

Министерство науки и высшего образования Российской Федерации
Муромский институт (филиал)
Федерального государственного бюджетного образовательного учреждения высшего
образования
«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»

Факультет _____ ИТР _____

Кафедра _____ ПИН _____

ЛАБОРАТОРНАЯ РАБОТА №3

По _____ Разработка корпоративных приложений _____

Тема _____ Репозиторий в Spring Data JPA _____

Руководитель

Кульков Я.Ю.

(фамилия, инициалы)

(подпись)

(дата)

Студент _____ ПИН - 121 _____
(группа)

Ермилов М.В.

(фамилия, инициалы)

(подпись)

(дата)

Муром 2024

Лабораторная работа №3

Тема: репозиторий в Spring Data JPA.

Цели и задачи: изучить способ использования репозитория в Spring Data JPA.

Ход работы: задание: реализовать приложение с функциями создания, редактирования, удаления и отображения в соответствии с вариантом. Реализовать методы получения списка всех сущностей, одной сущности по id, удаление по id, добавление и изменение. При поиске сущности, изменении или удалении проверять её существование в БД. При отображении страниц использовать стили оформления. Задание для варианта 7: сущность – Военнослужащий; атрибуты : фамилия; имя; отчество; национальность; дата рождения (год, месяц число); должность; звание.

Листинг кода 1 – MainController.java

```
package ru.kafpin.firstttt.controllers;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.PostMapping;
import ru.kafpin.firstttt.entities.Soldier;
import ru.kafpin.firstttt.services.SoldierService;

import java.util.List;

@Controller
public class MainController {

    private SoldierService soldierService;

    @Autowired
    public void setSoldierService(SoldierService soldierService){
        this.soldierService = soldierService;
    }

    @GetMapping("/")
    public String RootPage(Model model){
        return "redirect:/soldiers";
    }

    @GetMapping("/kazarma")
    public String KazarmaPage(Model model){
        return "redirect:/soldiers";
    }
}
```

					МИВлГУ 09.03.04 - 0.007							
Изм.	Лист	№ докум.	Подпись	Дата								
Разраб.		Ермилов М.В.			Репозиторий в Spring Data JPA			Лит.		Лист	Листов	
Провер.		Кульков Я.Ю.								2	7	
Реценз.								МИ ВлГУ ПИН-121				
Н. Контр.												
Утверд.												

```

@GetMapping("/soldiers")
public String soldiersPage(Model model){
    List<Soldier> allSoldiers = soldierService.getAllSoldiers();
    model.addAttribute("soldiers", allSoldiers);
    return "soldiers";
}

@GetMapping("/soldiers/details/{id}")
public String detailsPage(Model model, @PathVariable("id") Long id) {
    Soldier selectedSoldier = soldierService.getSoldierById(id);
    model.addAttribute("selectedSoldier", selectedSoldier);
    return "details";
}

@GetMapping("/soldiers/delete/{id}")
public String deleteSoldierById(@PathVariable("id") Long id){
    soldierService.deleteSoldierById(id);
    return "redirect:/soldiers";
}

@GetMapping("/soldiers/add")
public String addSoldier(Model model){
    model.addAttribute("soldier", new Soldier());
    return "addedsoldier";
}

@PostMapping("/soldiers/add")
public String mainForm(@ModelAttribute Soldier soldier, Model
    model){
    model.addAttribute("soldier", soldier);
    soldierService.addEditSoldier(soldier);
    return "redirect:/soldiers";
}

@GetMapping("/soldiers/update/{id}")
public String editSoldier(Model model, @PathVariable("id") Long id) {
    model.addAttribute("soldier", soldierService.getSoldierById(id));
    return "addedsoldier";
}

@PostMapping("/soldiers/update")
public String editSoldier(@ModelAttribute Soldier soldier, Model model) {
    soldierService.addEditSoldier(soldier);
    return "redirect:/soldiers";
}
}

```

Листинг кода 2 – Soldier.java

```

package ru.kafpin.firstttt.entities;

import jakarta.persistence.*;
import lombok.Data;
import lombok.NoArgsConstructor;
import java.time.LocalDate;

@Entity
@Table(name = "soldiers")
@Data
@NoArgsConstructor
public class Soldier {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "id")
    private Long id;
    @Column(name = "surname", nullable = false)
    private String surname;
    @Column(name = "name", nullable = false)
    private String name;
    @Column(name = "patronym", nullable = true)
    private String patronym;
    @Column(name = "nation", nullable = false)
    private String nation;
    @Column(name = "birthday", nullable = false)
    private LocalDate birthday;
    @Column(name = "post", nullable = false)
    private String post;
    @Column(name = "rank", nullable = false)

```

					МИВлГУ 09.03.04 – 0.007	Лист
						3
Изм.	Лист	№ докум.	Подпись	Дата		

```

        private String rank;

        public Soldier(Long id, String surname, String name, String patronym, String nation, LocalDate
birthday, String post, String rank) {
            this.id = id;
            this.surname = surname;
            this.name = name;
            this.patronym = patronym;
            this.nation = nation;
            this.birthday = birthday;
            this.post = post;
            this.rank = rank;
        }
    }
}

```

Листинг кода 3 – SoldierRepository.java

```

package ru.kafpin.firstttt.repositories;

import org.springframework.data.repository.CrudRepository;
import org.springframework.stereotype.Component;
import ru.kafpin.firstttt.entities.Soldier;

@Component
public interface SoldierRepository extends CrudRepository<Soldier, Long> {
}

```

Листинг кода 4 – SoldierService.java

```

package ru.kafpin.firstttt.services;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import ru.kafpin.firstttt.entities.Soldier;
import ru.kafpin.firstttt.repositories.SoldierRepository;

import java.util.List;

@Service
public class SoldierService {
    @Autowired
    private SoldierRepository soldierRepository;

    @Autowired
    public void setSoldierRepository(SoldierRepository soldierRepository) {
        this.soldierRepository = soldierRepository;
    }
    public List<Soldier> getAllSoldiers() {
        return (List<Soldier>) soldierRepository.findAll();
    }
    public Soldier getSoldierById(Long id) {
        return soldierRepository.findById(id).get();
    }
    public void deleteSoldierById(Long id) {
        if (soldierRepository.existsById(id))
            soldierRepository.deleteById(id);
    }
    public void addEditSoldier(Soldier soldier) {
        soldierRepository.save(soldier);
    }
}

```

Листинг кода 5 – addeditsoldier.html

```

<!DOCTYPE HTML>
<html xmlns:th="https://www.thymeleaf.org">

<head>
    <title>Добавление военнослужащего</title>
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css">
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js"></script>
</head>

<body>

```

					МИВЛГУ 09.03.04 – 0.007	Лист
Изм.	Лист	№ докум.	Подпись	Дата		4

```

<div class="container">
  <h1>Информация о военнослужащем</h1>
  <a class="btn" href="/kazarma">К списку военнослужащих</a>
<!--
  <a class="btn btn-secondary" th:href="@{/soldiers/update/' + $
{soldier.id}}">Редактировать</a>-->
  <form class="customfrm" action="#" th:action="@{/soldiers/add}" th:object="${soldier}"
method="post">
    <div class="form-group">
      <input type="hidden" class="form-control" th:field="*{id}" />
      <p>Фамилия: <input type="text" class="form-control" placeholder="Введите фамилию"
th:field="*{surname}" required/></p>
      <p>Имя: <input type="text" class="form-control" placeholder="Введите имя"
th:field="*{name}" required/></p>
      <p>Отчество: <input type="text" class="form-control" placeholder="Введите отчество"
th:field="*{patronym}" /></p>
      <p>Национальность: <input type="text" class="form-control" placeholder="Введите
национальность" th:field="*{nation}" required/></p>
      <p>Дата рождения: <input type="date" class="form-control" content=""
th:field="*{birthday}" required/></p>
      <p>Должность: <input type="text" class="form-control" placeholder="Введите
должность" th:field="*{post}" required/></p>
      <p>Звание: <input type="text" class="form-control" placeholder="Введите звание"
th:field="*{rank}" required/></p>
    </div>
    <p class="singleRow">
      <input type="submit" class="btn btn-success" value="Добавить" />
      <input type="reset" class="btn btn-secondary" value="Очистить поля ввода" />
    </p>
  </form>
</div>
</body>
</html>

```

Листинг кода 6 – details.html

```

<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.thymeleaf.org">

<head>
  <meta charset="UTF-8">
  <title>Информация о военнослужащем</title>

  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js"></script>
</head>

<body>
  <div class="container">
    <h1>Военнослужащий</h1>
    <ul class="list-group">
      <li class="list-group-item">ID: <span th:text="${selectedSoldier.id}"></span></li>
      <li class="list-group-item">Фамилия: <span
th:text="${selectedSoldier.surname}"></span></li>
      <li class="list-group-item">Имя: <span th:text="${selectedSoldier.name}"></span></li>
      <li class="list-group-item">Отчество: <span
th:text="${selectedSoldier.patronym}"></span></li>
      <li class="list-group-item">Национальность: <span th:text="$
{selectedSoldier.nation}"></span></li>
      <li class="list-group-item">Дата рождения: <span th:text="$
{selectedSoldier.birthday}"></span></li>
      <li class="list-group-item">Должность: <span
th:text="${selectedSoldier.post}"></span></li>
      <li class="list-group-item">Звание: <span th:text="${selectedSoldier.rank}"></span></li>
    </ul>

    <a th:href="@{/kazarma}" class="btn btn-primary mt-3">Назад к списку</a>
  </div>
</body>
</html>

```

Листинг кода 7 – soldiers.html

```

<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.thymeleaf.org">

```

					МИВЛГУ 09.03.04 – 0.007	Лист
Изм.	Лист	№ докум.	Подпись	Дата		5

```

<head>
  <meta charset="UTF-8">
  <title>Казарма</title>
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js"></script>
</head>

<body>
  <div class="container">
    <h1>Казарма</h1>
    <span th:if="${soldiers.size() == 0}">Список военнослужащих пустой</span>
    <table th:if="${soldiers.size() > 0}" class="table table-hover">
      <thead class="thead-dark">
        <tr>
          <td>ID</td>
          <td>Фамилия</td>
          <td>Имя</td>
          <td>Отчество</td>
          <td>Национальность</td>
          <td>Дата рождения</td>
          <td>Должность</td>
          <td>Звание</td>
          <td></td>
          <td></td>
          <td></td>
        </tr>
      </thead>
      <tbody>
        <tr th:each="soldier : ${soldiers}">
          <td th:text="${soldier.id}" />
          <td th:text="${soldier.surname}" />
          <td th:text="${soldier.name}" />
          <td th:text="${soldier.patronym}" />
          <td th:text="${soldier.nation}" />
          <td th:text="${soldier.birthday}" />
          <td th:text="${soldier.post}" />
          <td th:text="${soldier.rank}" />
          <td><a class="btn btn-secondary" th:href="@{'/soldiers/details/' + ${soldier.id}}">Подробнее</a></td>
          <td><a class="btn btn-secondary" th:href="@{'/soldiers/update/' + ${soldier.id}}">Редактировать</a></td>
          <td><a class="btn btn-danger" th:href="@{'/soldiers/delete/' + ${soldier.id}}">Удалить</a></td>
        </tr>
      </tbody>
    </table>
    <a class="btn" th:href="@{'/soldiers/add'}">Добавить военнослужащего</a>
  </div>
</body>
</html>

```

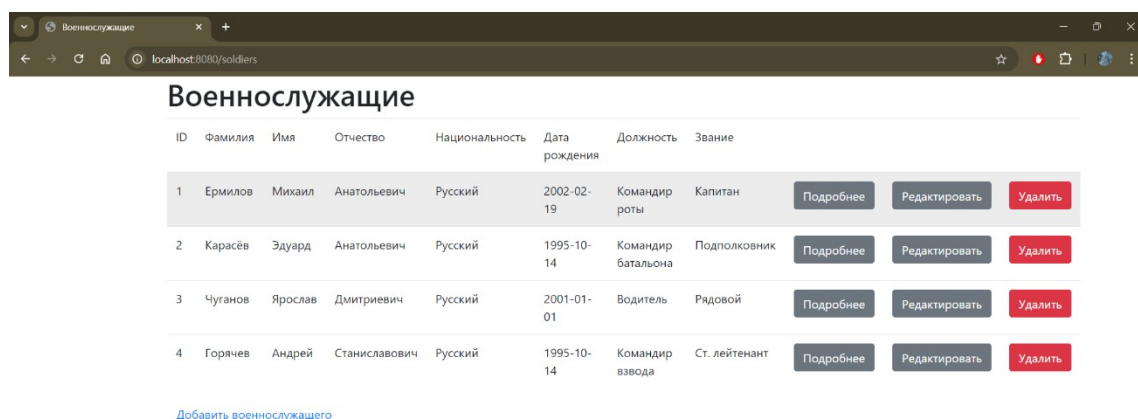


Рисунок 1 – Главная страница

Военнослужащий

ID: 1
Фамилия: Ермилов
Имя: Михаил
Отчество: Анатольевич
Национальность: Русский
Дата рождения: 2002-02-19
Должность: Командир роты
Звание: Капитан

[Назад к списку](#)

Рисунок 2 – Страница подробной информации

Информация о военнослужащем

[К списку военнослужащих](#)

Фамилия: Ермилов
Имя: Введите имя
Отчество: Анатольевич
Национальность: Русский
Дата рождения: 19.02.2002
Должность: Командир роты
Звание: Капитан

[Добавить](#) [Очистить поля ввода](#)

Рисунок 3 – Страница добавления/редактирования

Вывод: в ходе работы изучили способ использования репозитория в Spring Data JPA.