

Belajar JDBC dengan menggunakan Maven

1. Buat **database** dengan nama **belajar**

```
mysql> create database belajar;
```

2. Kemudian buat **table** dengan nama **mahasiswa**

```
mysql> use belajar;  
mysql> create table mahasiswa (nim varchar(12) primary key, nama varchar(50)) Engine=InnoDB;
```

3. Kemudian buat project menggunakan **maven** dengan nama **belajar.jdbc**

```
izwan@laptop:/tmp/belajar-maven-jdbc$ mvn archetype:create -DgroupId=belajar.jdbc \  
-DartifactId=my-app
```

4. Jika proses pembuatan project berhasil maka tampilan layar kurang lebih seperti ini :

```
[INFO] Scanning for projects...  
[INFO] Searching repository for plugin with prefix: 'archetype'.  
[INFO] -----  
[INFO] Building Maven Default Project  
[INFO] task-segment: [archetype:create] (aggregator-style)  
[INFO] -----  
  
Downloading: http://repo1.maven.org/maven2/org/apache/maven/wagon/wagon-file/1.0/wagon-file-  
1.0.pom  
  
Downloading: http://repo1.maven.org/maven2/org/apache/maven/wagon/wagon-providers/1.0/wagon-  
providers-1.0.pom  
  
Downloading: http://repo1.maven.org/maven2/org/apache/maven/wagon/wagon/1.0/wagon-1.0.pom  
  
Downloading: http://repo1.maven.org/maven2/org/apache/maven/wagon/wagon-provider-  
api/1.0/wagon-provider-api-1.0.pom  
  
Downloading: http://repo1.maven.org/maven2/org/codehaus/groovy/groovy/1.8.3/groovy-1.8.3.jar  
  
[INFO] [archetype:create {execution: default-cli}]  
[WARNING] This goal is deprecated. Please use mvn archetype:generate instead  
[INFO] Defaulting package to group ID: belajar.jdbc  
[INFO] artifact org.apache.maven.archetypes:maven-archetype-quickstart: checking for updates  
from central  
[INFO] -----  
[INFO] Using following parameters for creating project from Old (1.x) Archetype: maven-  
archetype-quickstart:RELEASE  
[INFO] -----  
[INFO] Parameter: groupId, value: belajar.jdbc  
[INFO] Parameter: packageName, value: belajar.jdbc  
[INFO] Parameter: package, value: belajar.jdbc  
[INFO] Parameter: artifactId, value: my-app  
[INFO] Parameter: basedir, value: /tmp/belajar-maven-jdbc  
[INFO] Parameter: version, value: 1.0-SNAPSHOT  
[INFO] project created from Old (1.x) Archetype in dir: /tmp/belajar-maven-jdbc/my-app  
[INFO] -----  
[INFO] BUILD SUCCESSFUL  
[INFO] -----  
[INFO] Total time: 1 minute 9 seconds  
[INFO] Finished at: Sun Mar 18 10:52:37 WIT 2012  
[INFO] Final Memory: 13M/142M  
[INFO] -----  
izwan@laptop:/tmp/belajar-maven-jdbc$
```

5. Buka file **pom.xml** dan tambahkan dependency untuk **mysql-jdbc-driver**, cari konfigurasi **pom.xml** untuk mysql-connector-java di <http://search.maven.org/>

```
<dependency>
  <groupId>mysql</groupId>
  <artifactId>mysql-connector-java</artifactId>
  <version>5.1.18</version>
</dependency>
```

6. Kemudian lakukan compile dengan perintah

```
izwan@laptop:/tmp/belajar-maven-jdbc/my-app$ mvn clean install

[INFO] Scanning for projects...
[INFO] -----
[INFO] Building my-app
[INFO]    task-segment: [clean, install]
[INFO] -----
[INFO] [clean:clean {execution: default-clean}]
[INFO] [resources:resources {execution: default-resources}]
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /tmp/belajar-maven-jdbc/my-app/src/main/resources
[INFO] [compiler:compile {execution: default-compile}]
[INFO] Compiling 1 source file to /tmp/belajar-maven-jdbc/my-app/target/classes
[INFO] [resources:testResources {execution: default-testResources}]
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /tmp/belajar-maven-jdbc/my-app/src/test/resources
[INFO] [compiler:testCompile {execution: default-testCompile}]
[INFO] Compiling 1 source file to /tmp/belajar-maven-jdbc/my-app/target/test-classes
[INFO] [surefire:test {execution: default-test}]
[INFO] Surefire report directory: /tmp/belajar-maven-jdbc/my-app/target/surefire-reports

-----
T E S T S
-----

Running belajar.jdbc.AppTest
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.054 sec

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

[INFO] [jar:jar {execution: default-jar}]
[INFO] Building jar: /tmp/belajar-maven-jdbc/my-app/target/my-app-1.0-SNAPSHOT.jar
[INFO] [install:install {execution: default-install}]
[INFO] Installing /tmp/belajar-maven-jdbc/my-app/target/my-app-1.0-SNAPSHOT.jar to
/home/adi/.m2/repository/belajar/jdbc/my-app/1.0-SNAPSHOT/my-app-1.0-SNAPSHOT.jar
[INFO] -----
[INFO] BUILD SUCCESSFUL
[INFO] -----
[INFO] Total time: 3 seconds
[INFO] Finished at: Sun Mar 18 11:41:10 WIT 2012
[INFO] Final Memory: 17M/169M
[INFO] -----
izwan@laptop:/tmp/belajar-maven-jdbc/my-app$
```

7. Kemudian buka project menggunakan **Netbeans** atau **Eclipse**, atau bisa juga pake **notepad**.

Buka file **App.java** kemudian edit isi method main menjadi seperti ini :

```
try {
    //parameter yang dikirim adalah host, port, dbname, dbuser, dbpasswd
    Connection conn = DriverManager.getConnection(
        "jdbc:mysql://localhost:3306/belajar","root","admin");
    String query = "insert into mahasiswa (nim, nama) values (?, ?)";
```

```

        PreparedStatement st = conn.prepareStatement(query);
        st.setString(1, "12101022");
        st.setString(2, "Djoko Susanto");
        st.executeUpdate();
    } catch (SQLException ex) {
        Logger.getLogger(App.class.getName()).log(Level.SEVERE, null, ex);
    }
}

```

8. Tambahkan konfigurasi di **pom.xml** agar project bisa di run menggunakan maven

```

<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-dependency-plugin</artifactId>
      <executions>
        <execution>
          <id>copy-dependencies</id>
          <phase>prepare-package</phase>
          <goals>
            <goal>copy-dependencies</goal>
          </goals>
          <configuration>
            <outputDirectory>${project.build.directory}/lib</outputDirectory>
            <overwriteReleases>true</overwriteReleases>
            <overwriteSnapshots>true</overwriteSnapshots>
            <overwriteIfNewer>true</overwriteIfNewer>
          </configuration>
        </execution>
      </executions>
    </plugin>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-jar-plugin</artifactId>
      <configuration>
        <archive>
          <manifest>
            <addClasspath>true</addClasspath>
            <classpathPrefix>${project.build.directory}/lib/</classpathPrefix>
            <mainClass>belajar.jdbc.App</mainClass>
          </manifest>
        </archive>
      </configuration>
    </plugin>
  </plugins>
</build>

```

9. Kemudian lakukan perintah berikut untuk *meng-compile project* dan *running project*

```

mvn clean install
mvn exec:java -Dexec.mainClass="belajar.jdbc.App" -Dexec.classpathScope=runtime

```

Hasilnya adalah sebagai berikut :

```

[INFO] Scanning for projects...
[INFO] Searching repository for plugin with prefix: 'exec'.
[INFO] -----
[INFO] Building my-app
[INFO]   task-segment: [exec:java]
[INFO] -----
[INFO] Preparing exec:java
[INFO] No goals needed for project - skipping
[INFO] [exec:java {execution: default-cli}]
[INFO] -----
[INFO] BUILD SUCCESSFUL
[INFO] -----
[INFO] Total time: 1 second

```

```
[INFO] Finished at: Sun Mar 18 22:38:35 WIT 2012
```

```
[INFO] Final Memory: 9M/103M
```

```
[INFO] -----  
izwan@laptop:/tmp/belajar-maven-jdbc/my-app$
```

10. Kemudian cek di database apakah data yang kita insert melalui aplikasi berhasil tersimpan di database

```
mysql> select * from mahasiswa;
```

```
+-----+-----+  
| nim      | nama          |
```

```
+-----+-----+  
| 12101022 | Djoko Susanto |
```

```
+-----+-----+  
1 row in set (0.00 sec)
```

```
mysql>
```