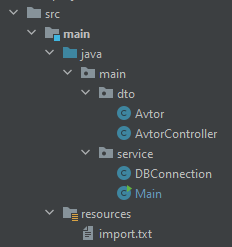
Структура проекта:



Класс DBConnection

Класс Avtor

Класс AvtorController

Изображение выглядит как текст, апельсин

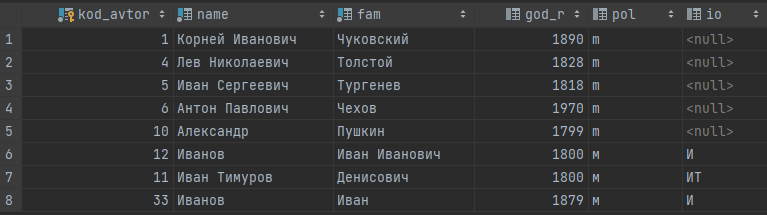
Автоматически созданное описаниеКласс Main – внутри создается экземпляр класса DBConnection, обеспечивающий подключение к БД, и экземпляр класса Avtor, создающий нового автора. Также вызываются функции класса AvtorController для манипулирования данными в БД.

Изображение выглядит как текст

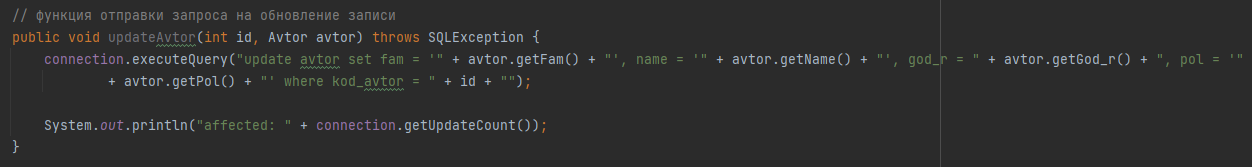
Автоматически созданное описаниеИзображение выглядит как текст

Автоматически созданное описаниеИзображение выглядит как текст, снимок экрана, монитор, экран

Автоматически созданное описание







Изображение выглядит как текст

Автоматически созданное описание



Изображение выглядит как текст

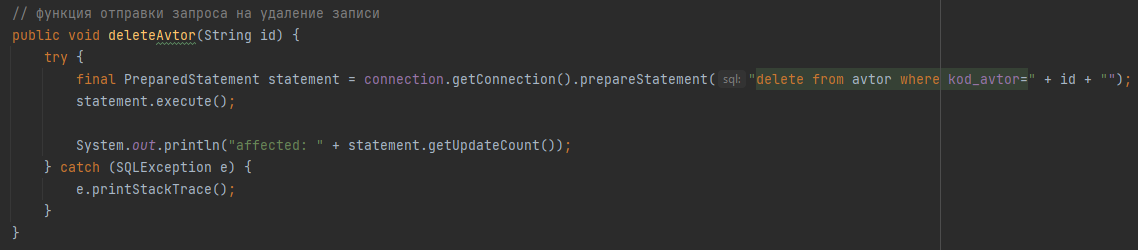
Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

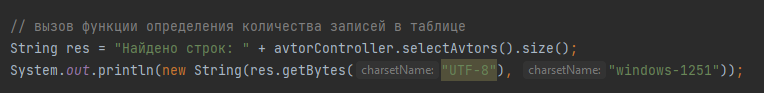


Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст, электроника, экран, черный

Автоматически созданное описание

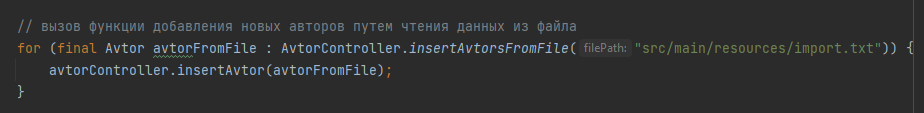


Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание



Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст, электроника, снимок экрана

Автоматически созданное описание

Изображение выглядит как текст, в помещении, черный, снимок экрана

Автоматически созданное описание

ПРИЛОЖЕНИЕ

1. Код класса DBConnection
2. Код класса Avtor
3. Код класса AvtorController
4. Код класса Main

package main.service;  
import main.dto.Avtor;  
import main.dto.AvtorController;  
import java.io.UnsupportedEncodingException;  
import java.sql.SQLException;  
  
public class Main {  
  
 public static void main(String[] args) throws SQLException, UnsupportedEncodingException {  
 // создание подключения к БД  
 DBConnection dbConn = new DBConnection();  
 // создание нового автора  
 Avtor avtor = new Avtor("Иван", "Иванов", 1879, 'м');  
  
 // добавление подключения к классу, содержащему управляющие функции  
 AvtorController avtorController = new AvtorController(dbConn);  
  
 // вызов функции добавления нового автора  
 avtorController.insertAvtor(avtor);  
 // вызов функции обновления автора по id  
 avtorController.updateAvtor(33, new Avtor("Романова", "Елизавета", 1900, 'ж'));  
 // вызов функции вывода всей информации об авторе по id  
 avtorController.absoluteSelect(33);  
 // вызов функции удаления автора по id  
 avtorController.deleteAvtor("35");  
  
  
 // вызов функции добавления новых авторов путем чтения данных из файла  
 for (final Avtor avtorFromFile : AvtorController.*insertAvtorsFromFile*("src/main/resources/import.txt")) {  
 avtorController.insertAvtor(avtorFromFile);  
 }  
  
 // вызов функции определения количества записей в таблице  
 String res = "Найдено строк: " + avtorController.selectAvtors().size();  
 System.*out*.println(new String(res.getBytes("UTF-8"), "windows-1251"));  
 }  
}