**Spring UPDATE**

**package com.springcore.jdbc\_entity;**

**public class Author {**

**private String aut\_id;**

**private String aut\_name;**

**private String aut\_email;**

**public String getAut\_id() {**

**return aut\_id;**

**}**

**public void setAut\_id(String aut\_id) {**

**this.aut\_id = aut\_id;**

**}**

**public String getAut\_name() {**

**return aut\_name;**

**}**

**public void setAut\_name(String aut\_name) {**

**this.aut\_name = aut\_name;**

**}**

**public String getAut\_email() {**

**return aut\_email;**

**}**

**public void setAut\_email(String aut\_email) {**

**this.aut\_email = aut\_email;**

**}**

**}**

**package com.springcore.jdbc\_dao;**

**import com.springcore.jdbc\_entity.Author;**

**public interface AuthorInter {**

**public int insert(Author author);**

**public int change(Author author);**

**}**

**package com.springcore.jdbc\_dao;**

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

**import com.springcore.jdbc\_entity.Author;**

**public class AuthorImplimentation implements AuthorInter {**

**private JdbcTemplate jdbcTemplate;**

**public int insert(Author author) {**

**// create an insert query variable**

**String insertQuery = "INSERT INTO authors (id,name,email) VALUES(?,?,?)";**

**int tot\_records = this.jdbcTemplate.update(insertQuery, author.getAut\_id(), author.getAut\_name(),**

**author.getAut\_email());**

**return tot\_records;**

**}**

**public int change(Author author) {**

**String query = "update authors set name=?, email=? where id=?";**

**int tot\_records\_updtd = this.jdbcTemplate.update(query, author.getAut\_name(), author.getAut\_email(),**

**author.getAut\_id());**

**return tot\_records\_updtd;**

**}**

**public JdbcTemplate getJdbcTemplate() {**

**return jdbcTemplate;**

**}**

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

**this.jdbcTemplate = jdbcTemplate;**

**}**

**}**

**<?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:context="http://www.springframework.org/schema/context"**

**xmlns:p="http://www.springframework.org/schema/p"**

**xmlns:util="http://www.springframework.org/schema/util"**

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

**http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context.xsd**

**http://www.springframework.org/schema/util http://www.springframework.org/schema/util/spring-util.xsd">**

**<context:annotation-config />**

**<context:component-scan base-package="com.baeldung.applicationcontext" />**

**<bean name="jdc"**

**class="org.springframework.jdbc.datasource.DriverManagerDataSource">**

**<property name="driverClassName" value="com.mysql.cj.jdbc.Driver" />**

**<property name="url" value="jdbc:mysql://localhost:3306/springdb?useSSL=false" />**

**<property name="username" value="root" />**

**<property name="password" value="root" />**

**</bean>**

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

**<property name="dataSource">**

**<ref bean="jdc" />**

**</property>**

**</bean>**

**<bean name="auth\_impl\_dao"**

**class="com.springcore.jdbc\_dao.AuthorImplimentation">**

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

**</bean>**

**</beans>**

**package com.app.springcore.jdbc\_with\_di;**

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

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

**import com.springcore.jdbc\_dao.AuthorImplimentation;**

**import com.springcore.jdbc\_entity.Author;**

**public class TestJDBC\_Connection\_DI {**

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

**AbstractApplicationContext context = new ClassPathXmlApplicationContext("jdbc\_config\_di.xml");**

**AuthorImplimentation ai = context.getBean("auth\_impl\_dao", AuthorImplimentation.class);**

**System.out.println("JDBC Bean Created !!!");**

**Author a = new Author();**

**/\***

**\* a.setAut\_id("1234"); a.setAut\_name("Madhava");**

**\* a.setAut\_email("madhav@gmail.com");**

**\*/**

**a.setAut\_name("ramya");**

**a.setAut\_email("ramya@outlook.com");**

**a.setAut\_id("1234");**

**int result = ai.change(a);**

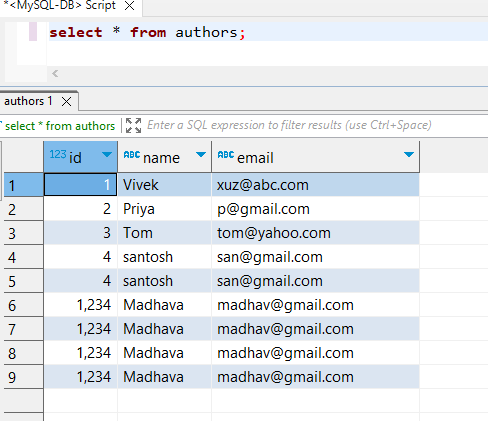
**System.out.println("No of records updated: " + result);**

**context.close();**

**}**

**}**

Before:



After:

