ГУАП КАФЕДРА № 43

ОТЧЕТ

ЗАЩИЩЕН С ОЦЕНКОЙ ПРЕПОДАВАТЕЛЬ

|  |  |  |
| --- | --- | --- |
| ассистент |  | П.А.Степанов |
| должность, уч. степень, звание | подпись, дата | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №1 |
| «Разработка простого серверного приложения c использованием сервлетов и фильтров» |
| по дисциплине: ТЕХНОЛОГИЯ РАЗРАБОТКИ СЕРВЕРНЫХ ИНФОРМАЦИОННЫХ СИСТЕМ |

РАБОТУ ВЫПОЛНИЛА

|  |  |  |
| --- | --- | --- |
| СТУДЕНТКА ГР. | 4232 | Е.А.Уткина |
|  | подпись, дата | инициалы, фамилия |

Санкт-Петербург

2024

1. Цель работы:

Разработать простое веб-приложение, публикующее себя на веб-сервере.

1. Задание:

Выполните следующие задачи:

1. В соответствии со своим вариантом разработайте набор из минимум двух экранных форм приложения
2. Соберите проект веб-приложения (war) на Maven (можно без использования Spring)
3. Реализуйте формы средствами сервлетов. Проект должен как минимум содержать формы просмотра, добавления и удаления данных. Не используйте технологию JSP (Java Server Pages)
4. Аргументируйте почему были выбраны те или иные запросы HTTP
5. Использовать базу данных можно, но не обязательно

Вариант 14: Сдача недвижимости в аренду.

1. Листинг программы:

Класс AddApartmentServlet:

package servlet;  
  
import entities.Apartment;  
import repository.ApartmentRepository;  
import java.io.IOException;  
import jakarta.servlet.annotation.WebServlet;  
import jakarta.servlet.http.\*;  
  
@WebServlet("/addApartment")  
public class AddApartmentServlet extends HttpServlet {  
 private static final String *HEADER\_HTML* = "<html><body><h2>Добавить квартиру для аренды</h2>\n";  
 private static final String *FORM\_HTML* = "<form method='post' action='addApartment' target='\_self'>"  
 + "Город: <input name='city'><br>"  
 + "Адрес: <input name='address'><br>"  
 + "Стоимость, рублей/мес: <input name='price' type='number'><br>"  
 + "<input type='submit' value='Отправить'>"  
 + "</form></body></html>";  
  
 @Override  
 protected void doGet(HttpServletRequest request, HttpServletResponse response) throws IOException {  
 renderResponse(response, null, null, null);  
 }  
  
 @Override  
 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws IOException {  
 String city = request.getParameter("city");  
 String address = request.getParameter("address");  
 String price = request.getParameter("price");  
  
 if (isValidInput(city, address, price)) {  
 Apartment apartment = new Apartment(city, address, price);  
 ApartmentRepository.*addApartment*(apartment);  
 }  
  
 renderResponse(response, city, address, price);  
 }  
  
 private void renderResponse(HttpServletResponse response, String city, String address, String price) throws IOException {  
 response.setStatus(HttpServletResponse.*SC\_OK*);  
 response.getWriter().write(*HEADER\_HTML*);  
  
 if (city != null && address != null && price != null) {  
 response.getWriter().write("<p>Вы добавили квартиру:</p>");  
 response.getWriter().write("<p>Город: " + city + "</p>");  
 response.getWriter().write("<p>Адрес: " + address + "</p>");  
 response.getWriter().write("<p>Цена: " + price + " рублей/мес</p>");  
 }  
  
 response.getWriter().write("<h3>Список доступных квартир:</h3>");  
 if (ApartmentRepository.*getApartments*().isEmpty()) {  
 response.getWriter().write("<p>Квартиры не найдены.</p>");  
 } else {  
 ApartmentRepository.*getApartments*().forEach(apartment -> {  
 try {  
 response.getWriter().write("<p>Город: " + apartment.getCity()  
 + ", Адрес: " + apartment.getAddress()  
 + ", Стоимость: " + apartment.getPrice() + " руб/мес</p>");  
 } catch (IOException e) {  
 throw new RuntimeException(e);  
 }  
 });  
 }  
  
 response.getWriter().write(*FORM\_HTML*);  
 }  
  
 private boolean isValidInput(String city, String address, String price) {  
 return city != null && address != null && price != null;  
 }  
}

Класс DeletePatientServlet:

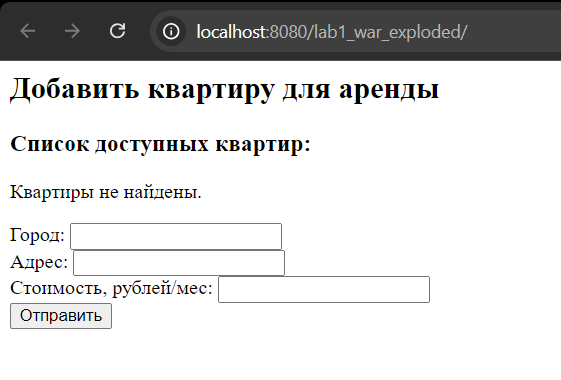
package servlet;  
  
import entities.Apartment;  
import repository.ApartmentRepository;  
import java.io.IOException;  
import jakarta.servlet.annotation.WebServlet;  
import jakarta.servlet.http.\*;  
  
@WebServlet("/deleteApartment")  
public class DeleteApartmentServlet extends HttpServlet {  
 private static final String *HEADER\_HTML* = "<html><body><h2>Удаление квартиры</h2>\n";  
 private static final String *FORM\_HTML* = "<form method='post' action='deleteApartment' target='\_self'>"  
 + "ID квартиры: <input name='apartmentId' type='number'><br>"  
 + "<input type='submit' value='Удалить'>"  
 + "</form></body></html>";  
  
 @Override  
 protected void doGet(HttpServletRequest request, HttpServletResponse response) throws IOException {  
 handleRequest(response, null, null);  
 }  
  
 @Override  
 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws IOException {  
 String apartmentIdStr = request.getParameter("apartmentId");  
 if (apartmentIdStr != null) {  
 try {  
 int apartmentId = Integer.*parseInt*(apartmentIdStr);  
  
 if (isValidApartmentId(apartmentId)) {  
 ApartmentRepository.*deleteApartment*(apartmentId);  
 handleRequest(response, apartmentIdStr, null);  
 } else {  
 handleRequest(response, null, "Квартиры с данным ID не существует.");  
 }  
 } catch (NumberFormatException e) {  
 handleRequest(response, null, "Некорректный ID. Пожалуйста, введите число.");  
 }  
 } else {  
 handleRequest(response, null, "ID квартиры не был введён.");  
 }  
 }  
  
 private void handleRequest(HttpServletResponse response, String apartmentId, String errorMessage) throws IOException {  
 response.setStatus(HttpServletResponse.*SC\_OK*);  
 response.getWriter().write(*HEADER\_HTML*);  
  
 if (errorMessage != null) {  
 response.getWriter().write("<p style='color:red;'>" + errorMessage + "</p>");  
 } else if (apartmentId != null) {  
 response.getWriter().write("<p>Квартира с ID " + apartmentId + " успешно удалена.</p>");  
 }  
  
 response.getWriter().write("<h3>Список доступных квартир:</h3>");  
 if (ApartmentRepository.*getApartments*().isEmpty()) {  
 response.getWriter().write("<p>Нет доступных квартир.</p>");  
 } else {  
 for (int i = 0; i < ApartmentRepository.*getApartments*().size(); i++) {  
 Apartment apartment = ApartmentRepository.*getApartments*().get(i);  
 response.getWriter().write("<p>ID: " + i + ", Город: " + apartment.getCity()  
 + ", Адрес: " + apartment.getAddress() + ", Цена: " + apartment.getPrice() + " рублей/мес</p>");  
 }  
 }  
  
 response.getWriter().write(*FORM\_HTML*);  
 }  
  
 private boolean isValidApartmentId(int id) {  
 return id >= 0 && id < ApartmentRepository.*getApartments*().size();  
 }  
}

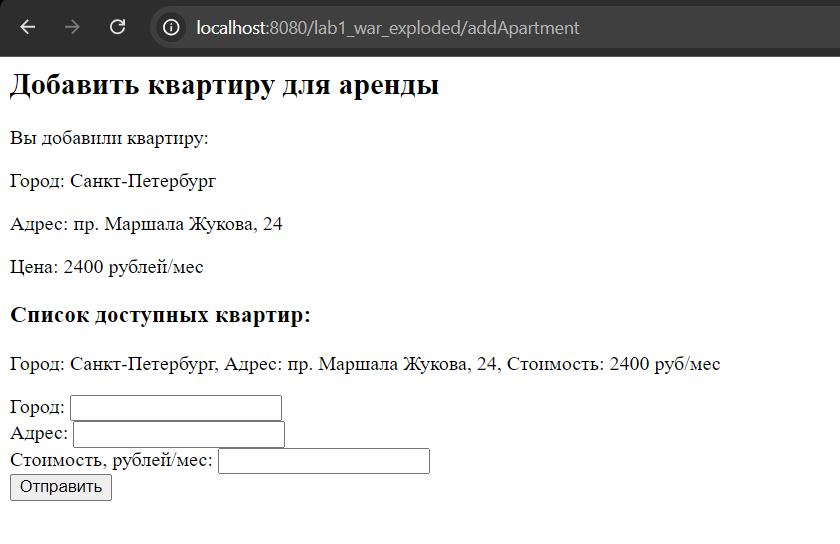
Класс EncodingFilter:

package filter;  
  
import jakarta.servlet.\*;  
  
import java.io.IOException;  
  
public class EncodingFilter implements Filter {  
 @Override  
 public void doFilter(ServletRequest servletRequest, ServletResponse servletResponse,  
 FilterChain filterChain) throws IOException, ServletException {  
 filterChain.doFilter(servletRequest, servletResponse);  
 servletResponse.setContentType("text/html;charset=UTF-8");  
 }  
}

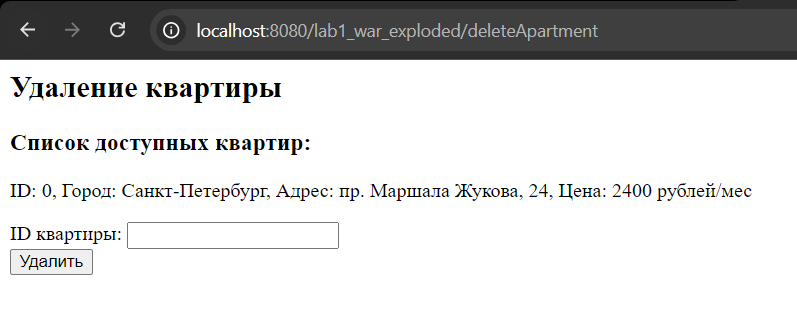
1. Пример выполнения программы:

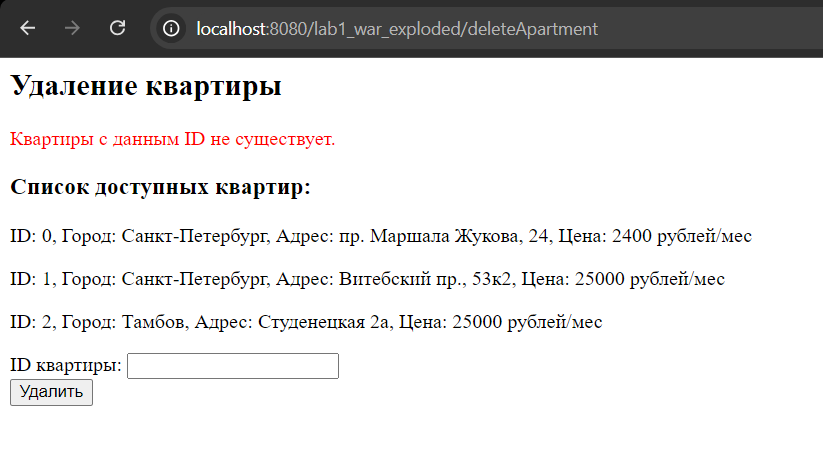
Форма добавления квартиры:

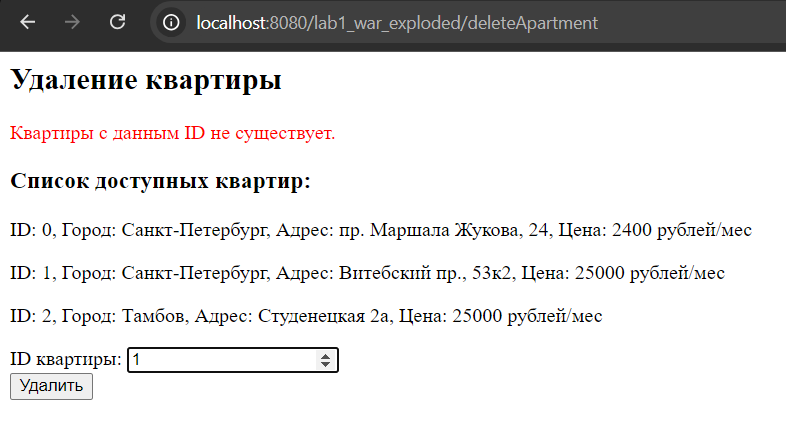


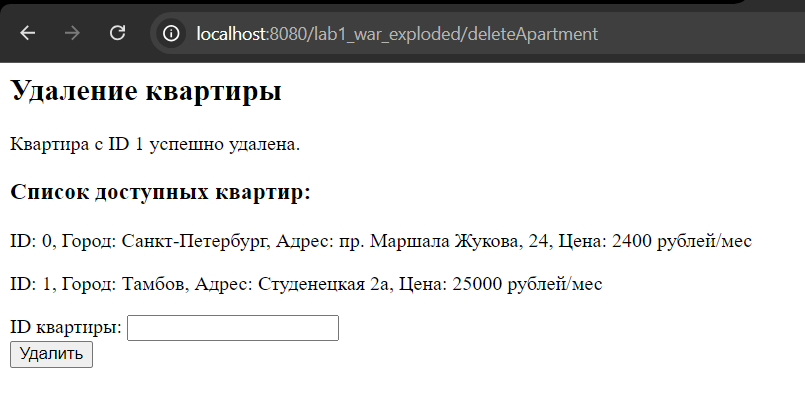


Форма удаления квартиры:









1. Вывод:

Разработано простое веб-приложение, публикующее себя на веб-сервере с использование сервлетов.