



Program No:	23		
Roll No:	1525		
Title of Program:	Spring jdbc:employee application		
Objective :	1.program to insert from the given table		
	2.To demonstrate RowMapper interface to fetch the records		
	from the database		

```
Source code:
Employee.java
package edu.met.p1;
public class Employee {
int eid;
String ename;
int salary;
public Employee(int eid, String ename, int salary) {
super();
this.eid = eid;
this.ename = ename;
this.salary = salary;
public Employee() {
// TODO Auto-generated constructor stub
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 getSalary() {
return salary;
}
public void setSalary(int salary) {
this.salary = salary;
}
@Override
public String toString() {
return "Employee [eid=" + eid + ", ename=" + ename + ",
salary=" + salary + "]";
}
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="empbean" class="edu.met.p1.Employee">
    <constructor-arg value="102"></constructor-arg>
    <constructor-arg value="ELLIOT"></constructor-arg>
    <constructor-arg value="25000"></constructor-arg>
   </bean>
   <bean id="ds"
class="org.springframework.jdbc.datasource.DriverManagerDataSou
rce">
    cproperty name="driverClassName"
value="org.postgresql.Driver">
    cproperty name="url"
value="jdbc:postgresql://Localhost:5432/postgres">
    cproperty name="username" value="postgres"></property>
    cproperty name="password" value="password"></property>
   </bean>
   <bean id="jdbcT"</pre>
class="org.springframework.jdbc.core.JdbcTemplate">
    cproperty name="dataSource" ref="ds">
   </bean>
   <bean id="empD" class="edu.met.p1.EmpLoyeeDao">
    <constructor-arg ref="jdbcT"></constructor-arg>
   </bean>
</beans>
```

EmployeeDao.java



```
package edu.met.p1;
import java.sql.*;
import java.util.List;
import org.springframework.jdbc.core.*;
public class EmployeeDao {
JdbcTemplate jdbcT;
public EmployeeDao(JdbcTemplate jdbcT) {
super();
this.jdbcT = jdbcT;
}
public int saveEmp(Employee e1)
String ins="Insert into employee values("+e1.getEid()
+",'"+e1.getEname()+"',"+e1.getSalary()+")";
return jdbcT.update(ins);
}
public List<Employee> getAll()
{
    String selQ="Select * from employee";
    return jdbcT.query(selQ, new EmployeeRowMapper());
}
}
```



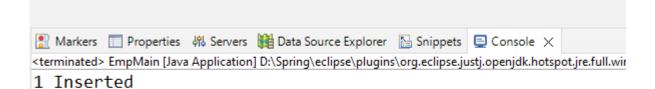
## EmpMain.java

```
package edu.met.p1;
import java.util.List;
import org.springframework.context.ApplicationContext;
import
org.springframework.context.support.ClassPathXmlApplicationCont
public class EmpMain {
static ApplicationContext ctx;
public static void main(String[] args)
// TODO Auto-generated method stub
ctx=new ClassPathXmlApplicationContext("AppCtx.xml");
Employee e1=(Employee)ctx.getBean("empbean");
EmployeeDao ed=(EmployeeDao)ctx.getBean("empD");
//System.out.println(ed.saveEmp(e1)+" Inserted");
List<Employee> elist=ed.getAll();
System.out.println("Eid\tEname\tSsalary");
for(Employee e: elist)
         System.out.println(e.getEid()+"\t"+e.getEname()
+"\t"+e.getSalary());
}
}
```



```
EmployeeRowMapper.java
package edu.met.p1;
import java.sql.ResultSet;
import java.sql.SQLException;
import org.springframework.jdbc.core.RowMapper;
public class EmployeeRowMapper implements RowMapper<Employee> {
    @Override
    public Employee mapRow(ResultSet arg0, int arg1) throws
SQLException {
         // TODO Auto-generated method stub
         Employee temp=new Employee();
         temp.setEid(arg0.getInt(1));
         temp.setEname(arg0.getString(2));
         temp.setSalary(arg0.getInt(3));
         return temp;
    }
}
OUTPUT:
```





## Data Output Explain Messages Notifications ename eid salary integer 🖴 character varying (50) integer 101 sharukh 10000 1 102 ELLIOT 25000 2

Data Output		Explain	Messages	Notifications
4	eid integer	ename character	Read-only co	salary 🛕
1	101	sharukh	, , , , ,	10000
2	102	ELLIOT		25000
3	102	ELLIOT		25000

