

Eclipse provides an excellent plugin [m2eclipse](#) which seamlessly integrates Maven and Eclipse together.

Some of features of m2eclipse are listed below

- You can run Maven goals from Eclipse.
- You can view the output of Maven commands inside the Eclipse using its own console.
- You can update maven dependencies with IDE.
- You can Launch Maven builds from within Eclipse.
- It does the dependency management for Eclipse build path based on Maven's pom.xml.
- It resolves Maven dependencies from the Eclipse workspace without installing to local Maven repository *requires dependency project be in same workspace.*
- It automatic downloads required dependencies and sources from the remote Maven repositories.
- It provides wizards for creating new Maven projects, pom.xml and to enable Maven support on existing projects
- It provides quick search for dependencies in remote Maven repositories

Installing m2eclipse plugin

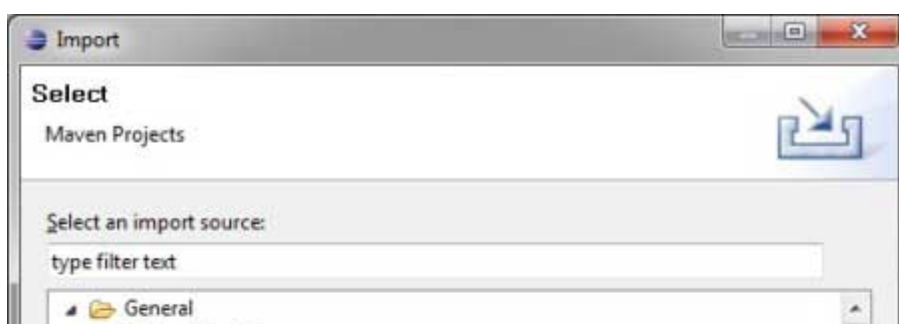
Use one of the following links to install m2eclipse:

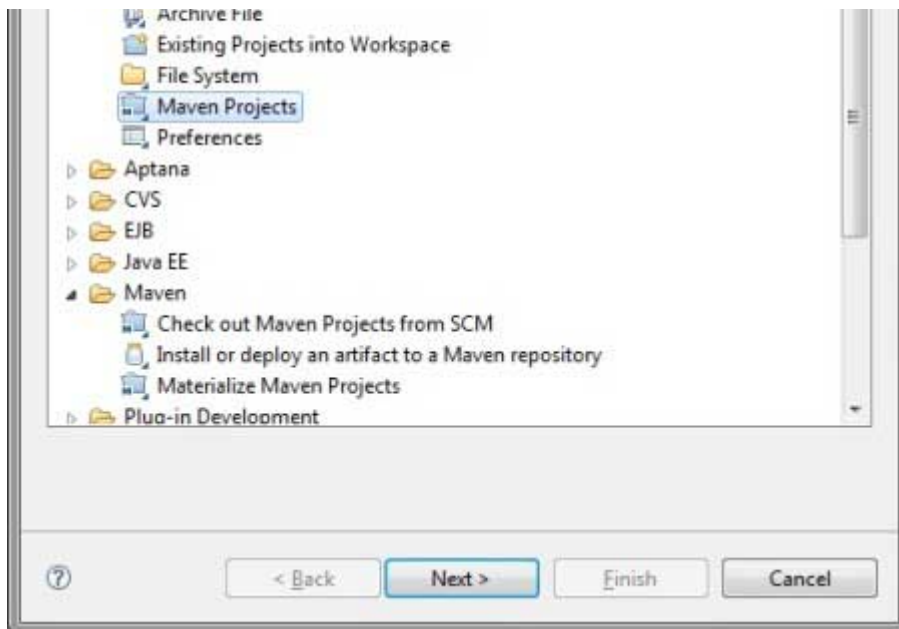
Eclipse	URL
Eclipse 3.5 <i>Gallileo</i>	Installing m2eclipse in Eclipse 3.5 <i>Gallileo</i>
Eclipse 3.6 <i>Helios</i>	Installing m2eclipse in Eclipse 3.6 <i>Helios</i>

Following example will help you to leverage benefits of integrating Eclipse and maven.

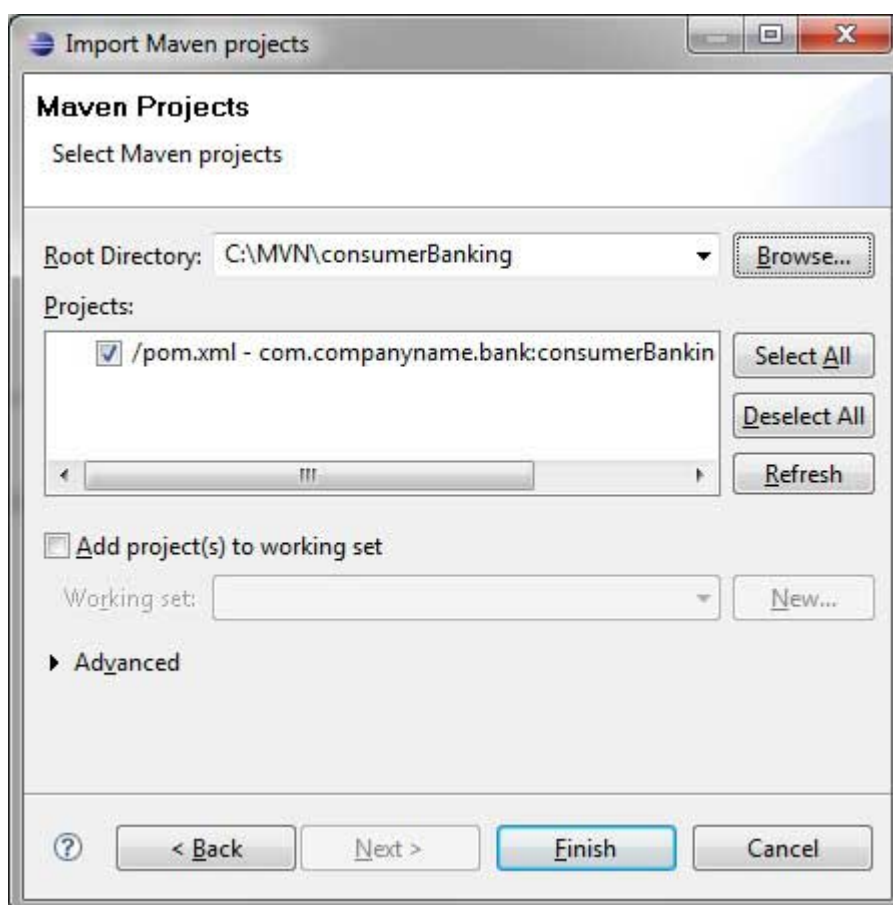
Import a maven project in Eclipse

- Open Eclipse.
- Select **File > Import >** option.
- Select Maven Projects Option. Click on Next Button.



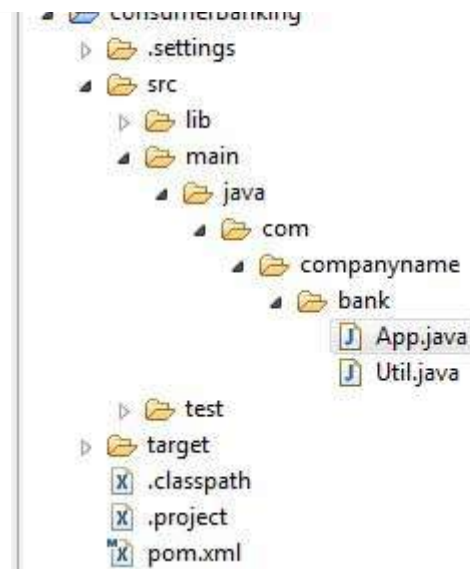


- Select Project location, where a project was created using Maven. We've create a Java Project consumerBanking. See [Maven Creating Project](#) to see how to create a project using Maven.
- Click Finish Button.

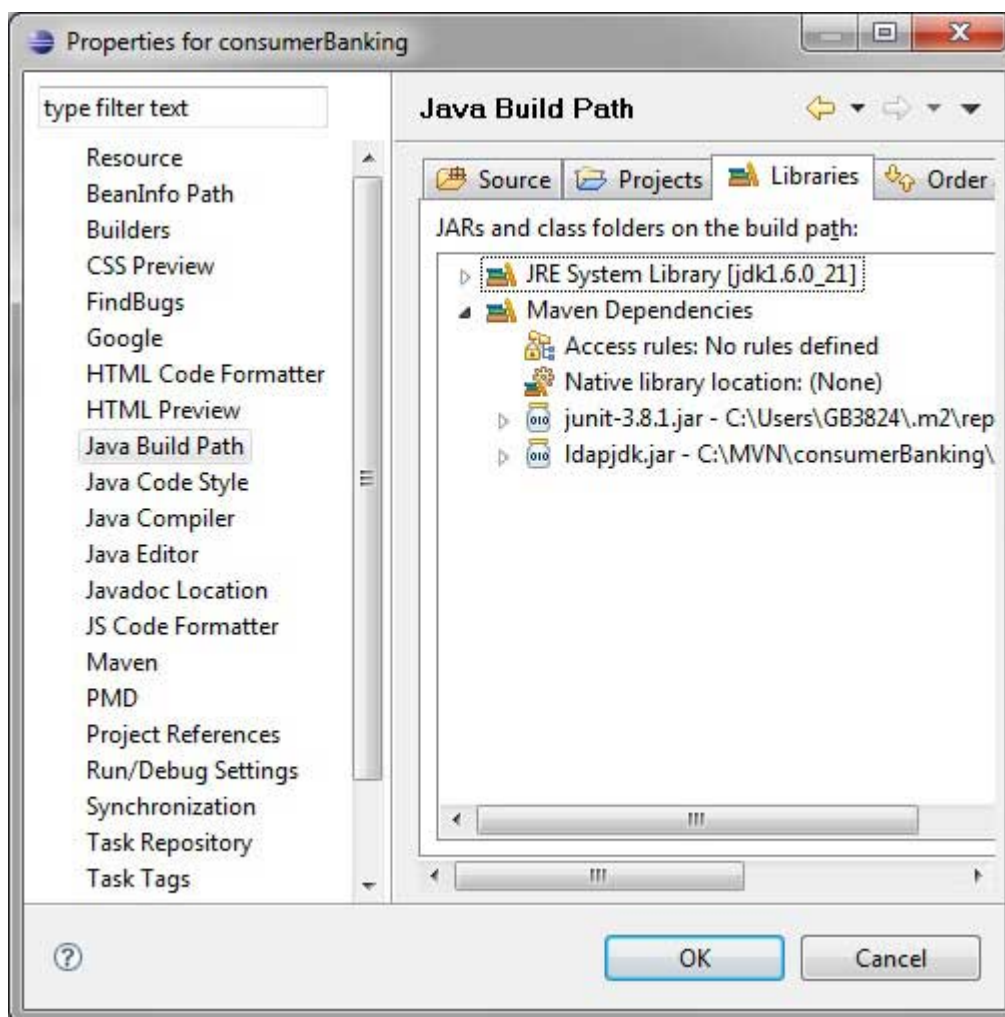


Now, you can see the maven project in eclipse.





Now, have a look at consumerBanking project properties. You can see that Eclipse has added Maven dependencies to java build path.



Now, Its time to build this project using maven capability of eclipse.

- Right Click on consumerBanking project to open context menu.
- Select Run as option
- Then maven package option

Maven will start building the project. You can see the output in Eclipse Console

```
[INFO] Scanning for projects...
[INFO]
```

```

[INFO] -----
[INFO] Building consumerBanking 1.0-SNAPSHOT
[INFO] -----
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ consumerBanking ---
[INFO] Deleting C:\MVN\consumerBanking\target
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ consumerBanking ---
[WARNING] Using platform encoding (Cp1252 actually) to copy filtered resources, i.e.
build is platform dependent!
[INFO] skip non existing resourceDirectory C:\MVN\consumerBanking\src\main\resources
[INFO] --- maven-compiler-plugin:3.1:compile (default-compile) @ consumerBanking ---
[INFO] Changes detected - recompiling the module!
[WARNING] File encoding has not been set, using platform encoding Cp1252, i.e. build is
platform dependent!
[INFO] Compiling 1 source file to C:\MVN\consumerBanking\target\classes
[INFO] --- maven-resources-plugin:2.6:testResources (default-testResources) @
consumerBanking ---
[WARNING] Using platform encoding (Cp1252 actually) to copy filtered resources,
i.e. build is platform dependent!
[INFO] skip non existing resourceDirectory C:\MVN\consumerBanking\src\test\resources
[INFO] --- maven-compiler-plugin:3.1:testCompile (default-testCompile) @ consumerBanking
---
[INFO] Changes detected - recompiling the module!
[WARNING] File encoding has not been set, using platform encoding Cp1252, i.e. b
uild is platform dependent!
[INFO] Compiling 1 source file to C:\MVN\consumerBanking\target\test-classes
[INFO] --- maven-surefire-plugin:2.12.4:test (default-test) @ consumerBanking ---
[INFO] Surefire report directory: C:\MVN\consumerBanking\target\surefire-reports

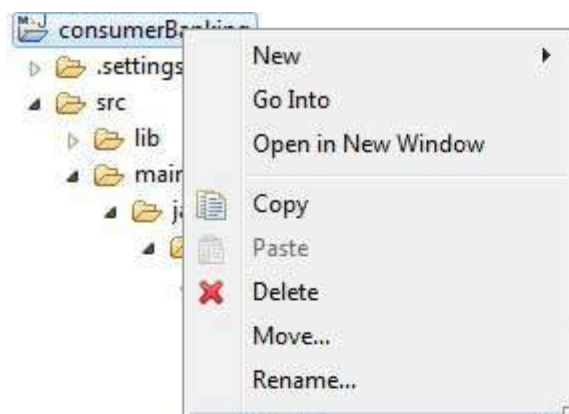
-----
T E S T S
-----
Running com.companyname.bank.AppTest
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.063 sec

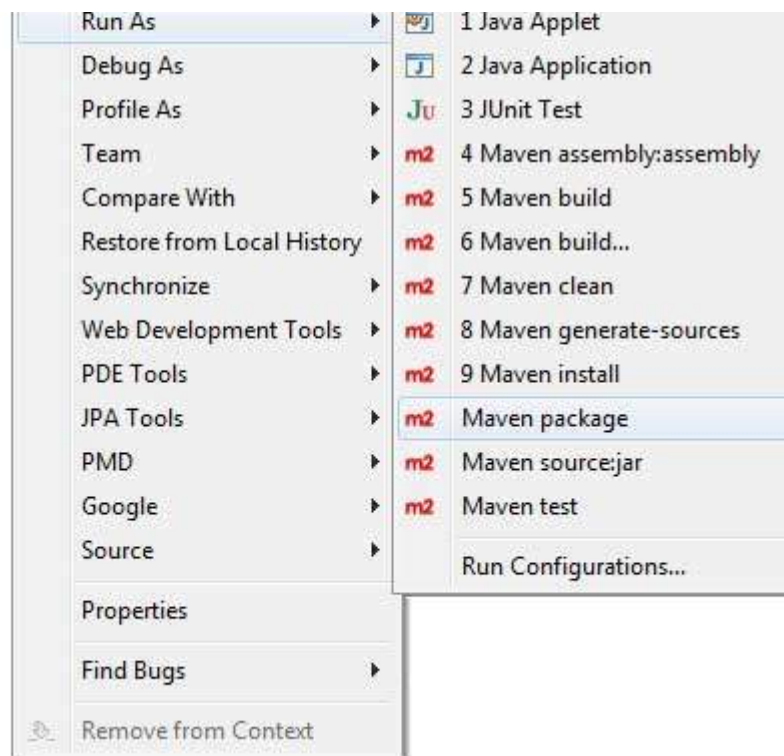
Results :

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0

[INFO]
[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ consumerBanking ---
[INFO] Building jar: C:\MVN\consumerBanking\target\consumerBanking-1.0-SNAPSHOT.jar
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 14.406 s
[INFO] Finished at: 2015-09-27T17:58:06+05:30
[INFO] Final Memory: 14M/247M
[INFO] -----

```





Now, right click on App.java. Select Run As option. Select As Java Application.

You will see the result

Hello World!

Loading [MathJax]/jax/output/HTML-CSS/jax.js