**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

Кафедра вычислительной техники

**Отчет по лабораторной работе №3  
по дисциплине «Web-программирование»**

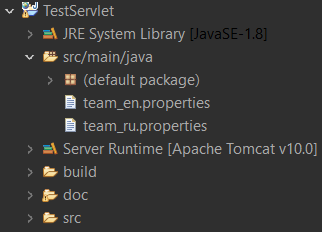
**Тема: интернационализация web-приложений**

|  |  |
| --- | --- |
| Студент гр. 2310 | Альсакма О.С.М |
| Преподаватель | Павловский М.Г. |

**Цель работы:** знакомство со способами отображения данных на раз-

личных языках при использовании файлов ресурсов

Было создано два файла .properties: с русским и английский языком



//team\_en.properties

title = There is list of footballers

name = Name

spec = Specialization

city = City

salary = Salary

condition = with salary

//team\_ru.properties

title = \u0412\u043E\u0442 \u043C\u043E\u0438 \u0444\u0443\u0442\u0431\u043E\u043B\u0438\u0441\u0442\u044B

name = \u0418\u043C\u044F \u0424\u0430\u043C\u0438\u043B\u0438\u044F

spec = \u0421\u043F\u0435\u0446\u0438\u0430\u043B\u0438\u0437\u0430\u0446\u0438\u044F

city = \u0413\u043E\u0440\u043E\u0434

salary = \u0417\u0430\u0440\u043F\u043B\u0430\u0442\u0430

condition = \u0441 \u0437\u0430\u0440\u043F\u043B\u0430\u0442\u043E\u0439 >=

Да, последний файл выглядит немного странно, однако только с Юникодом надписи выводятся правильно

**Сервлет web-приложения**

Создадим сервлет, который выводит информацию о футболистах на русском и английском языках

import java.io.IOException;

import java.io.PrintWriter;

import java.util.Locale;

import java.util.ResourceBundle;

import jakarta.servlet.ServletException;

//import jakarta.servlet.annotation.WebServlet;

import jakarta.servlet.http.HttpServlet;

import jakarta.servlet.http.HttpServletRequest;

import jakarta.servlet.http.HttpServletResponse;

//@WebServlet("/TeamList")

/\*\*

\* Servlet implementation class Testing

\*/

public class Testing extends HttpServlet {

private static final long serialVersionUID = 1L;

private Object[][] team;

private String[] roles;

ResourceBundle res;

String salary, lang;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public Testing() {

super();

team = new Object[][] {

{"Билли Херрингтон", 0, "Махачкала", 15000},

{"Антон Чехов", 1, "Санкт-Петербург", 30000},

{"Илья Антонов", 2, "Екатеринбург", 25000},

{"Андрей Сачков", 3, "Вологда", 19000}

};

roles = new String[] {"Вратарь", "Нападающий", "Полузащитник", "Защитник"};

}

/\*\*

\* Processes requests for both HTTP <code>GET</code> and <code>POST</code>

methods.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

protected void processRequest(HttpServletRequest request, HttpServletResponse

response)

throws ServletException, IOException {

request.setCharacterEncoding("utf-8");

salary = request.getParameter("salary");

lang = request.getParameter("lang");

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

if (lang == null) lang = "ru";

if (!"en".equalsIgnoreCase(lang) && !"ru".equalsIgnoreCase(lang)) {

response.sendError(HttpServletResponse.SC\_NOT\_ACCEPTABLE,

"Параметр lang может принимать значения ru или en вместо \"" + lang + "\"");

return;

}

res = ResourceBundle.getBundle("team", ("ru".equals(lang))?

Locale.getDefault() : Locale.ENGLISH);

PrintWriter out = response.getWriter();

try {

// Создание HTML-страницы

out.println("<html>");

out.println("<head><title>Список моих футболистов</title></head>");

out.println("<body>");

out.println("<h1>" + res.getString("title") + ((salary == null)? "":

res.getString("condition") + salary + "$") + "</h1>");

out.println("<table border='1'>");

out.println("<tr><td><b>" + res.getString("name") + "</b></td>"

+ "<td><b>" + res.getString("spec") + "</b></td>"

+ "<td><b>" + res.getString("city") + "</b></td>"

+ "<td><b>" + res.getString("salary") + "</b></td></tr>");

for (Object[] temp : team)

if (salary == null || (int)temp[3] >= Integer.parseInt(salary))

out.println("<tr><td>" + temp[0] + "</td><td>" + roles[(int)temp[1]] + "</td><td>"

+ temp[2] + "</td><td>" + Integer.toString((int)temp[3]) + "</td></tr>");

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

String notlang = (lang.equals("ru"))? "en" : "ru";

out.println("<form method=\"post\">"

+ "<p><select name = \"lang\">"

+ "<option value = \"" + lang + "\">" + lang + "</option>"

+ "<option value = \""+notlang+"\">"+notlang + "</option>"

+ "</select>"

+ "<input type=\"Submit\" value=\"Submit\"/></p>"

+ "</form>");

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

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

} finally {

// Закрытие потока вывода

out.close();

}

}

/\*\*

\* Handles the HTTP

\* <code>GET</code> method.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Handles the HTTP

\* <code>POST</code> method.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

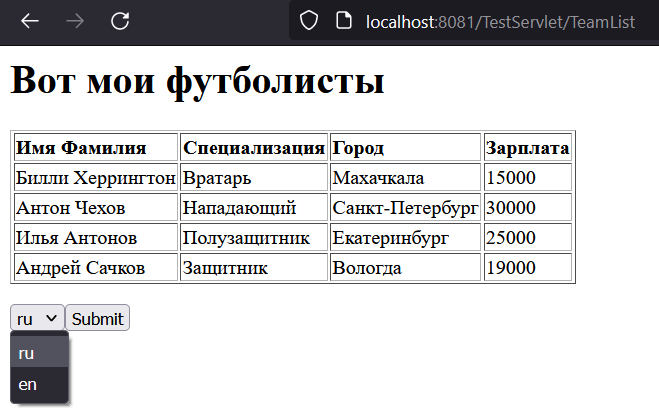
}

}

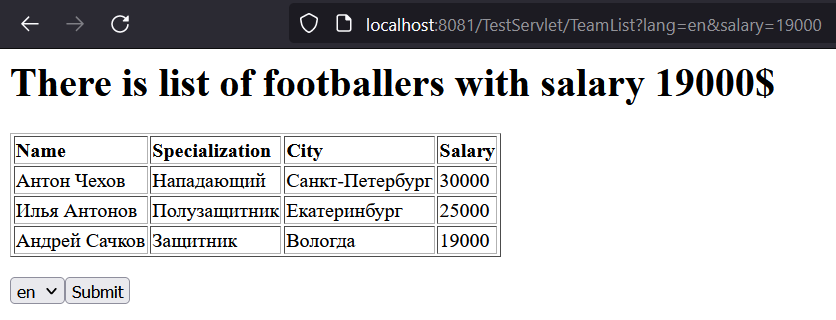
Web.xml файл остался таким же, как в предыдущей работе

**Запуск web-приложения и обращение к сервлету через браузер**

Поменять язык можно двумя способами: либо напрямую через адресную строку с указанием динамического параметра, либо через форму внизу (сразу менять ссылку при выборе опции списка я не могу, так что добавил кнопку Submit в помощь)



Здесь же поменял язык через адресную строку



**Текст документации**

Документация генерируется так же, как и в предыдущей работе

Результат:

//index.html

<!DOCTYPE HTML>

<!-- NewPage -->

<html lang="ru">

<head>

<!-- Generated by javadoc (11.0.10) on Sat Oct 09 19:43:46 MSK 2021 -->

<title>Generated Documentation (Untitled)</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

<script type="text/javascript">window.location.replace('Testing.html')</script>

<noscript>

<meta http-equiv="Refresh" content="0;Testing.html">

</noscript>

<link rel="canonical" href="Testing.html">

<link rel="stylesheet" type="text/css" href="stylesheet.css" title="Style">

</head>

<body>

<main role="main">

<noscript>

<p>JavaScript is disabled on your browser.</p>

</noscript>

<p><a href="Testing.html">Testing.html</a></p>

</main>

</body>

</html>

**Вывод**

В ходе выполнения лабораторной работы были изучены и освоены способы интернационализации Web-приложений с использованием файлов ресурсов. В частности, были выполнены следующие шаги:

1. **Создание файлов ресурсов**: Были созданы два файла .properties — один для русского языка (team\_ru.properties) и другой для английского (team\_en.properties). В этих файлах были определены строки для отображения текста на соответствующих языках.
2. **Модификация сервлета**: Сервлет был изменен для поддержки интернационализации. В зависимости от параметра lang, переданного в запросе, сервлет использует соответствующий файл ресурсов для отображения текста на русском или английском языке.
3. **Тестирование сервлета**: Сервлет был протестирован через браузер, где была возможность изменять язык через адресную строку или через форму на странице.
4. **Генерация документации**: Как и в предыдущей работе, была сгенерирована документация для сервлета с использованием инструмента JavaDoc.

Таким образом, в результате выполнения лабораторной работы было успешно реализовано Web-приложение с поддержкой интернационализации, что позволило отображать данные на разных языках.