

## Code for Searching and Updating Information

### Pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com</groupId>
  <artifactId>SearchingandUpdating</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>war</packaging>
  <name>SearchingandUpdating Maven Webapp</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>
  <properties>
    <project.build.sourceEncoding>UTF-
8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.11</version>
      <scope>test</scope>
    </dependency>
    <!--spring context -->
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-context</artifactId>
      <version>5.3.22</version>
    </dependency>
    <!--spring core -->
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-core</artifactId>
      <version>5.3.22</version>
    </dependency>
    <!--Spring beans -->
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-beans</artifactId>
      <version>5.3.22</version>
    </dependency>
    <!--spring web -->
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-web</artifactId>
      <version>5.3.22</version>
    </dependency>
    <!--spring webmvc -->
    <dependency>
```

```

        <groupId>org.springframework</groupId>
        <artifactId>spring-webmvc</artifactId>
        <version>5.3.22</version>
    </dependency>
    <!--spring jdbc -->
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-jdbc</artifactId>
        <version>5.3.22</version>
    </dependency>
    <!--mysql connector -->
    <dependency>
        <groupId>mysql</groupId>
        <artifactId>mysql-connector-java</artifactId>
        <version>8.0.30</version>
    </dependency>
    <!--java servlet jstl -->
    <dependency>
        <groupId>javax.servlet</groupId>
        <artifactId>jstl</artifactId>
        <version>1.2</version>
    </dependency>
    <!--java servlet API -->
    <dependency>
        <groupId>javax.servlet</groupId>
        <artifactId>javax.servlet-api</artifactId>
        <version>4.0.1</version>
        <scope>provided</scope>
    </dependency>
</dependencies>
<build>
    <finalName>SearchingandUpdating</finalName>
    <pluginManagement><!-- lock down plugins versions to avoid using
Maven
defaults (may be moved to parent pom) -->
        <plugins>
            <plugin>
                <artifactId>maven-clean-plugin</artifactId>
                <version>3.1.0</version>
            </plugin>
            <!-- see http://maven.apache.org/ref/current/maven-
bindings.html#Plugin_bindings_for_war_packaging -->
            <plugin>
                <artifactId>maven-resources-plugin</artifactId>
                <version>3.0.2</version>
            </plugin>
            <plugin>
                <artifactId>maven-compiler-plugin</artifactId>
                <version>3.8.0</version>
            </plugin>
            <plugin>
                <artifactId>maven-surefire-plugin</artifactId>
                <version>2.22.1</version>
            </plugin>
            <plugin>
                <artifactId>maven-war-plugin</artifactId>

```

```

        <version>3.2.2</version>
    </plugin>
    <plugin>
        <artifactId>maven-install-plugin</artifactId>
        <version>2.5.2</version>
    </plugin>
    <plugin>
        <artifactId>maven-deploy-plugin</artifactId>
        <version>2.8.2</version>
    </plugin>
</plugins>
</pluginManagement>
</build>
</project>

```

### **MainController.java**

```

package com.SearchingandUpdating.controller;

import java.util.List;

import javax.servlet.http.HttpServletRequest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.ModelMap;
import org.springframework.web.bind.annotation.GetMapping;

import com.SearchingandUpdating.dao.UserDao;
import com.SearchingandUpdating.entity.UserEntity;

@Controller
public class MainController {

    @Autowired

```

```
UserDao dao;
```

```
@GetMapping("/listuser")
```

```
public String getAllUsers(ModelMap map) {  
    List<UserEntity> list = dao.getAllUsers();  
    map.addAttribute("list", list);  
    return "users";  
}
```

```
@GetMapping("/details")
```

```
public String getUser(HttpServletRequest request, ModelMap map) {  
    long id = Long.parseLong(request.getParameter("id"));  
    UserEntity entity = dao.getUserById(id);  
    map.addAttribute("obj", entity);  
    return "details";  
}
```

```
@GetMapping("/success")
```

```
public void updateUser(HttpServletRequest request, ModelMap map) {  
    long id = Long.parseLong(request.getParameter("id"));  
    String name = request.getParameter("name");  
    String salary = request.getParameter("salary");  
    dao.updateUser(id, name, salary);  
}
```

```
}
```

## **UserDao.java**

```
package com.SearchingandUpdating.dao;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.dao.DataAccessException;

import org.springframework.jdbc.core.JdbcTemplate;

import org.springframework.jdbc.core.ResultSetExtractor;

import org.springframework.jdbc.core.RowMapper;

import org.springframework.stereotype.Repository;

import com.SearchingandUpdating.entity.UserEntity;

@Repository

public class UserDao {

    @Autowired

    JdbcTemplate jdbcTemplate;

    public List<UserEntity> getAllUsers() {

        return jdbcTemplate.query("select * from user", new RowMapper<UserEntity>() {

            public UserEntity mapRow(ResultSet rs, int rowNum) throws SQLException {

                UserEntity obj = new UserEntity();

                obj.setId(rs.getLong(1));

                obj.setName(rs.getString(2));

            }

        });

    }

}
```

```

        obj.setSalary(rs.getBigDecimal(3));

        return obj;
    }

});
}

```

```

public UserEntity getUserById(long id) {

    return jdbcTemplate.query("select * from user where id=" + id, new
ResultSetExtractor<UserEntity>() {

        public UserEntity extractData(ResultSet rs) throws SQLException,
        DataAccessException {

            // TODO Auto-generated method stub

            if (rs.next()) {

                UserEntity obj = new UserEntity();

                obj.setId(rs.getLong(1));

                obj.setName(rs.getString(2));

                obj.setSalary(rs.getBigDecimal(3));

                return obj;

            }

            return null;

        }

    });
}

```

```

public void updateUser(long id,String name,String salary) {

String sql = "update user set name='"+name+"',"+"salary="+salary+" where id="+id;

```

```
jdbcTemplate.update(sql);
```

```
}
```

```
}
```

### UserEntity.java

```
package com.SearchingandUpdating.entity;
```

```
import java.math.BigDecimal;
```

```
public class UserEntity {  
    private long id;  
    private String name;  
    private BigDecimal salary;  
  
    public UserEntity() {  
    }  
  
    public UserEntity(long id, String name, BigDecimal salary) {  
        super();  
        this.id = id;  
        this.name = name;  
        this.salary = salary;  
    }  
  
    public long getId() {  
        return id;  
    }  
  
    public void setId(long id) {  
        this.id = id;  
    }  
  
    public String getName() {  
        return name;  
    }  
  
    public void setName(String name) {  
        this.name = name;  
    }  
  
    public BigDecimal getSalary() {  
        return salary;  
    }  
  
    public void setSalary(BigDecimal salary) {  
        this.salary = salary;  
    }  
  
    @Override  
    public String toString() {  
        return "UserEntity [id=" + id + ", name=" + name + "];"  
    }  
}
```

```
}
```

## Index.jsp

```
<html>
<body>
<h2>Spring Application</h2>
<a href="listuser">List All Users</a>
<br><br>
<form action="details">
<h3>Search for User</h3>
Enter Id : <input type="text" name="id" >
<input type="submit" value="submit">
</form>
</body>
</html>
```

## details.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8" isELIgnored="false"%>
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Update of Details</title>
</head>
<body>
<h3>Details of the User</h3>
<c:set var="user" value="${obj }"></c:set>
UserID: ${user.id}
<br> Name: ${user.name }
<br> Salary: ${user.salary }
<br>
<br>
<h3>You can update any UserDetails here</h3>
<form action="success">
user id : <input type="text" name="id"><br> <br>
Name: <input type="text" name="name"><br> <br>
Salary : <input type="text" name="salary"><br> <br>
<input type="submit" value="submit">
</form>
</body>
</html>
```

## success.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
```



```

<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h1> successfully updated</h1>
<br><br>
<a href="listuser">To see all updated users, click here</a>
</body>
</html>

```

## Users.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h1>Available Users </h1>
<table border=1 cellpadding=10 cellspacing=0>
<tr><th>ID</th><th>NAME</th><th>SALARY</th></tr>
<c:forEach var="p" items="${list }">
<tr>
<td>${p.id }</td>
<td>${p.name }</td>
<td>${p.salary }</td>
</tr>
</c:forEach>
</table>
</body>
</html>

```

## test-servlet.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:context="http://www.springframework.org/schema/context"
    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">
<context:annotation-config></context:annotation-config>
<!-- jsp output -->
<context:component-scan base-
package="com.SearchingandUpdating.controller"></context:component-scan>
<bean id="jspViewResolver"
class="org.springframework.web.servlet.view.InternalResourceViewResolver">
<property name="suffix" value=".jsp"></property>

```

```

<property name="prefix" value="/WEB-INF/view/"></property>
</bean>
<!-- database -->
<bean id="dataSource"
class="org.springframework.jdbc.datasource.DriverManagerDataSource">
<property name="driverClassName" value="com.mysql.cj.jdbc.Driver"></property>
<property name="url" value="jdbc:mysql://localhost:3306/pp1"></property>
<property name="username" value="root"></property>
<property name="password" value="ratnamla1999"></property>
</bean>
<bean id="jdbcTemplate" class="org.springframework.jdbc.core.JdbcTemplate">
<property name="dataSource" ref="dataSource"></property>
</bean>
<bean id="dao" class="com.SearchingandUpdating.dao.UserDao"></bean>
</beans>

```

## web.xml

```

<web-app>
<display-name>Archetype Created Web Application</display-name>
<servlet>
<servlet-name>test</servlet-name>
<servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-
class>
</servlet>
<servlet-mapping>
<servlet-name>test</servlet-name>
<url-pattern>/</url-pattern>
</servlet-mapping>
<context-param>
<param-name>contextConfigLocation</param-name>
<param-value>/WEB-INF/test-servlet.xml</param-value>
</context-param>
<listener>
<listener-
class>org.springframework.web.context.ContextLoaderListener</listener-class>
</listener>
</web-app>

```