Employeecontroller.java

package com.employeeProject.spring.employeeproject.controller;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.List;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

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

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

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

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

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

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

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

import com.employeeProject.spring.employeeproject.entity.Employee;

import com.employeeProject.spring.employeeproject.repository.EmployeeRepository;

import com.employeeProject.spring.employeeproject.services.EmployeeService;

@Controller

public class EmployeeController {

Logger logger = LoggerFactory.getLogger(this.getClass());

@Autowired

private EmployeeService employeeService;

@GetMapping("/")

public String index() {

return "index";

}

@GetMapping("/addnewemployee")

public String addEmployee(Model model) {

model.addAttribute("employee", new Employee());

return "createEmployee";

}

@PostMapping(“/storeEmployee")

public String addEmployee(@ModelAttribute("employee") Employee employee) {

System.out.println("-----------------------------------" + employee + "-------------------");

employeeService.saveEmployee(employee);

return "redirect:/";

}

@GetMapping("/display")

public String displayEmployee(Model model) {

List<Employee> listEmployee = employeeService.getAllEmployees();

model.addAttribute("employee", listEmployee);

return "display";

}

@GetMapping("/delete")

public String removeEmployee(Model model) {

model.addAttribute("employee",new Employee());

return "DeleteEmployee";

}

@RequestMapping(value="/deletedEmployee", method = RequestMethod.POST)

public String removeEmployee(@ModelAttribute("employee") Employee employee){

List<Employee> listEmployees = employeeService.getAllEmployees();

System.out.println("-----------------------------------" + employee + "-------------------");

for(Employee employee2 : listEmployees) {

System.out.println(employee2.getName());

if(employee2.getName().equals(employee.getName())) {

employeeService.deleteEmployeeById(employee2.getId());

}

}

return "redirect:/";

}

@GetMapping("/update")

public String updateEmployee(Model model) {

model.addAttribute("employee",new Employee());

return "UpdateEmployee";

}

@RequestMapping(value="/updatedEmployee", method = RequestMethod.POST)

public String updateEmployee(@ModelAttribute("employee") Employee employee){ List<Employee> listEmployees = employeeService.getAllEmployees();

System.out.println("-----------------------------------" + employee + "-------------------");

Employee emp = null;

for(Employee employee2 : listEmployees) {

System.out.println(employee2.getName());

if(employee2.getName().equals(employee.getName())) {

emp = employeeService.getEmployeeById(employee2.getId());

}

}

System.out.println( emp);

System.out.println(" emp.getId() );

employeeService.saveEmployee(emp);

return "redirect:/";

}

}

Entityclass.java

package com.employeeProject.spring.employeeproject.entity;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

@Entity

public class Employee {

@Emp\_id

@GeneratedValue(strategy=GenerationType.AUTO)

private long emp\_id;

private String emp\_name;

private String designation;

private int emp\_salary;

private int emp\_age;

public String getEmp\_name() {

return emp\_name;

}

public voemp\_id setEmp\_name(String emp\_name) {

this.emp\_name = emp\_name;

}

public String getDesignation() {

return designation;

}

public voemp\_id setDesignation(String designation) {

this.designation = designation;

}

public int getEmp\_salary() {

return emp\_salary;

}

public voemp\_id setEmp\_salary(int emp\_salary) {

this.emp\_salary = emp\_salary;

}

public int getEmp\_age() {

return emp\_age;

}

public voemp\_id setEmp\_age(int emp\_age) {

this.emp\_age = emp\_age;

}

public long getEmp\_id() {

return emp\_id;

}

@Overremp\_ide

public String toString() {

return "Employee [emp\_id=" + emp\_id + ", emp\_name=" + emp\_name + ", designation=" + designation + ", emp\_salary=" + emp\_salary

+ ", emp\_age=" + emp\_age + "]";

}

}

Employeerepository.java

package com.employeeProject.spring.employeeproject.repository;

import org.springframework.data.jpa.repository.JpaRepository;

import org.springframework.stereotype.Repository;

import com.employeeProject.spring.employeeproject.entity.Employee;

@Repository

public interface EmployeeRepository extends JpaRepository<Employee, Long>{

}

EmployeeService.java

package com.employeeProject.spring.employeeproject.services;

import java.util.List;

import org.springframework.data.domain.Page;

import com.employeeProject.spring.employeeproject.entity.Employee;

public interface EmployeeService {

List<Employee> getAllEmployees();

void saveEmployee(Employee employee);

Employee getEmployeeById(long emp\_ id);

void deleteEmployeeById(long emp\_id);

}

EmployeeServiceImple.java

package com.employeeProject.spring.employeeproject.services;

import java.util.List;

import java.util.Optional;

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

import org.springframework.data.domain.Page;

import org.springframework.stereotype.Service;

import com.employeeProject.spring.employeeproject.entity.Employee;

import com.employeeProject.spring.employeeproject.repository.EmployeeRepository;

@Service

public class EmployeeServiceImpl implements EmployeeService{

@Autowired

EmployeeRepository employeeRepository;

@Override

public List<Employee> getAllEmployees() {

return employeeRepository.findAll();

}

@Override

public void saveEmployee(Employee employee) {

employeeRepository.save(employee);

}

@Override

public Employee getEmployeeById(long emp\_ id) {

Optional<Employee> optional = employeeRepository.findById(emp\_id);

Employee employee = null;

if (optional.isPresent()) {

employee = optional.get();

} else {

throw new RuntimeException(" Employee not found for id :: " + emp\_id);

}

return employee;

}

@Override

public void deleteEmployeeById(long emp\_id) {

employeeRepository.deleteById(emp\_id);

}

}

EmployeeApp.java

package com.employeeProject.spring.employeeproject;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

import org.springframework.boot.CommandLineRunner;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.EnableAutoConfiguration;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.context.annotation.ComponentScan;

import com.employeeProject.spring.employeeproject.entity.Employee;

import com.employeeProject.spring.employeeproject.repository.EmployeeRepository;

@SpringBootApplication

public class EmployeeApp{

public static void main(String[] args) {

SpringApplication.run(EmployeeApp.class, args);

}}

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 https://maven.apache.org/xsd/maven-4.0.0.xsd"> |
|  | <modelVersion>4.0.0</modelVersion> |
|  | <parent> |
|  | <groupId>org.springframework.boot</groupId> |
|  | <artifactId>spring-boot-starter-parent</artifactId> |
|  | <version>2.6.4</version> |
|  | <relativePath/> <!-- lookup parent from repository --> |
|  | </parent> |
|  | <groupId>com.employeeProject.spring</groupId> |
|  | <artifactId>employee-project-two</artifactId> |
|  | <version>0.0.1-SNAPSHOT</version> |
|  | <name>employee-project-two</name> |
|  | <packaging>war</packaging> |
|  | <description>Demo project for Spring Boot</description> |
|  | <properties> |
|  | <java.version>11</java.version> |
|  | </properties> |
|  | <dependencies> |
|  | <dependency> |
|  | <groupId>org.springframework.boot</groupId> |
|  | <artifactId>spring-boot-starter-data-jpa</artifactId> |
|  | </dependency> |
|  | <dependency> |
|  | <groupId>org.springframework.boot</groupId> |
|  | <artifactId>spring-boot-starter-jdbc</artifactId> |
|  | </dependency> |
|  | <dependency> |
|  | <groupId>org.springframework.boot</groupId> |
|  | <artifactId>spring-boot-starter-web</artifactId> |
|  | </dependency> |
|  | <dependency> |
|  | <groupId>mysql</groupId> |
|  | <artifactId>mysql-connector-java</artifactId> |
|  | <scope>runtime</scope> |
|  | </dependency> |
|  | <dependency> |
|  | <groupId>org.springframework.boot</groupId> |
|  | <artifactId>spring-boot-starter-thymeleaf</artifactId> |
|  | </dependency> |
|  | <dependency> |
|  | <groupId>org.springframework.boot</groupId> |
|  | <artifactId>spring-boot-starter-test</artifactId> |
|  | <scope>test</scope> |
|  | </dependency> |
|  | </dependencies> |
|  |  |
|  | <build> |
|  | <plugins> |
|  | <plugin> |
|  | <groupId>org.springframework.boot</groupId> |
|  | <artifactId>spring-boot-maven-plugin</artifactId> |
|  | </plugin> |
|  | </plugins> |
|  | </build> |
|  |  |
|  | </project> |