

## Simple Command Line JAVA Application

Apache Maven was used to create and manage the project.

Once the dependency list was added to the Maven project through editing POM.xml, *maven install* was used to install the dependencies. The dependencies are - JDBC driver which is the database driver to connect to PostgreSQL, and JUnit, the testing framework.

### JDBC

JDBC is Java Database Connectivity driver. The dependency was given in the POM.xml file. The version of the JDBC for PostgreSQL used is 9.3-1104-jdbc41 for PostgreSQL 9.3.

### JUnit

JUnit is the testing framework which is built in java. It is used for unit testing. I have written one JUnit class to test three functions here. This also needs to be given as the dependency in POM.xml. Here the JUnit version I am using is 4.12.

### XML - org.w3c.dom API

The XML is

```
<?xml version="1.0"?>
<persons>
  <person No="1">
    <firstname>meenu</firstname>
    <lastname>vincent</lastname>
  </person>
  <person No="2">
    <firstname>John</firstname>
    <lastname>Smith</lastname>
  </person>
</persons>
```

Here *person No* is not ID. The ID is automatically generated by the system. Only the '*firstname*' and '*lastname*' are taken from the xml and stored in the database. org.w3c.dom API – Java API for XML processing is used for XML parsing.

### Database Specification

DBMS: PostgreSQL

Database name: usermanagement

User: admin

Password: test123

Table name: persons

(user *admin* has all permissions on table *persons*)

Table attributes and attribute types:

1. id - serial NOT NULL primary key

2. firstname - character(15)
3. lastname - character(15)

## Application Specification

### Packages:

1. com.usermanagement.application – For classes for the logic.

#### Classes:

- ReadXML.java
  - UserDetails.java – POJO class
  - UserManagementConsole.java
2. com.usermanagement.dbmodel – For Database connection and operations.

#### Classes:

- DBConnection.java
  - DBOperations.java
3. com.usermanagement.testers – Junit class.

#### Classes:

- UserManagementJUnit.java

## Test Coverage

JUnit is used for testing. Tested three functions which are AddPerson, EditPerson and DeletePerson.

## Execution

The project is exported to runnable JAR file which is usermanagement.jar. the jar file can be run through the following set of commands:

```
java -jar usermanagement.jar
```

The different operations will be displayed onscreen and corresponding options can be selected from the command line.

## Source Code

The folder EclipseProject has a copy of the project from eclipse workspace.