**12.1 Aim: Write a program to demonstrate Spring JdbcTemplate class to store data in database table.**

**Code –**

**On pgAdmin4**

create table mymovies(mid int,title varchar,actor varchar);

select \* from mymovies;

**Movies.java**

**package** org.viva;

**public** **class** Movies {

**int** mid;

String title;

String actor;

**public** Movies(**int** mid, String title, String actor) {

**super**();

**this**.mid = mid;

**this**.title = title;

**this**.actor = actor;

}

/\*\*

\* **@return** the mid

\*/

**public** **int** getMid() {

**return** mid;

}

/\*\*

\* **@param** mid the mid to set

\*/

**public** **void** setMid(**int** mid) {

**this**.mid = mid;

}

/\*\*

\* **@return** the title

\*/

**public** String getTitle() {

**return** title;

}

/\*\*

\* **@param** title the title to set

\*/

**public** **void** setTitle(String title) {

**this**.title = title;

}

/\*\*

\* **@return** the actor

\*/

**public** String getActor() {

**return** actor;

}

/\*\*

\* **@param** actor the actor to set

\*/

**public** **void** setActor(String actor) {

**this**.actor = actor;

}

}

**MoviesDao.java**

**package** org.viva;

**import** org.springframework.jdbc.core.\*;

**public** **class** MoviesDao {

JdbcTemplate jdbcTemplate;

**public** JdbcTemplate getJdbcTemplate() {

**return** jdbcTemplate;

}

**public** JdbcTemplate setJdbcTemplate(JdbcTemplate jdbcTemplate) {

**return** **this**.jdbcTemplate = jdbcTemplate;

}

**public** **int** insMovie(Movies m1)

{

String insSql="insert into mymovies values("+m1.getMid()+",'"+m1.getTitle()+"','"+m1.getActor()+"')";

**return** jdbcTemplate.update(insSql);//returns rows affected

}

}

**MovieTest.java**

**package** org.viva;

**import** org.springframework.beans.factory.BeanFactory;

**import** org.springframework.context.ApplicationContext;

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

**public** **class** MovieTest {

**private** **static** BeanFactory *appCon*;

**public** **static** **void** main(String[] args) {

*appCon* = **new** ClassPathXmlApplicationContext("Appctx2.xml");

MoviesDao m1=(MoviesDao)*appCon*.getBean("moviebean");

Movies t1=**new** Movies(1,"A Beautiful Mind","Russel Crow");

System.***out***.println(m1.insMovie(t1));

}

}

**Appctx2.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"

xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd">

<bean id="ds" class="org.springframework.jdbc.datasource.DriverManagerDataSource">

<property name="driverClassName" value="org.postgresql.Driver" />

<property name="url" value="jdbc:postgresql://localhost:5433/postgres" />

<property name="username" value="postgres" />

<property name="password" value="abc" />

</bean>

<bean id="jdbcTemplate" class="org.springframework.jdbc.core.JdbcTemplate">

<property name="dataSource" ref="ds"></property>

</bean>

<bean id="moviebean" class="org.viva.MoviesDao">

<property name="jdbcTemplate" ref="jdbcTemplate"></property>

</bean>

</beans>