ГУАП

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| кандидат технических наук |  |  |  | П. А. Степанов |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №1 |
| Разработка простого серверного приложения J2EE c использованием сервлетов |
| по курсу: Технология разработки серверных информационных систем |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4134к |  |  |  | Столяров Н.С. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2023

**Текст и вариант задания:**

19 вариант

База данных с рецептами блюд

**Описание разрабатываемого продукта:**

В данной программе создан сервис по работе с рецептами блюд (вывод, добавление, удаление и поиск)

**Текст основных фрагментов кода :**

package com**.**mycompany**.**lab1**.**servlets**;**

**import** com**.**mycompany**.**lab1**.**base**.**base**;**

**import** java**.**io**.**IOException**;**

**import** java**.**io**.**PrintWriter**;**

**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**.**nio**.**file**.**Files**;**

**import** java**.**nio**.**file**.**Paths**;**

**import** java**.**nio**.**file**.**StandardOpenOption**;**

**import** java**.**io**.\*;**

@WebServlet**(**name **=** "DeleteCreateServlet"**,** urlPatterns **=** **{**"/manage"**})**

public class DeleteCreateServlet **extends** HttpServlet **{**

protected void processRequest**(**HttpServletRequest request**,** HttpServletResponse response**)**

**throws** ServletException**,** IOException **{**

String find\_name **=** request**.**getParameter**(**"name"**);**

response**.**setContentType**(**"text/html;charset=UTF-8"**);**

**try** **(**PrintWriter out **=** response**.**getWriter**())** **{**

out**.**println**(**"<!DOCTYPE html>"**);**

out**.**println**(**"<html>"**);**

out**.**println**(**"<head>"**);**

out**.**println**(**"<style>"**);**

out**.**println**(**"tr, th {border: 1px solid #000000}"**);**

out**.**println**(**"</style>"**);**

out**.**println**(**"</head>"**);**

out**.**println**(**"<body>"**);**

out**.**println**(**"<h1>Результаты поиска</h1>"**);**

out**.**println**(**"<table>"**);**

out**.**println**(**"<tr>"**);**

out**.**println**(**"<th><p>Блюдо</p></th>"**);**

out**.**println**(**"<th><p>Рецепт</p></th>"**);**

out**.**println**(**"</tr>"**);**

**if** **(**find\_name **!=** **null** **&&** **!**find\_name**.**isEmpty**())** **{**

File file **=** **new** File**(**"base.txt"**);**

FileReader fr **=** **new** FileReader**(**file**);**

BufferedReader reader **=** **new** BufferedReader**(**fr**);**

String line **=** reader**.**readLine**();**

String recept **=** ""**;**

**while** **(**line **!=** **null)** **{**

recept **=** reader**.**readLine**();**

**if** **(**line**.**contains**(**find\_name**))** **{**

out**.**println**(**"<tr>"**);**

out**.**println**(**"<th><p>"**);**

out**.**println**(**line**);**

out**.**println**(**"</p></th>"**);**

out**.**println**(**"<th><p>"**);**

out**.**println**(**recept**);**

out**.**println**(**"</p></th>"**);**

out**.**println**(**"</tr>"**);**

**}**

line **=** reader**.**readLine**();**

}

} else {

out.println("<h1>Пустой запрос</h1>");

}

out.println("</table>");

out.println("</body>");

out.println("</html>");

}

}

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

File sourceFile = new File("base.txt");

File outputFile = new File("base\_buff.txt");

String find\_name = request.getParameter("name");

BufferedReader reader = new BufferedReader(new FileReader(sourceFile));

BufferedWriter writer = new BufferedWriter(new FileWriter(outputFile));

String line = "";

boolean find = false;

while ((line = reader.readLine()) != null) {

if (find) {

find = false;

} else {

if (!line.contains(find\_name)) {

writer.write(line);

writer.newLine();

} else {

find = true;

}

}

}

reader.close();

writer.close();

sourceFile.delete();

outputFile.renameTo(sourceFile);

response.sendRedirect(request.getContextPath() + "/list");

}

}

package com**.**mycompany**.**lab1**.**servlets**;**

**import** com**.**mycompany**.**lab1**.**base**.**base**;**

**import** java**.**io**.**IOException**;**

**import** java**.**io**.**PrintWriter**;**

**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**.**BufferedReader**;**

**import** java**.**io**.**File**;**

**import** java**.**io**.**FileNotFoundException**;**

**import** java**.**io**.**FileReader**;**

**import** java**.**io**.**IOException**;**

**import** java**.**nio**.**file**.**Files**;**

**import** java**.**nio**.**file**.**Paths**;**

**import** java**.**nio**.**file**.**StandardOpenOption**;**

@WebServlet**(**name **=** "ExampleServlet"**,** urlPatterns **=** **{**"/list"**})**

public class ExampleServlet **extends** HttpServlet **{**

String data**[]** **=** **new** String**[**10**];**

protected void processRequest**(**HttpServletRequest request**,** HttpServletResponse response**)**

**throws** ServletException**,** IOException **{**

response**.**setContentType**(**"text/html;charset=UTF-8"**);**

**try** **(**PrintWriter out **=** response**.**getWriter**())** **{**

out**.**println**(**"<!DOCTYPE html>"**);**

out**.**println**(**"<html>"**);**

out**.**println**(**"<head>"**);**

out**.**println**(**"<style>"**);**

out**.**println**(**"tr, th {border: 1px solid #000000}"**);**

out**.**println**(**"</style>"**);**

out**.**println**(**"</head>"**);**

out**.**println**(**"<body>"**);**

out**.**println**(**"<form method='get' action='/manage'>"**);**

out**.**println**(**"<input type='text' name='name' placeholder='Название'>"**);**

out**.**println**(**"<input type='submit' value='Найти'>"**);**

out**.**println**(**"</form>"**);**

out**.**println**(**"<br>"**);**

out**.**println**(**"<form method='post' action='/manage'>"**);**

out**.**println**(**"<input type='text' name='name' placeholder='Название'>"**);**

out**.**println**(**"<input type='submit' value='Удалить'>"**);**

out**.**println**(**"</form>"**);**

out**.**println**(**"<br>"**);**

out**.**println**(**"<form method='post' action='/list'>"**);**

out**.**println**(**"<input type='text' name='name' placeholder='Название'>"**);**

out**.**println**(**"<input type='text' name='recept' placeholder='Рецепт'>"**);**

out**.**println**(**"<input type='submit' value='Добавить блюдо'>"**);**

out**.**println**(**"</form>"**);**

out**.**println**(**"<br>"**);**

out**.**println**(**"<table>"**);**

out**.**println**(**"<tr>"**);**

out**.**println**(**"<th><p>Блюдо</p></th>"**);**

out**.**println**(**"<th><p>Рецепт</p></th>"**);**

out.println("<th></th>");

out.println("</tr>");

try {

File file = new File("base.txt");

//создаем объект FileReader для объекта File

FileReader fr = new FileReader(file);

//создаем BufferedReader с существующего FileReader для построчного считывания

BufferedReader reader = new BufferedReader(fr);

// считаем сначала первую строку

String line = reader.readLine();

while (line != null) {

out.println("<tr>");

out.println("<th><p>");

out.println(line);

out.println("</p></th>");

line = reader.readLine();

out.println("<th><p>");

out.println(line);

out.println("</p></th>");

out.println("<th><p>Удалить</p></th>");

out.println("</tr>");

line = reader.readLine();

}

} catch (FileNotFoundException e) {

e.printStackTrace();

} catch (IOException e) {

e.printStackTrace();

}

out.println("</table>");

out.println("</body>");

out.println("</html>");

}

}

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

String name = request.getParameter("name") + "\n";

String recept = request.getParameter("recept") + "\n";

if (name != null && !name.isEmpty() && recept != null && !recept.isEmpty()) {

try {

Files.write(Paths.get("base.txt"), name.getBytes(),StandardOpenOption.APPEND);

Files.write(Paths.get("base.txt"), recept.getBytes(), StandardOpenOption.APPEND);

}

catch (IOException e) {

System.out.println(e);

}

}

processRequest(request, response);

}

}