Министерство образования Республики Беларусь

Министерство науки и высшего образования Российской Федерации

МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Автоматизированные системы управления»»

**Лабораторная работа № 6. Соединение с базой данных**

Выполнил: студент гр.ИСИТ-191

Харкевич А. В.

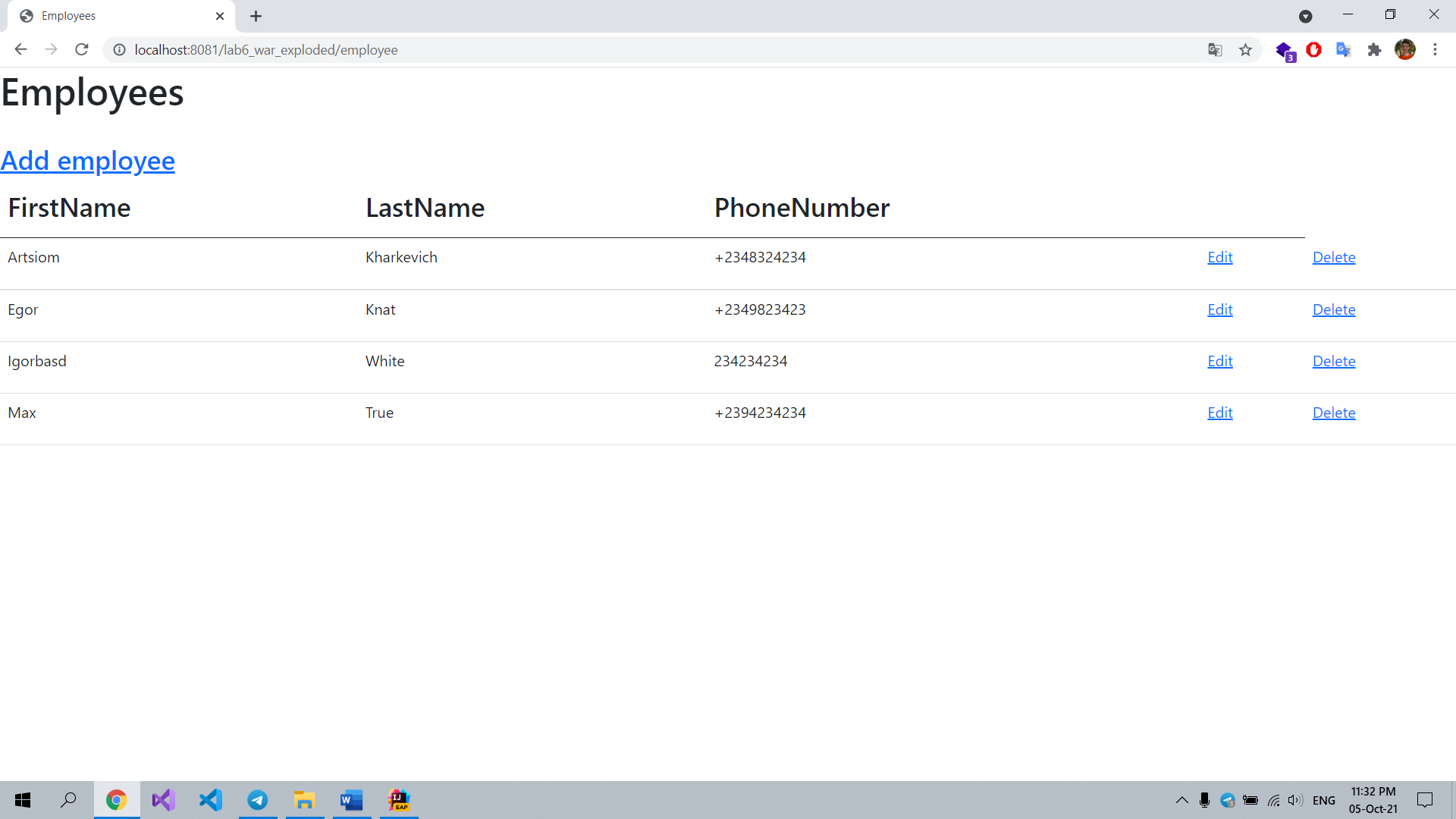
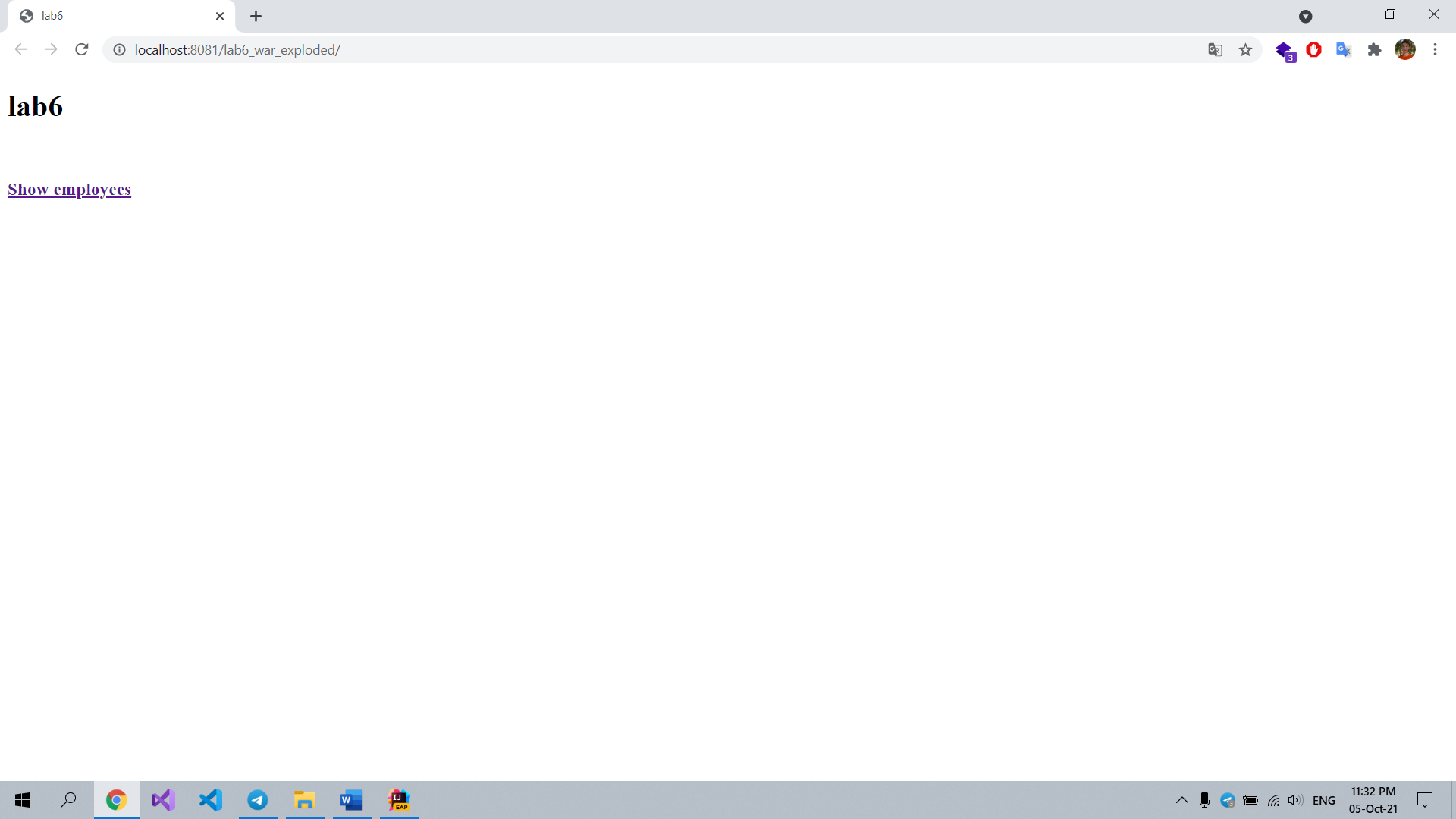
Проверил: Денисевич Д. А.

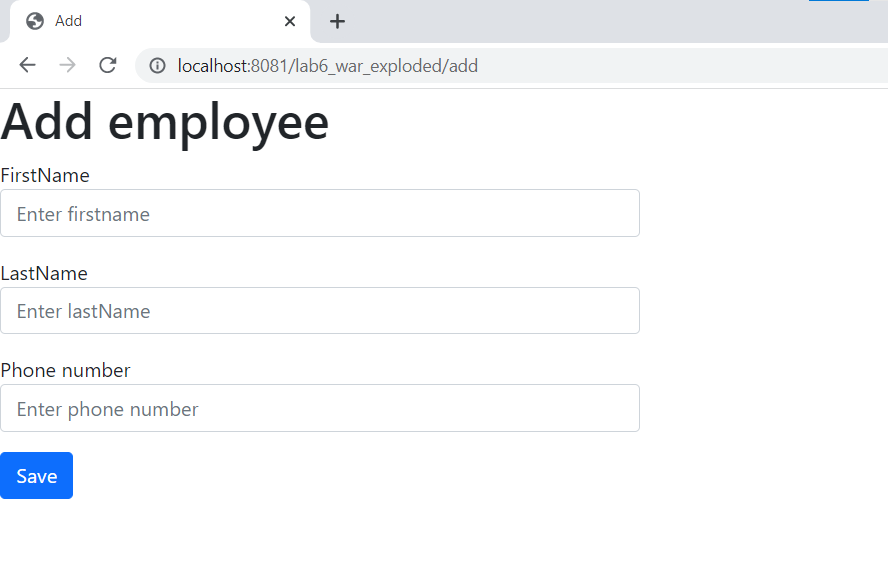
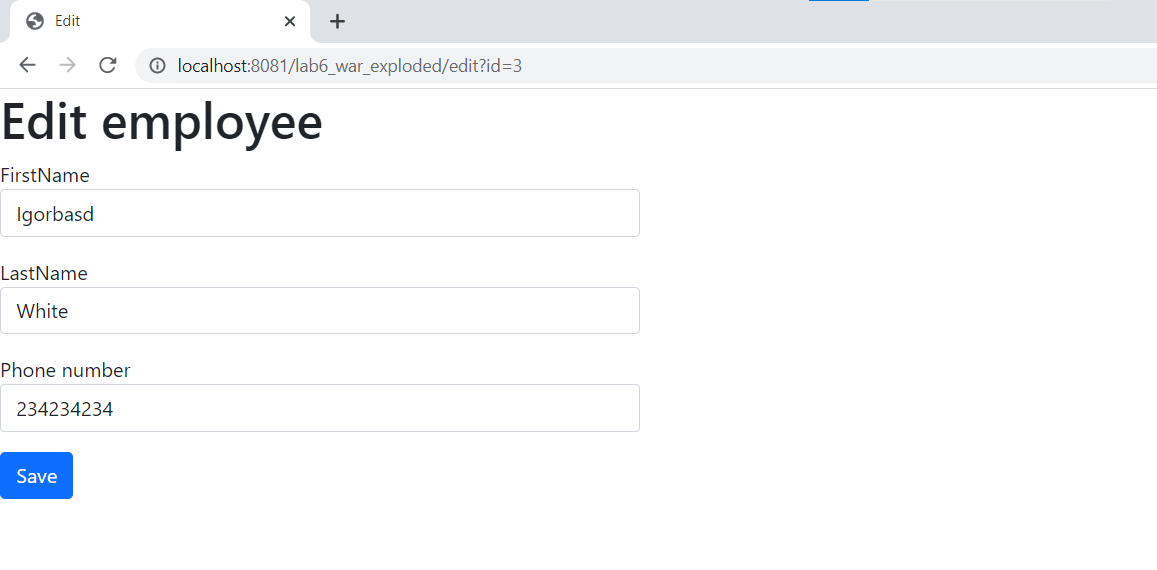
Могилев, 2021

**Цель работы:** приобретение навыков проектирования и разработки приложений в архитектуре клиент-сервер

**Задание:** в соответствии с заданием, выданным преподавателем, необходимо выполнить следующие действия.

1. Организацию соединения с базой данных вынести в отдельный класс, метод которого возвращает соединение.
2. Создать БД. Привести таблицы к одной из нормированных форм.
3. Создать класс для выполнения запросов на извлечение информации из БД с использованием компилированных запросов.
4. Создать класс на добавление информации.
5. Создать HTML-документ с полями для формирования запроса.
6. Результаты выполнения запроса передать клиенту в виде HTMLдокумента.





Employee.java

package lab6.lab6;

public class Employee {

private int id;

private String firstName;

private String lastName;

private String phoneNumber;

public void setFirstName(String firstName){

this.firstName = firstName;

}

public void setLastName(String lastName){

this.lastName = lastName;

}

public void setPhoneNumber(String phoneNumber){

this.phoneNumber = phoneNumber;

}

public void setId(int id){

this.id = id;

}

public String getFirstName(){

return firstName;

}

public String getLastName(){

return lastName;

}

public String getPhoneNumber(){

return phoneNumber;

}

public int getId(){

return id;

}

}

EmployeeRepository.java

package lab6.lab6;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.Statement;

import java.sql.\*;

import java.util.ArrayList;

public class EmployeeRepository {

private String url = "jdbc:sqlserver://localhost:1433;"

+ "database=lab6;"

+ "user=artsiom;"

+ "password=1234qwerQ!!;"

+ "encrypt=true;"

+ "trustServerCertificate=true;"

+ "loginTimeout=30;";

private String driver = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

public void Create(Employee employee) {

try {

Class.forName(driver).getDeclaredConstructor().newInstance();

try (Connection conn = DriverManager.getConnection(url)) {

String sql = "INSERT Employee(FirstName, LastName, PhoneNumber) VALUES(?, ?, ?)";

try (PreparedStatement preparedStatement = conn.prepareStatement(sql)) {

preparedStatement.setString(1, employee.getFirstName());

preparedStatement.setString(2, employee.getLastName());

preparedStatement.setString(3, employee.getPhoneNumber());

ResultSet resultSet = preparedStatement.executeQuery();

}

}

} catch (Exception ex) {

System.out.println(ex);

}

}

public Employee GetById(int id) {

Employee employee = new Employee();

try {

Class.forName(driver).getDeclaredConstructor().newInstance();

try (Connection conn = DriverManager.getConnection(url)) {

String sql = "SELECT \* FROM Employee WHERE id=?";

try (PreparedStatement preparedStatement = conn.prepareStatement(sql)) {

preparedStatement.setInt(1, id);

ResultSet resultSet = preparedStatement.executeQuery();

if (resultSet.next()) {

employee.setId(resultSet.getInt(1));

employee.setFirstName(resultSet.getString(2));

employee.setLastName(resultSet.getString(3));

employee.setPhoneNumber(resultSet.getString(4));

}

}

}

} catch (Exception ex) {

System.out.println(ex);

}

return employee;

}

public ArrayList<Employee> GetAll() {

ArrayList<Employee> employees = new ArrayList<Employee>();

try {

Class.forName(driver).getDeclaredConstructor().newInstance();

try (Connection conn = DriverManager.getConnection(url)) {

Statement statement = conn.createStatement();

ResultSet resultSet = statement.executeQuery("SELECT \* FROM Employee");

while (resultSet.next()) {

Employee employee = new Employee();

employee.setId(resultSet.getInt(1));

employee.setFirstName(resultSet.getString(2));

employee.setLastName(resultSet.getString(3));

employee.setPhoneNumber(resultSet.getString(4));

employees.add(employee);

}

}

} catch (Exception ex) {

System.out.println(ex);

}

return employees;

}

public void Update(int id, Employee employee) {

try {

Class.forName(driver).getDeclaredConstructor().newInstance();

try (Connection conn = DriverManager.getConnection(url)) {

String sql = "UPDATE Employee SET Employee.FirstName=?, Employee.LastName=?, Employee.PhoneNumber=? WHERE id=?";

try (PreparedStatement preparedStatement = conn.prepareStatement(sql)) {

preparedStatement.setString(1, employee.getFirstName());

preparedStatement.setString(2, employee.getLastName());

preparedStatement.setString(3, employee.getPhoneNumber());

preparedStatement.setInt(4, id);

ResultSet resultSet = preparedStatement.executeQuery();

}

}

} catch (Exception ex) {

System.out.println(ex);

}

}

public void Delete(int id) {

try {

Class.forName(driver).getDeclaredConstructor().newInstance();

try (Connection conn = DriverManager.getConnection(url)) {

String sql = "DELETE Employee WHERE id=?";

try (PreparedStatement preparedStatement = conn.prepareStatement(sql)) {

preparedStatement.setInt(1, id);

ResultSet resultSet = preparedStatement.executeQuery();

}

}

} catch (Exception ex) {

System.out.println(ex);

}

}

}

EmployeeService.java

package lab6.lab6;

import java.util.ArrayList;

public class EmployeeService {

private EmployeeRepository employeeRepository = new EmployeeRepository();

public Employee GetById(int id) {

return employeeRepository.GetById(id);

}

public ArrayList<Employee> GetAll() {

return employeeRepository.GetAll();

}

public void Update(int id, Employee employee) {

employeeRepository.Update(id, employee);

}

public void Delete(int id) {

employeeRepository.Delete(id);

}

public void Create(Employee employee) {

employeeRepository.Create(employee);

}

}

ShowEmployeeServlet.java

package lab6.lab6;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.ArrayList;

@WebServlet(name = "employee", value = "/employee")

public class ShowEmployeeServlet extends HttpServlet {

private EmployeeService employeeService = new EmployeeService();

public void doGet(HttpServletRequest request, HttpServletResponse response) throws IOException, ServletException {

response.setContentType("text/html");

ArrayList<Employee> employees = employeeService.GetAll();

request.setAttribute("employees", employees);

getServletContext().getRequestDispatcher("/ShowEmployee.jsp").forward(request, response);

}

public void destroy() {

}

}

AddEmployeeServlet.java

package lab6.lab6;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import java.io.IOException;

@WebServlet(name = "add", value = "/add")

public class AddEmployeeServlet extends HttpServlet {

private EmployeeService employeeService = new EmployeeService();

public void doGet(HttpServletRequest request, HttpServletResponse response) throws IOException, ServletException {

response.setContentType("text/html");

getServletContext().getRequestDispatcher("/AddEmployee.jsp").forward(request, response);

}

public void doPost(HttpServletRequest request, HttpServletResponse response) throws IOException, ServletException {

response.setContentType("text/html");

Employee employee = new Employee();

employee.setFirstName(request.getParameter("firstName"));

employee.setLastName(request.getParameter("lastName"));

employee.setPhoneNumber(request.getParameter("phoneNumber"));

employeeService.Create(employee);

response.sendRedirect("employee");

}

public void destroy() {

}

}

DeleteEmployeeServlet.java

package lab6.lab6;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import java.io.IOException;

@WebServlet(name = "delete", value = "/delete")

public class DeleteEmployeeServlet extends HttpServlet {

private EmployeeService employeeService = new EmployeeService();

public void doGet(HttpServletRequest request, HttpServletResponse response) throws IOException, ServletException {

response.setContentType("text/html");

employeeService.Delete(Integer.parseInt(request.getParameter("id")));

response.sendRedirect("employee");

}

public void destroy() {

}

}

EditEmployeeServlet.java

package lab6.lab6;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import java.io.IOException;

import java.util.ArrayList;

@WebServlet(name = "edit", value = "/edit")

public class EditEmployeeServlet extends HttpServlet {

private EmployeeService employeeService = new EmployeeService();

private int id;

public void doGet(HttpServletRequest request, HttpServletResponse response) throws IOException, ServletException {

response.setContentType("text/html");

id = Integer.parseInt(request.getParameter("id"));

Employee employee = employeeService.GetById(id);

request.setAttribute("employee", employee);

getServletContext().getRequestDispatcher("/EditEmployee.jsp").forward(request, response);

}

public void doPost(HttpServletRequest request, HttpServletResponse response) throws IOException, ServletException {

response.setContentType("text/html");

Employee employee = new Employee();

employee.setFirstName(request.getParameter("firstName"));

employee.setLastName(request.getParameter("lastName"));

employee.setPhoneNumber(request.getParameter("phoneNumber"));

employeeService.Update(id, employee);

response.sendRedirect("employee");

}

public void destroy() {

}

}

ShowEmployee.jsp

<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

<%@ page contentType="text/html;charset=UTF-8" language="java" %>

<%@page import="lab6.lab6.ShowEmployeeServlet"%>

<html>

<head>

<title>Employees</title>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"

integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3AhiU" crossorigin="anonymous">

</head>

<body>

<h1>Employees</h1>

</br>

<h3><a href="add">Add employee</a></h3>

<table class="table">

<thead>

<tr>

<th>

<h3>FirstName</h3>

</th>

<th>

<h3>LastName</h3>

</th>

<th>

<h3>PhoneNumber</h3>

</th>

<th></th>

</tr>

</thead>

<tbody>

<c:forEach var="employee" items="${employees}">

<tr>

<td>

<c:out value="${employee.firstName}"/>

</td>

<td>

<p><c:out value="${employee.lastName}"/></p>

</td>

<td>

<p><c:out value="${employee.phoneNumber}"/></p>

</td>

<td>

<a href="edit?id=${employee.id}">Edit</a>

</td>

<td>

<a href="delete?id=${employee.id}">Delete</a>

</td>

</c:forEach>

</tbody>

</table>

</body>

</html>

AddEmployee.jsp

<%@ page contentType="text/html;charset=UTF-8" language="java" %>

<html>

<head>

<title>Add</title>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"

integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3AhiU" crossorigin="anonymous">

</head>

<body>

<h1>Add employee</h1>

<form action="add" method="post">

<div class="form-group">

<div class="col-md-4 mb-3">

<label for="firstname">FirstName</label>

<input class="form-control mb" id="firstname" name="firstName" placeholder="Enter firstname">

</div>

</div>

<div class="form-group">

<div class="col-md-4 mb-3">

<label for="lastName">LastName</label>

<input class="form-control mb" id="lastName" name="lastName" placeholder="Enter lastName">

</div>

</div>

<div class="form-group">

<div class="col-md-4 mb-3">

<label for="phoneNumber">Phone number</label>

<input class="form-control mb" id="phoneNumber" name="phoneNumber" placeholder="Enter phone number">

</div>

</div>

<button type="submit" class="btn btn-primary">Save</button>

</form>

</body>

</html>

EditEmployee.jsp

<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

<%@ page contentType="text/html;charset=UTF-8" language="java" %>

<html>

<head>

<title>Edit</title>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"

integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3AhiU" crossorigin="anonymous">

</head>

<body>

<h1>Edit employee</h1>

<form action="edit" method="post">

<div class="form-group">

<div class="col-md-4 mb-3">

<label for="firstname">FirstName</label>

<input class="form-control mb" id="firstname" name="firstName" placeholder="Enter firstname" value="${employee.firstName}">

</div>

</div>

<div class="form-group">

<div class="col-md-4 mb-3">

<label for="lastName">LastName</label>

<input class="form-control mb" id="lastName" name="lastName" placeholder="Enter lastName" value="${employee.lastName}">

</div>

</div>

<div class="form-group">

<div class="col-md-4 mb-3">

<label for="phoneNumber">Phone number</label>

<input class="form-control mb" id="phoneNumber" name="phoneNumber" placeholder="Enter phone number"

value="${employee.phoneNumber}">

</div>

</div>

<button type="submit" class="btn btn-primary">Save</button>

</form>

</body>

</html>