**MAVEN:**

Pre-requisites:

1. Verify JDK is installed, if not then Install JDK and setup JAVA\_HOME environment variable
2. Verify Maven is installed, if not the download maven from <http://maven.apache.org> and setup
3. Use any IDE of your choice (eclipse is considered for this assignment but any other IDE is also just fine)
4. Configure your IDE to use the external maven installation
5. Use Tomcat for web project assignment
6. Basic Command line Project: Do not use any IDE for this activity. Create a basic Maven quickstart project from the command line using the below parameters.
   1. Group: com.wipro.topgear
   2. Artifact: basicMaven
   3. Archetype ArtifactId: maven-archetype-quickstart

Examine the generated project structure and create one Main class and one Test class. The main class will jest print the message “TopGear-Standalone Maven”.

Build the project using Maven from command line.

Run the main class using Maven from command line.

Run the test classes using Maven from command line.

Create/Update a README file in the project root folder and specify the steps to execute the main project & execute the test class. Also document your learnings from this assignment.

1. Basic Maven Project: Using Maven command line create a new project as specified in Assignment#1 but with different artifact name. Open the eclipse IDE and import this project (Hint - Use import as “maven project”). The goal is to write a simple standalone java program that uses apache-commons-lang api’s StringUtils class to perform some operations (You are free to write any few lines of code using StringUtils class). Do not hard code the version numbers for any libraries in the dependency section of pom file, instead define a common “property” for each library and use it.

Using Maven, build the project (Make sure there are no build errors), Run the main class, Run the test class.

Create/Update a README file in the project root folder and specify the steps to execute the main project & execute the test class. Also document your learnings from this assignment.

1. Basic Maven Project - Plugins: Create a new maven quick start standalone project using IDE. Create a main class that performs some operations using apache-commons-lang library, also add some sample data (any kind of data) into database (in-memory) and print data from database. So, when you run the main class it should insert some data into database and read the same data and prints to the console. You can use any in-memory databases for this action and add corresponding libraries (Hint – Derby, HSQL, etc.,). Use the below maven plugins and make them execute at “package” stage

maven-compiler-plugin

maven-shade-plugin

Build the project using “mvn clean package” from console

Run the main class using maven.

Create/Update a README file in the project root folder and specify the steps to execute the main project & execute the test class. Also document your learnings from this assignment.

1. Maven Web Project: Create a maven web project using the IDE. Create a simple index.jsp page with some static message. Build and deploy the artifact on Tomcat. Make sure the index page is displayed properly.

Create/Update a README file in the project root folder and specify the steps to execute the main project & execute the test class. Also document your learnings from this assignment.

1. Maven Profiles Project: Create a basic maven web application with simple index.jsp page. Make sure your project structure follows standard Maven structure (Ex: src/main/java; src/main/resources; src/test/java etc.,). Create a folder “env” under src/main and create sub-folders under src/main/env with name “dev\_webapp” and “itg\_webapp”. Create a properties file called “myapp.properties” in both the folders with the below data.

dev\_webapp/myapp.properties:

msg=Hello I am executing in DEV

itg\_webapp/myapp.properties:

msg=Hello I am executing in ITG environment

Your index.jsp page should display the message from “myapp.properties” file.

Create two profiles called “dev” and “itg” in project’s pom file.

Build the project with “itg” profile and deploy the application, and see what message comes.

Build the project with “dev” profile and deploy the application, and see what message comes.

If you don’t supply any profile name, maven should take “dev” as default.

(Tip: Use maven’s -P parameter for supplying profile name from command line)

Create/Update a README file in the project root folder and specify the steps to execute the main project & execute the test class. Also document your learnings from this assignment.