. Il suffit de référencer les dépendances nécessaires dans la pom.xml et Maven les téléchargera et les intégrera automatiquement au projet.

1. Quelle est l’arborescence d’un projet d’une application web utilisant Maven ?

Une image contenant texte

Description générée automatiquement

Vous trouverez ci-dessus l’arborescence d’un projet appli web avec Maven (fraichement créé) pour mieux imager la chose.

Dans l’arborescence de notre projet nous trouvons :

Un dossier java qui va contenir nos classes

Un dossier resources ou nous pourront mettre les ressources

Un dossier webapp qui contient notre web.xml

Un dossier test qui contient les tests du projet

Un dossier target qui contient les fichiers de classes compilés

Et évidemment le fichier pom.xml où nous référençons les plugins, les dépendances etc. que nous souhaitons installer grâce à Maven

(On peut d’ailleurs apercevoir les dépendances déjà installées grâce à Maven dans l’arborescence « External Libraries »)

1. Indiquer les 3 attributs nécessaires pour décrire un artefact Maven

Une image contenant texte

Description générée automatiquement

Vous trouverez ci-dessus la déclaration de 3 dépendances/artefacts Maven dans la pom.xml. Chaque artefact Maven est unique et est défini par un ID de groupe, un ID d’artefact et un numéro de version.

1. Citer 2 autres outils « concurrents » de Maven

Un des plus grands concurrents de Maven est Gradle qui est un moteur de production multi-language, alors que Maven se concentre uniquement sur les projets Java et Java EE.

Comme autre « concurrent » je citerais Jenkins, qui est un outil d’intégration continue et qui permet aussi d’exécuter des projets basés sur Maven.

1. Qu’est-ce que le « TDD » ?

Le TDD où Test Driven Development est un modèle de développement de Logiciel qui à pour but d’augmenter la qualité du logiciel et sa de réduire le temps et les frais de maintenance. Elle est caractérisée par 3 phases de développement :

1. La phase rouge où on développe les tests pour un système qui lui n’a pas encore été développé en se mettant à la place de l’utilisateur final. Cette phase permet de se rendre compte des fonctionnalités qui sont réellement indispensable.
2. La Phase verte où on imagine que le test échoue et où on se met de nouveau à la place du programmeur pour trouver une solution simple et efficace à ce problème.
3. La phase de refactoring où on revoie son code pour le structurer proprement et le rendre lisible et compréhensif pour les autres développeurs.
4. Quels sont les principes « SOLID » ?

SOLID regroupe 5 principes de développement ayant pour but de d’augmenter la maintenabilité de l’architecture logiciel du programme développé et de le rendre plus compréhensif et plus facile à développer dans le futur.

Les 5 principes sont les suivant :

Single Responsibility principe : une classe ou méthode ne doit avoir qu’une seule et unique responsabilité.

Open/closed principle : Une classe, méthode ou fonction ne doit pas être fermée à la modification mais ouverte au développement.

Substitution de Liskov : une instance A doit être remplaçable par une instance B tel que B est un sous-type de A mais cela ne doit pas casser la cohérence du code.

Interface segregation principle : Il est préférable de développer une interface pour chaque client plutôt que d’avoir une énorme interface générale.

Dependency inversion principle : Les modules de haut niveau ne doivent pas dépendre des modules de bas niveau. Chaque module doit dépendre d’abstractions pour éviter de créer des dépendances qui peuvent se révélé nuisible à l’intégrité du code.

1. Citer 3 serveurs d’applications java

Apache Tomcat, JBoss et GlassFish sont des serveurs d’application compatible avec Java.

1. Dans quel fichier se trouve la configuration des servlets d’une application web ?

Une image contenant texte

Description générée automatiquement

Ils se trouvent dans le fichier web.xml qui lui se trouve dans src/main/webapp/WEB-INF.

1. Quelles sont les 3 directives spécifiées par les spécifications des JSP ? (les citer et indiquer leurs rôles)

La directive « page » qui définit les options de configuration.

La directive « include » qui permet d’inclure des fichiers dans la JSP avant de générer la servlet.

La directive « taglib » qui permet de créer des tags personnalisés.

Docker

1. Donnez une rapide définition de Docker

Docker est un magnifique logiciel permettant de créer un conteneur pour un application et donc de lancer cette application sur n’importe quel serveur.