МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

ИНСТИТУТ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ И ПРОГРАММИРОВАНИЯ

КАФЕДРА компьютерных технологий и программной инженерии

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ассистент |  |  |  | Кочин Д.А. |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №1 |
| «Разработка простого серверного приложения J2EE с использованием сервлетов» |
| по курсу: Технологии разработки серверных информационных систем |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4932 |  | 18.06.21 |  | Н.С. Иванов |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2021

## Цель работы

Познакомится с основами написания веб приложений с использованием servlet.

## Задание:

В соответствии со своим вариантом разработайте набор экранных форм приложения, соберите проект веб-приложения (war) на Maven, реализуйте формы средствами servlet. Проект должен как минимум содержать формы просмотра, добавления и удаления данных.

Вариант: 8. Учет трат в бюджете семьи.

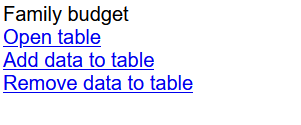
## Описание разрабатываемого продукта:

Веб-приложение для учета трат в бюджете семьи.

Функционал: добавление, удаление, просмотр информации.

### Формы:

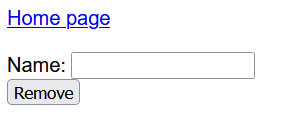
#### Главное окно



#### Форма добавления



#### Форма удаления



#### Форма просмотра



## Выводы

* Servlet можно использовать для написания простых веб приложений и обработки HTTP/HTTPS запросов, однако для более сложных вещей больше подходят фреймворки, такие как spring.

## Приложение:

### // @filename \HomePageServlet.java

import Model.Costs;

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;

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

public class HomePageServlet extends HttpServlet {

private final Costs costs = Costs.getInstance();

@Override

protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

resp.setContentType("text/html");

PrintWriter out = resp.getWriter();

out.write(""

+ "<html><body>"

+ "<label>Family budget</label>"

+ "<br>"

+ "<a href=\"table-view-servlet\">Open table</a>"

+ "<br>"

+ "<a href=\"table-add-data-servlet\">Add data to table</a>"

+ "<br>"

+ "<a href=\"table-remove-data-servlet\">Remove data to table</a>"

+ ""

+ "</body></html>"

);

out.close();

resp.setStatus(200);

}

}

### // @filename \TableAddDataServlet.java

import Model.Costs;

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 javax.servlet.http.HttpSession;

import java.io.IOException;

import java.io.PrintWriter;

@WebServlet(name = "tableAddDataServlet", value = "/table-add-data-servlet")

public class TableAddDataServlet extends HttpServlet {

private final Costs costs = Costs.getInstance();

@Override

protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

PrintWriter out = resp.getWriter();

resp.setContentType("text/html");

out.write(""

+ "<head>"

+ " <meta charset=\"UTF-8\">"

+ " <title>Add cost</title>"

+ "</head>"

+ "<body>"

+ " <a href=\"/\">Home page</a>"

+ " <br>"

+ " <br>"

+ "<form name=\"AddForm\" method=\"post\" action=\"table-add-data-servlet\">"

+ " Name: <input type=\"text\" name=\"name\"/> <br/>"

+ " Price: <input type=\"text\" name=\"price\"/> <br/>"

+ " <input type=\"submit\" value=\"Add\" />"

+ "</form>"

+ ""

+ "</body>"

);

out.close();

}

@Override

protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

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

String price = req.getParameter("price");

try {

costs.addItem(name, Integer.parseInt(price));

} catch (Exception e) {

e.printStackTrace();

}

resp.sendRedirect("/");

// getServletContext().getRequestDispatcher("/hello").forward(req, resp);

}

}

### // @filename \TableRemoveDataServlet.java

import Model.Costs;

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 javax.servlet.http.HttpSession;

import java.io.IOException;

import java.io.PrintWriter;

@WebServlet(name = "tableRemoveDataServlet", value = "/table-remove-data-servlet")

public class TableRemoveDataServlet extends HttpServlet {

private final Costs costs = Costs.getInstance();

@Override

protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

PrintWriter out = resp.getWriter();

resp.setContentType("text/html");

out.write(""

+ "<head>"

+ " <meta charset=\"UTF-8\">"

+ " <title>Remove cost</title>"

+ "</head>"

+ "<body>"

+ " <a href=\"/\">Home page</a>"

+ " <br>"

+ " <br>"

+ "<form name=\"RemoveForm\" method=\"post\" action=\"table-remove-data-servlet\">"

+ " Name: <input type=\"text\" name=\"name\"/> <br/>"

+ " <input type=\"submit\" value=\"Remove\" />"

+ "</form>"

+ ""

+ "</body>"

);

out.close();

}

@Override

protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

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

try {

costs.rmItem(name);

} catch (Exception e) {

e.printStackTrace();

}

resp.sendRedirect("/");

// getServletContext().getRequestDispatcher("/hello").forward(req, resp);

}

}

### // @filename \TableViewServlet.java

import Model.Costs;

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 javax.servlet.http.HttpSession;

import java.io.IOException;

import java.io.PrintWriter;

@WebServlet(name = "tableViewServlet", value = "/table-view-servlet")

public class TableViewServlet extends HttpServlet {

private final Costs costs = Costs.getInstance();

@Override

protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

PrintWriter out = resp.getWriter();

String username = req.getParameter("username");

resp.setContentType("text/html");

out.write(""

+ "<head>"

+ " <meta charset=\"UTF-8\">"

+ " <title>Budget costs</title>"

+ "</head>"

+ "<body>"

+ " <a href=\"/\">Home page</a>"

+ " <br>"

+ " <br>"

+ " <table border=\"1\">"

+ " <caption> Cost items </caption>"

+ ""

+ " <tr>"

+ " <th>Name</th>"

+ " <th>Price</th>"

+ " </tr>"

);

for (int i = 0; i < costs.getNames().size(); i++) {

out.write(""

+ " <tr>"

+ " <th>"+ costs.getNames().get(i) +"</th>"

+ " <th>"+ costs.getPrices().get(i) +"</th>"

+ " </tr>"

+""

);

}

out.write(""

+ ""

+ " </table>"

+ ""

+ "</body>"

);

out.close();

}

}

### // @filename \Model\Costs.java

package Model;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.LinkedList;

import java.util.List;

public class Costs {

private static volatile Costs instance;

private Costs(){}

public static Costs getInstance() {

Costs result = instance;

if (result != null) {

return result;

}

synchronized(Costs.class) {

if (instance == null) {

instance = new Costs();

}

return instance;

}

}

private Integer totalCost = 0;

private final ArrayList<String> names = new ArrayList<String>();

private final ArrayList<Integer> prices = new ArrayList<Integer>();

public void addItem(String name, Integer price) throws Exception {

if(price < 0){

throw new Exception("Price must be positive");

}

else if(names.contains(name)){

return;

}

totalCost += price;

names.add(name);

prices.add(price);

}

public void rmItem(String name) {

int index = names.indexOf(name);

if(index < 0) { return; }

totalCost -= prices.get(index);

names.remove(index);

prices.remove(index);

}

public List<String> getNames(){

return names;

}

public List<Integer> getPrices(){

return prices;

}

}