

Assignment No	7 - Assignment
Title	Employee Application
Objective	To Insert and Update a row using JDBC
Roll No	MCA2511

1. To Insert and Update a row using JDBC.

Source Code:

EmployeeBean.java

```
package edu.met.p1;

public class EmployeeBean {
    int eid;
    String ename;
    int esalary;

    public int getEid() {
        return eid;
    }
    public void setEid(int eid) {
        this.eid = eid;
    }
    public String getEname() {
        return ename;
    }
    public void setEname(String ename) {
        this.ename = ename;
    }
    public int getEsalary() {
        return esalary;
    }
    public void setEsalary(int esalary) {
        this.esalary = esalary;
    }
}
```

```
    }
    @Override
    public String toString() {
        return "EmployeesBean [eid=" + eid + ", ename=" + ename + ", esalary=" +
esalary + "]";
    }
}
```

EmployeesDAO.java

```
package edu.met.p1;

import org.springframework.jdbc.core.JdbcTemplate;

public class EmployeesDAO {

    JdbcTemplate jdbcT;

    public EmployeesDAO (JdbcTemplate jdbcT) {
        super();
        this.jdbcT = jdbcT;
    }

    // method to insert record in table
    void insertRow(EmployeeBean e1) {
        String insertQ = "insert into myemployees
values("+e1.getEid()+","+e1.getEname()+"','"+"+e1.getEsalary()+"')";
        jdbcT.update(insertQ);
    }

    //method to update a record in table
    void updateRow(EmployeeBean m1)
    {
        String updateQ = "update myemployees set ename = ?, esalary = ? where
eid=?";
        jdbcT.update(updateQ,m1.getEname(),m1.getEsalary(),m1.getEid());
    }
}
```

EmployeesImpl.java

```
package edu.met.p1;

import org.springframework.context.ApplicationContext;
```

```
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class EmployeesImpl {

    private static ApplicationContext ctx;
    public static void main(String[] args) {
        // TODO Auto-generated method stub

        ctx = new ClassPathXmlApplicationContext("AppCtx.xml");
        //get movies bean Object
        EmployeeBean m1 = (EmployeeBean)ctx.getBean("employee");
        // Get the DAO bean object
        EmployeesDAO dao = (EmployeesDAO)ctx.getBean("employeesDao");
        // insert the record
        dao.insertRow(m1);
        m1.setEid(101);
        m1.setEname("Jay");
        m1.setEsalary(400000);
        dao.insertRow(m1);

        m1.setEid(102);
        m1.setEname("Omkar");
        m1.setEsalary(500000);
        dao.insertRow(m1);

        m1.setEid(103);
        m1.setEname("Nimish");
        m1.setEsalary(450000);
        dao.insertRow(m1);

        m1.setEid(104);
        m1.setEname("Aniket");
        m1.setEsalary(550000);
        dao.insertRow(m1);

        m1.setEid(105);
        m1.setEname("Altamash");
        m1.setEsalary(500000);
        dao.insertRow(m1);

        m1.setEid(106);
        m1.setEname("Bhavesh");
```

```

        m1.setEsalary(500000);
        dao.insertRow(m1);

        m1.setEid(106);
        m1.setEname("Sumit");
        m1.setEsalary(520000);
        dao.updateRow(m1);
    }

}

```

AppCtx.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<beans
    xmlns="http://www.springframework.org/schema/beans"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:p="http://www.springframework.org/schema/p"
    xsi:schemaLocation="http://www.springframework.org/schema/beans
        http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">

    <bean id="ds"
        class="org.springframework.jdbc.datasource.DriverManagerDataSource">
        <property name="driverClassName" value="org.postgresql.Driver" />
        <property name="url" value="jdbc:postgresql://localhost:5432/postgres"/>
        <property name="username" value="postgres" />
        <property name="password" value="password" />
    </bean>

    <bean id="jdbcT" class="org.springframework.jdbc.core.JdbcTemplate">
        <property name="dataSource" ref="ds"/>
    </bean>

    <bean id="employeesDao" class="edu.met.p1.EmployeesDAO">
        <constructor-arg name="jdbcT" ref="jdbcT"/>
    </bean>

    <bean id="employee" class="edu.met.p1.EmployeeBean">
        <property name="eid" value="101"/>
        <property name="ename" value="Mukund"/>
        <property name="esalary" value="500000"/>
    </bean>
</beans>

```

Output:

```

    EmployeeApplication
      > Spring Elements
      < src
        < edu.met.p1
          > EmployeeBean.java
          > EmployeesDAO.java
          > EmployeesImpl.java
          < AppCtx.xml
      > JRE System Library [JavaSE-17]
      > postgres
      > springlibs
  
```

eid integer	ename character varying (20)	esalary integer
1	101	Mukund
2	101	Jay
3	102	Omkar
4	103	Nimish
5	104	Aniket
6	105	Altamash
7	106	Sumit
8	106	Bhavesh