Index.jsp:

<html>

<body>

<h2>Ecommerce Product</h2>

<form action="productid">

<input type="number" name="id" placeholder="Enter id">

<input type="submit" value="Enter">

</form>

</body>

</html> Update.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>

<h2>Enter details:</h2>

<form action="update">

<input type="number" name="id" placeholder="Enter product id"><br>

<input type="text" name="name" placeholder="Enter product name"><br>

<input type="number" name="price" placeholder="Enter product price"><br>

<input type="submit" value="Confirm">

</form>

</body>

</html>

Details.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>

<h3>Products Details Page</h3>

<c:set var="product" value="${obj }"></c:set> Id: ${product.id } <br>

Name: ${product.name } <br> Price: ${product.price } <br><br>

<a href="update.jsp">Click here to update a product</a> <br><br>

<a href="index.jsp">Click here go to home page</a>

</body>

</html>

Test-servlet.xml:

<?xml version="1.0" encoding="UTF-8"?>

<beans xmlns="[http://www.springframework.org/schema/beans"](http://www.springframework.org/schema/beans) xmlns:xsi="[http://www.w3.org/2001/XMLSchema-instance"](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.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/ecommerce1"></property>

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

<property name="password" value="Vaishu18"></property>

</bean>

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

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

</bean>

<bean id="dao" class="com.dao.EProductDao"></bean>

</beans>

Web.xml

<web-app>

<display-name>Archetype Created Web Application</display-name>

<servlet>

</servlet>

<servlet-name>test</servlet-name>

<servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>

<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> Pom.xml:

<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/maven-> v4\_0\_0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>SpringDemo</groupId>

<artifactId>Updateproject</artifactId>

<packaging>war</packaging>

<version>0.0.1-SNAPSHOT</version>

<name>Updateproject Maven Webapp</name>

<ur[l>h](http://maven.apache.org/)t[tp://maven.apache.org<](http://maven.apache.org/)/url>

<dependencies>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>3.8.1</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-->

<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>

<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>

<!-- https://mvnrepository.com/artifact/log4j/log4j -->

<dependency>

<groupId>log4j</groupId>

<artifactId>log4j</artifactId>

<version>1.2.17</version>

</dependency>

<build>

<finalName>Phase3\_lesson1\_UserInformation</finalName>

<plugins>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-war-plugin</artifactId>

<version>3.3.1</version>

</plugin>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.10.1</version>

</plugin>

</plugins>

</build>

</project>

/MainController.java:

package com.controller;

import java.io.IOException;

import java.io.PrintWriter; import java.math.BigDecimal; import java.text.ParseException;

import java.text.SimpleDateFormat; import java.util.Date;

import javax.servlet.ServletException; 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.dao.EProductDao;

import com.entity.EProductEntity; import org.apache.log4j.Logger; @Controller

public class MainController {

@Autowired EProductDao dao;

private Logger logger = Logger.getLogger(MainController.class);

@GetMapping("/productid")

public String getProduct(HttpServletRequest request,ModelMap map, HttpServletResponse response)

{

long id=Long.parseLong(request.getParameter("id")); if(id>0)

{

EProductEntity entity=dao.getProductById(id);

logger.info("This Is An Info Log Entry !");

logger.info(entity); map.addAttribute("obj",entity); return "details";

}

else

{

try { try {

request.getRequestDispatcher("/index.jsp").include(request, response); response.setContentType("text/html");

PrintWriter out = response.getWriter(); out.println("Invalid Credentials:)");

} catch (ServletException e) { e.printStackTrace();

}

} catch (IOException e) { e.printStackTrace();

}

}

return null;

}

@GetMapping("/update")

public String updateProduct(HttpServletRequest request,ModelMap map) { long id=Long.parseLong(request.getParameter("id"));

String name=request.getParameter("name");

BigDecimal price = new BigDecimal(request.getParameter("price"));

{

EProductEntity entity= new EProductEntity(); entity.setId(id);

entity.setName(name); entity.setPrice(price); dao.updateProduct(entity);

map.addAttribute("obj",entity); return "details";

}

}

}

EProductDao.java:

package com.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.entity.EProductEntity;

public class EProductDao {

@Autowired

JdbcTemplate jdbcTemplate;

public List<EProductEntity> getAllProducts() {

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

EProductEntity obj = new EProductEntity(); obj.setId(rs.getLong(1)); obj.setName(rs.getString(2)); obj.setPrice(rs.getBigDecimal(3));

return obj;

}

});

}

public EProductEntity getProductById(long id) {

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

public EProductEntity extractData(ResultSet rs) throws SQLException, DataAccessException { if (rs.next()) {

EProductEntity obj = new EProductEntity(); obj.setId(rs.getLong(1)); obj.setName(rs.getString(2)); obj.setPrice(rs.getBigDecimal(3));

return obj;

}

return null;

}

});

public void updateProduct(EProductEntity obj) { if (obj.getId() > 0) {

String sql = "update eproduct set name=?,price=? where id=?"; jdbcTemplate.update(sql, obj.getName(), obj.getPrice(), obj.getId());

} else {

String sql = "insert into eproduct (name,price) values (?,?)"; jdbcTemplate.update(sql, obj.getName(), obj.getPrice());

}

}

}

EProductEntity.java:

package com.entity;

import java.math.BigDecimal;

public class EProductEntity {

private long id; private String name;

private BigDecimal price;

public EProductEntity() {

}

public EProductEntity(String name, BigDecimal price) { super();

this.name = name; this.price = price;

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 getPrice() { return price;

}

public void setPrice(BigDecimal price) { this.price = price;

}

@Override

public String toString() {

return "EProductEntity [id=" + id + ", name=" + name + ", price=" + price + "]";

}