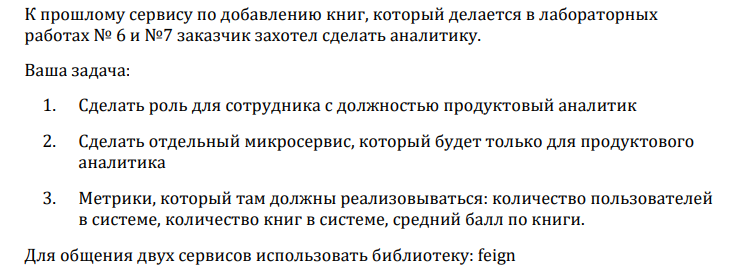
**Поволжский государственный университет телекоммуникаций и информатики**

**Отчёт по дисциплине «Разработка серверной части web-приложений»**

**Лабораторная работа №8.**

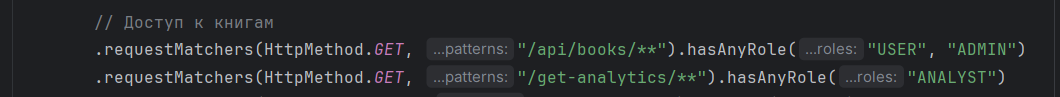
**Выполнил студент группы ИВТ-23**

**Ермишин Максим**

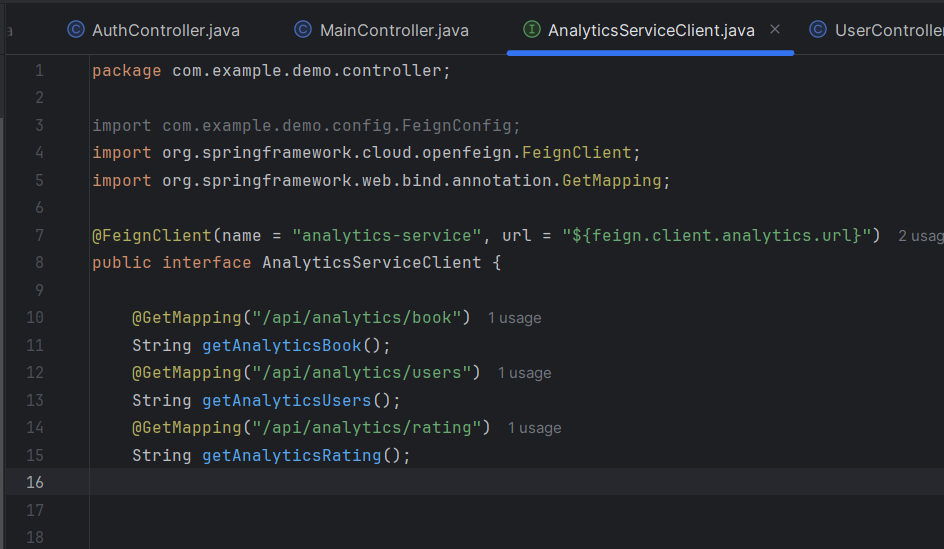


1. Основное приложение

1.1 Создана роль новая роль сотрудника:



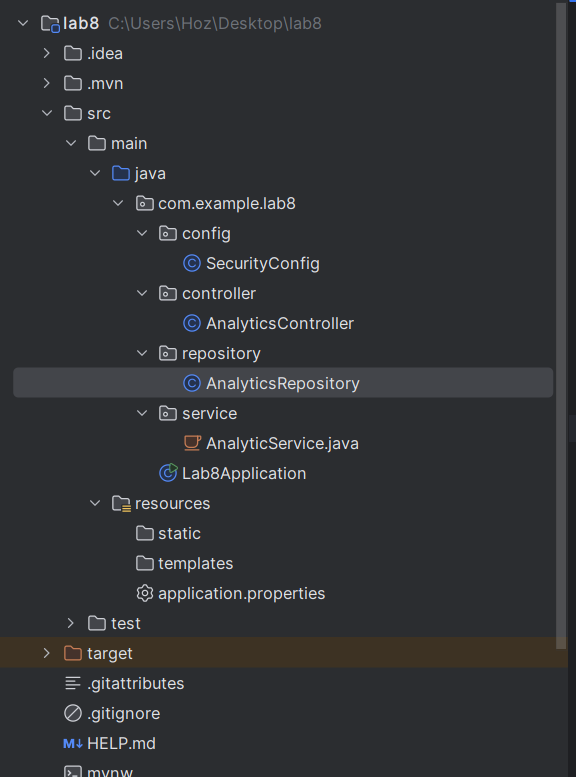
1.2 Создан Feign client



1.3 Создан новый контроллер для аналитики



1. Создан отдельный микросервис



3 Создан Репозиторий для получения соответствующих данных из базы

package com.example.lab8.repository;  
  
import org.springframework.jdbc.core.JdbcTemplate;  
import org.springframework.stereotype.Repository;  
  
@Repository  
public class AnalyticsRepository {  
 private final JdbcTemplate jdbcTemplate;  
  
 public AnalyticsRepository(JdbcTemplate jdbcTemplate) {  
 this.jdbcTemplate = jdbcTemplate;  
 }  
  
 public int getUserCount() {  
 return jdbcTemplate.queryForObject(  
 "SELECT COUNT(\*) FROM users",  
 Integer.class  
 );  
 }  
  
 public int getBookCount() {  
 return jdbcTemplate.queryForObject(  
 "SELECT COUNT(\*) FROM book",  
 Integer.class  
 );  
 }  
  
 public double getAvgBookRating() {  
 return jdbcTemplate.queryForObject(  
 "SELECT AVG(rating) FROM books",  
 Double.class  
 );  
 }  
}

3.1 Создан Контроллер

package com.example.lab8.controller;  
  
import com.example.lab8.repository.AnalyticsRepository;  
import org.springframework.web.bind.annotation.GetMapping;  
import org.springframework.web.bind.annotation.RequestMapping;  
import org.springframework.web.bind.annotation.RestController;  
  
import java.util.List;  
  
@RestController  
@RequestMapping(path = "/api/analytics")  
public class AnalyticsController {  
  
 private final AnalyticsRepository analyticsRepository;  
  
  
 public AnalyticsController(AnalyticsRepository analyticsRepository) {  
 this.analyticsRepository = analyticsRepository;  
 }  
  
 @GetMapping  
 @RequestMapping(path = "/book")  
 public int GetAnalyticsBook(){  
  
 return analyticsRepository.getBookCount();  
  
 }  
 @GetMapping  
 @RequestMapping(path = "/users")  
 public int GetAnalyticsUsers(){  
  
 return analyticsRepository.getUserCount();  
  
 }  
 @GetMapping  
 @RequestMapping(path = "/rating")  
 public double GetAnalyticsRating(){  
  
 return analyticsRepository.getAvgBookRating();  
  
 }  
  
  
}

