**Package-1:-**

Controller:-

//EmployeeOperationsController.java

package com.nt.controller;

import java.util.List;

import java.util.Map;

import org.springframework.beans.BeanUtils;

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

import org.springframework.stereotype.Controller;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.RequestParam;

import org.springframework.web.servlet.mvc.support.RedirectAttributes;

import com.nt.model.Employee;

import com.nt.service.IEmployeeServiceMgmt;

@Controller

public class EmployeeOperationsController {

@Autowired

private IEmployeeServiceMgmt service;

@GetMapping("/")

public String showHomePage() {

//return LVN

return "home";

}

@GetMapping("/employee\_report")

public String showEmployeeReport(Map<String,Object> map) {

//use service

List<Employee> list=service.fetchAllEmps();

// add model attribute to map collection

map.put("empsData", list);

//return LVN

return "show\_emp\_report";

}

@GetMapping("/employee\_add")

public String showAddEmployeeFormPage(@ModelAttribute("emp") Employee emp) {

//return lvn

return "employee\_register";

}

@PostMapping("/employee\_add")

public String processAddEmployeeForm(@ModelAttribute("emp") Employee emp, RedirectAttributes attrs) {

// use service

String msg=service.regisgterEmployee(emp);

//keep result in model attribute

attrs.addFlashAttribute("resultMsg",msg);

// redirect to other handler method

return "redirect:employee\_report";

}

@GetMapping("/employee\_delete")

public String processDeleteEmployee(@RequestParam("no") int no,RedirectAttributes attrs) {

//use service

String msg=service.deleteEmployee(no);

//add Redirect attributes

attrs.addFlashAttribute("resultMsg", msg);

//redirect to other handler method

return "redirect:employee\_report";

}

@GetMapping("/employee\_edit")

public String showUpdateEmployeeFormPage(@ModelAttribute("emp") Employee emp,@RequestParam("no") int no) {

// get Employee record by no

Employee emp1=service.fetchEmployeeByNo(no);

//copy emp1 objet data to emp object

BeanUtils.copyProperties(emp1, emp);

//return LVN

return "employee\_modify";

}

@PostMapping("/employee\_edit")

public String processUpdateEmployeeFormPage(@ModelAttribute("emp") Employee emp, RedirectAttributes attrs) {

//use service

String msg=service.modifyEmployee(emp);

// add model attributes to RedirectAttrs object

attrs.addFlashAttribute("resultMsg", msg);

//redirect the request to another handler method

return "redirect:employee\_report";

}

}

**Package-2:-dao**

//EmployeeDAOImpl.java

package com.nt.dao;

import java.util.ArrayList;

import java.util.List;

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

import org.springframework.jdbc.core.JdbcTemplate;

import org.springframework.stereotype.Repository;

import com.nt.model.Employee;

@Repository("empDAO")

public class EmployeeDAOImpl implements IEmployeeDAO {

private static final String GET\_ALL\_EMPS="SELECT EMPNO,ENAME,JOB,SAL FROM EMP";

@Autowired

private JdbcTemplate jt;

@Override

public List<Employee> getAllEmployees() {

List<Employee> list=jt.query(GET\_ALL\_EMPS, rs->{

List<Employee> al=new ArrayList();

while(rs.next()) {

Employee e=new Employee();

e.setEmpno(rs.getInt(1));

e.setEname(rs.getString(2));

e.setJob(rs.getString(3));

e.setSal(rs.getFloat(4));

al.add(e);

}//whilke

return al;

});

return list;

} //method

private static final String INSERT\_EMP\_QUERY="INSERT INTO EMP(EMPNO,ENAME,JOB,SAL) VALUES(EMP\_NO\_SEQ.NEXTVAL,?,?,?)";

@Override

public int insert(Employee emp) {

int count=jt.update(INSERT\_EMP\_QUERY, emp.getEname(),emp.getJob(),emp.getSal());

return count;

}

private static final String DELETE\_EMP\_BY\_NO="DELETE FROM EMP WHERE EMPNO=?";

@Override

public int remove(int eno) {

int count=jt.update(DELETE\_EMP\_BY\_NO,eno);

return count;

}

private static final String GET\_EMP\_BY\_NO="SELECT EMPNO,ENAME,JOB,SAL FROM EMP WHERE EMPNO=?";

@Override

public Employee getEmployeeByNo(int no) {

Employee emp=jt.queryForObject(GET\_EMP\_BY\_NO,

(rs,rownum)->{

// copying ResultSet obj record to Employee object

Employee e=new Employee();

e.setEmpno(rs.getInt(1));

e.setEname(rs.getString(2));

e.setJob(rs.getString(3));

e.setSal(rs.getFloat(4));

return e;

},

no);

return emp;

}//method

private static final String UPDATE\_EMP\_BY\_NO="UPDATE EMP SET ENAME=?,JOB=?,SAL=? WHERE EMPNO=?";

@Override

public int update(Employee emp) {

int count=jt.update(UPDATE\_EMP\_BY\_NO, emp.getEname(),emp.getJob(),emp.getSal(),emp.getEmpno());

return count;

}

}//class

//IEmployeeDAO.java

//IEmployeeDAO .java (DAO Interface)

package com.nt.dao;

import java.util.List;

import com.nt.model.Employee;

public interface IEmployeeDAO {

public List<Employee> getAllEmployees();

public int insert(Employee emp);

public int remove(int eno);

public Employee getEmployeeByNo(int no);

public int update(Employee emp);

}

**Package-3:-model**

//Employee.java

package com.nt.model;

public class Employee {

private Integer empno;

private String ename;

private String job;

private Float sal;

// setters && getters (alt+shit+s,r)

public Integer getEmpno() {

return empno;

}

public void setEmpno(Integer empno) {

this.empno = empno;

}

public String getEname() {

return ename;

}

public void setEname(String ename) {

this.ename = ename;

}

public String getJob() {

return job;

}

public void setJob(String job) {

this.job = job;

}

public Float getSal() {

return sal;

}

public void setSal(Float sal) {

this.sal = sal;

}

//toStirng ( alt+shit+s,s)

@Override

public String toString() {

return "Employee [empno=" + empno + ", ename=" + ename + ", job=" + job + ", sal=" + sal + ", getClass()="

+ getClass() + ", hashCode()=" + hashCode() + ", toString()=" + super.toString() + "]";

}

}

**Package-4:-service**

//EmployeeServiceMgmtImpl.java

package com.nt.service;

import java.util.List;

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

import org.springframework.stereotype.Service;

import com.nt.dao.IEmployeeDAO;

import com.nt.model.Employee;

@Service("empService")

public class EmployeeServiceMgmtImpl implements IEmployeeServiceMgmt {

@Autowired

private IEmployeeDAO empDAO;

@Override

public List<Employee> fetchAllEmps() {

//use DAO

List<Employee> list=empDAO.getAllEmployees();

return list;

}

@Override

public String regisgterEmployee(Employee emp) {

//use DAO

int count=empDAO.insert(emp);

return count==0?"Employee not registered":"Employee registered";

}

@Override

public String deleteEmployee(int eno) {

int count=empDAO.remove(eno);

return count==0?"Employee not removed":"Employee removed";

}

@Override

public Employee fetchEmployeeByNo(int no) {

Employee emp=empDAO.getEmployeeByNo(no);

return emp;

}

@Override

public String modifyEmployee(Employee emp) {

//use DAO

int count=empDAO.update(emp);

return count==0?" Employee not updated":"Employee updated";

}

}

//IEmployeeServiceMgmt.java (service interface)

package com.nt.service;

import java.util.List;

import com.nt.model.Employee;

public interface IEmployeeServiceMgmt {

public List<Employee> fetchAllEmps();

public String regisgterEmployee(Employee emp);

public String deleteEmployee(int eno);

public Employee fetchEmployeeByNo(int no);

public String modifyEmployee(Employee emp);

}

<?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>nit</groupId>

<artifactId>MVCProject02-SecondApp</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>war</packaging>

<name>MVCProject02-SecondApp 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>17</maven.compiler.source>

<maven.compiler.target>17</maven.compiler.target>

</properties>

<dependencies>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.11</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>javax.servlet-api</artifactId>

<version>4.0.0</version>

<scope>provided</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>5.3.22</version>

</dependency>

</dependencies>

<build>

<finalName>MVCProject02-SecondApp</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-core/default-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>