ГУАП

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

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

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ |
| РАЗРАБОТКА ПРИЛОЖЕНИЯ С ИСПОЛЗОВАНИЕМ THYMELEAF |
| по курсу: ТЕХНОЛОГИИ РАЗРАБОТКИ СЕРВЕРНЫХ ИНФОРМАЦИОННЫХ СИСТЕМ |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ гр. № | 4136 |  |  |  | Н.С. Бобрович |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2023

**Задание (Вариант 3)**

Расписание занятий в институте

**Описание разрабатываемого продукта**

1 Реализуйте задачи из лабораторной работы 1 средствами Thymeleaf  
2 Обратите внимание на локализацию приложения. Русский язык должен  
поддерживаться.

**Текст программы**

**DBHelper**

package com.example.Lab3;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class DBHelper {  
 static public List<Lesson> *dbLesson* = new ArrayList<Lesson>();  
}

**Lab4Application**

package com.example.Lab3;  
  
import org.springframework.boot.SpringApplication;  
import org.springframework.boot.autoconfigure.SpringBootApplication;  
  
@SpringBootApplication  
public class Lab3Application {  
  
 public static void main(String[] args) {  
 DBHelper.*dbLesson*.add(new Lesson(0, "4132", "23-09", "Ivanov", "Physics", 1));  
 DBHelper.*dbLesson*.add(new Lesson(1, "4133", "23-08", "Petrov", "Math", 3));  
 DBHelper.*dbLesson*.add(new Lesson(2, "4134", "23-10", "Smirnov", "Programmin", 5));  
 SpringApplication.*run*(Lab3Application.class, args);  
 }  
  
}

**Lesson**

package com.example.Lab4;  
  
public class Lesson {  
 private int id;  
 private String num\_group; // номер группы  
 private String audience; // номер аудитории  
 private String teacher; // фамилия преподавателя  
 private String name; // название предмета  
 private int number; // порядковый номер пары  
 public Lesson(int id,String group,String audience,String teacher,String name,int number){  
 this.id = id;  
 this.num\_group = group;  
 this.audience = audience;  
 this.teacher = teacher;  
 this.name = name;  
 this.number = number;  
 }  
 public int getId(){  
 return id;  
 }  
 public void setId(int id){  
 this.id = id;  
 }  
 public String getGroup(){  
 return num\_group;  
 }  
 public void setGroup(String group){  
 this.num\_group = group;  
 }  
 public String getAudience(){  
 return audience;  
 }  
 public void setAudience(String audience){  
 this.audience = audience;  
 }  
 public String getTeacher(){  
 return teacher;  
 }  
 public void setTeacher(String teacher){  
 this.teacher = teacher;  
 }  
 public String getName(){  
 return name;  
 }  
 public void setName(String name){  
 this.name = name;  
 }  
  
 public int getNumber(){  
 return number;  
 }  
 public void setNumber(int number){  
 this.number = number;  
 }  
  
}

**Service**

package com.example.Lab4;  
  
import java.util.List;  
  
public interface Service{  
  
 void create(String group,String audience,String teacher,String name,int number);  
  
 List<Lesson> getAll();  
  
 List<Lesson> getByGroup(String group);  
  
 boolean delete(int id);  
}

**ServiceImpl**

package com.example.Lab4;  
  
import org.springframework.stereotype.Service;  
  
import java.util.ArrayList;  
import java.util.List;  
  
@Service  
public class ServiceImpl implements com.example.Lab4.Service {  
  
  
 @Override  
 public void create(String group, String audience, String teacher, String name, int number) {  
  
 DBHelper.dbLesson.add(new Lesson(DBHelper.dbLesson.get(DBHelper.dbLesson.size()-1 > 0 ? DBHelper.dbLesson.size()-1 : 0).getId()+1,  
 group,  
 audience,  
 teacher,  
 name,  
 number  
 ));  
  
 }  
  
 @Override  
 public List<Lesson> getAll() {  
 return DBHelper.dbLesson;  
 }  
  
 @Override  
 public List<Lesson> getByGroup(String group) {  
 List<Lesson> les = new ArrayList<>();  
 for(int i = 0; i<DBHelper.dbLesson.size(); ++i){  
 if(DBHelper.dbLesson.get(i).getGroup().contentEquals(group)){  
 les.add(DBHelper.dbLesson.get(i));  
 }  
 }  
 return les;  
 }  
  
 @Override  
 public boolean delete(int id) {  
 for(int i = 0; i<DBHelper.dbLesson.size(); ++i){  
 if(DBHelper.dbLesson.get(i).getId() == id){  
 DBHelper.dbLesson.remove(i);  
 return true;  
 }  
 }  
 return false;  
 }  
}

**SimpleController**

package com.example.Lab3;  
  
import org.apache.juli.logging.Log;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.http.HttpStatus;  
import org.springframework.http.ResponseEntity;  
import org.springframework.stereotype.Controller;  
import org.springframework.ui.Model;  
import org.springframework.validation.BindingResult;  
import org.springframework.validation.annotation.Validated;  
import org.springframework.web.bind.annotation.\*;  
import org.springframework.web.servlet.ModelAndView;  
  
import java.util.List;  
  
@Controller  
public class SimpleController {  
 private final Service service;  
 @Autowired  
 public SimpleController(Service service){  
 this.service = service;  
 }  
  
 @PostMapping("/save")  
 public String create(@ModelAttribute Lesson lesson, BindingResult bindingResult, Model model){  
 service.create(lesson.getGroup(), lesson.getAudience(), lesson.getTeacher(), lesson.getName(), lesson.getNumber());  
 return "redirect:/main";  
 }  
  
 @GetMapping("/lessons/{group}")  
 public ModelAndView getGroup(@PathVariable(name="group") String group){  
 final List<Lesson> lessons = service.getByGroup(group);  
 ModelAndView mav = new ModelAndView("main");  
 mav.addObject("lessons",service.getAll());  
 return mav;  
 }  
  
 @GetMapping("/delete/{id}")  
 public String delete(@PathVariable("id") int id){  
 final boolean deleted = service.delete(id);  
 return "redirect:/main";  
 }  
  
 @GetMapping("/main")  
 public ModelAndView getMainPage(){  
 Lesson lesson = new Lesson(DBHelper.*db*.size(),"","","","",1);  
 ModelAndView mav = new ModelAndView("main");  
 mav.addObject("lessons",service.getAll());  
 mav.addObject("lesson",lesson);  
 return mav;  
 }  
}

**index.html**

<!DOCTYPE html>  
<html xmlns:th="http://www.w3.org/1999/xhtml">  
 <head>  
 <meta content="text/html;charset=utf-8">  
 </head>  
 <body>  
 <h2 th:text="#{schedule.schedule}">Расписание занятий студентов</h2>  
 <a href="/main" th:text="#{view.view}">View schedule</a><br>  
 </body>  
</html>

**main.html**

<!DOCTYPE html>  
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:th="http://www.thymeleaf.org"  
 xmlns:sec="http://www.thymeleaf.org/thymeleaf-extras-springsecurity3" xmlns:form="http://www.w3.org/1999/xhtml">  
 <head>  
 <meta http-equiv="Conternt-Type" content="text/html;charset=utf-8">  
 </head>  
 <body>  
 <h2 th:text="#{schedule.schedule}">Расписание занятий студентов</h2>  
 <div>  
 <table id="lessonsTable">  
 <thead>  
 <tr>  
 <th th:text="#{lesson.number}">Number</th>  
 <th th:text="#{lesson.name}">Subject</th>  
 <th th:text="#{lesson.audience}">Audience</th>  
 <th th:text="#{lesson.teacher}">Teacher</th>  
 <th th:text="#{lesson.group}">Group</th>  
 <th></th>  
 </tr>  
 </thead>  
 <tbody>  
 <tr th:each="lesson : ${lessons}">  
 <td th:text="${lesson.number}">lesson.number</td>  
 <td th:text="${lesson.name}">lesson.name</td>  
 <td th:text="${lesson.audience}">lesson.audience</td>  
 <td th:text="${lesson.teacher}">lesson.teacher</td>  
 <td th:text="${lesson.group}">lesson.group</td>  
 <td><a href="#" th:href="@{'delete/'+${lesson.id}}" th:text="#{delete.delete}">Delete</a></td>  
 </tr>  
 </tbody>  
 </table>  
 </div><br>  
 <div>  
 <form method="POST" th:action="@{'/save'}" th:object="${lesson}" >  
 <label th:text="#{lesson.number}"></label><input id="number" th:field="\*{number}" type="text"><br>  
 <label th:text="#{lesson.name}"></label><input id="name" th:field="\*{name}" type="text"><br>  
 <label th:text="#{lesson.audience}"></label><input id="audience" th:field="\*{audience}" type="text"><br>  
 <label th:text="#{lesson.teacher}"></label><input id="teacher" th:field="\*{teacher}" type="text"><br>  
 <label th:text="#{lesson.group}"></label><input id="num\_group" th:field="\*{group}" type="text"><br>  
 <input type="submit">  
 </form>  
 </div>  
 </body>  
</html>

**messages.properties**

schedule.schedule=Schedule  
lesson.number=Number  
lesson.name=Subject  
lesson.audience=Audience  
lesson.teacher=Teacher  
lesson.group=Group  
view.view=View schedule  
delete.delete=Delete  
send.send=Send

**messages\_ru\_RU.preperties**

schedule.schedule=Расписание  
lesson.number=Номер пары  
lesson.name=Предмет  
lesson.audience=Аудитория  
lesson.teacher=Преподаватель  
lesson.group=Группа  
view.view=Посмотреть расписание  
delete.delete=Удалить  
send.send=Отправить