#### Министерство науки и высшего образования Российской Федерации

#### Федеральное государственное бюджетное образовательное учреждение

#### высшего образования

#### «Владимирский государственный университет

#### имени Александра Григорьевича и Николая Григорьевича Столетовых»

**(ВлГУ)**

**Кафедра информационных систем и программной инженерии**

Отчет

по лабораторной работе №

по дисциплине «Распределенные программные системы»

на тему «JSP: Spring Tag Library, Validation, Formatting.»

Выполнил:

студент группы ПРИ-121

Зубов Н.А.

Проверила:

Доцент кафедры ИСПИ

Проскурина Г.В.

Владимир 2023

Цель работы

Познакомиться библиотекой тегов Spring для JSP страниц, получить навыки валидации вводимых данных и форматирования выводимых данных.

Задание

Необходимо доработать приложение из лабораторной работы 6 добавив сложную форму на основе Spring Form.Продемонстрировать владение технологией Spring Validation и навыками форматирования данных на JSP.

Ход работы

Листинг Модели:

public class Student {

private Long id;

private long groupId;

@Pattern(regexp = "[a-zA-Z]+(\\s[a-zA-Z]+)\*", message = "Name can contain only latin letters and spaces")

private String name;

@Email(message = "Email not be valid")

private String email;

@Min(value = 16, message = "Age shouldn't be less than 16")

@Max(value = 100, message = "Age shouln't be greater than 100")

private int age;

@PositiveOrZero(message = "countOfOnlineToday shouldn't be negative")

private int countOfOnlineToday;

@NotNull

@DateTimeFormat(pattern = "yyyy-MM-dd")

private Date createdAt;

private Date lastTimeOnline;

Листинг формы:

<%@ page contentType="text/html;charset=UTF-8" language="java" %>

<%@ page isELIgnored="false" %>

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>

<%@ taglib prefix="sf" uri="http://www.springframework.org/tags/form" %>

<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>

<h1>форма студента для <c:out value="${isCreate ? 'Создание' : 'Редактирование'}"/></h1>

<sf:form method="post" action="Form" modelAttribute="student">

<c:if test="${!isCreate}">

<sf:input type="hidden" path="id" />

<label for="countOfOnlineToday">Count of Online Today:</label>

<sf:input path="countOfOnlineToday"/><br/>

</c:if>

<label for="name">Name:</label>

<sf:input path="name"/><sf:errors path="name" cssClass="error"/><br/>

<label for="email">Email:</label>

<sf:input path="email"/><sf:errors path="email" cssClass="error"/><br/>

<label for="age">Age:</label>

<sf:input path="age"/><sf:errors path="age" cssClass="error"/><br/>

<label for="createdAt">Age:</label>

<sf:input type="date" path="createdAt"/><sf:errors path="createdAt" cssClass="error"/><br/>

<label for="groupId">Group:</label>

<sf:select path="groupId">

<sf:options items="${groupList}" itemLabel="name" itemValue="id"/>

</sf:select><br/>

<input type="submit" value="Save"/>

</sf:form>

Методы:

@GetMapping("/Student/Form")

public String StudentEdit(@RequestParam(required = false, name="id") Integer id,Model model){

if(id == null){

model.addAttribute("isCreate", true);

model.addAttribute("student", new Student());

}else{

model.addAttribute("isCreate", false);

Student student = studentDAO.getById(id);

model.addAttribute("student", student);

}

model.addAttribute("groupList", groupDAO.getAll());

return "StudentForm";

}

@PostMapping("/Student/Form")

public String updateStudent(@Valid @ModelAttribute("student") Student student, BindingResult result,Model model) {

if(result.hasErrors()){

model.addAttribute("groupList", groupDAO.getAll());

model.addAttribute("isCreate", Objects.isNull(student.getId()));

return "StudentForm";

}

if (Objects.isNull(student.getId())) {

student.setCountOfOnlineToday(0);

student.setLastTimeOnline(new Date());

studentDAO.create(student);

} else {

Student timeSt = studentDAO.getById(student.getId());

student.setLastTimeOnline(timeSt.getLastTimeOnline());

studentDAO.update(student);

}

return "redirect:/Student/List";

}

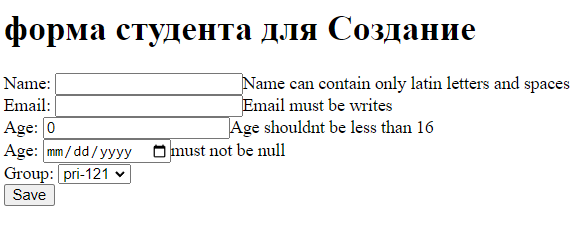


Рисунок 1 — Валидация создания

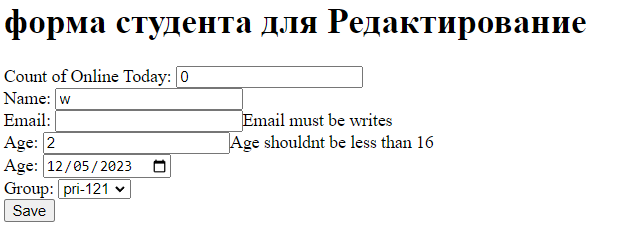


Рисунок 2 — Валидация редактирования

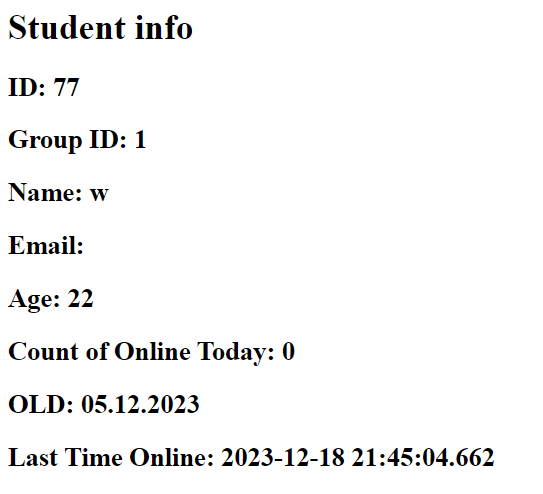


Рисунок 3 — Форматированный вывод “OLD”

Вывод

Познакомился библиотекой тегов Spring для JSP страниц, получил навыки валидации вводимых данных и форматирования выводимых данных.