МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

(ФГБОУ ВО «ВятГУ»)

Институт математики и информационных систем

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Отчёт

Лабораторная работа №2 по дисциплине

«Объектно-ориентированное программирование»

Выполнил студент группы ИВТб-2305-05-00 \_\_\_\_\_\_\_\_ /Огородникова Д.А.

Проверил зав. кафедры ЭВМ \_\_\_\_\_\_\_\_ /Долженкова М.Л.

Киров 2025

**Цель работы:** создание приложения с системой классов, закрепление их свойств.

**Задание:** создать приложение, где есть родительский класс Фильмы и подклассы Развлекательные и Научные. У каждого класса должно быть 2 данных, 2 собственных, 2 перекрытых.

**Создание структуры классов**

Создание базового класса Film.

1. Инкапсуляция: свойства Scenario и Montage объявлены как protected, что ограничивает доступ к ним только внутри класса и его подклассов; для взаимодействия с внешним кодом используются публичные методы (например, конструктор и виртуальные методы).
2. Виртуальные методы: методы Financing(), Shooting(), Sounds(), EmotionalImpact() помечены как virtual, что позволяет подклассам переопределять их поведение.



Рисунок 1 — Виртуальный метод

1. Связь с базой данных: для работы с PostgreSQL используется библиотека Npgsql; данные сохраняются в таблицы films\_schema.films (для развлекательных фильмов) и films\_schema.films2 (для научных).

Создание подклассов.

1. Развлекательные фильмы (EntertainmentFilm).

Свои данные: Type, AgeRestriction.

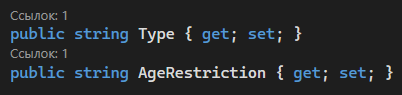


Рисунок 2 — Данные подкласса EntertainmentFilm

Свои собственные методы: Entertain(), Relax().

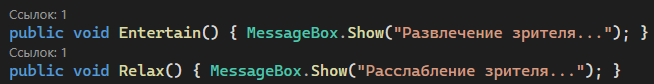


Рисунок 3 — Собственные методы подкласса EntertainmentFilm

Перекрывающие методы: Sounds(), EmotionalImpact().

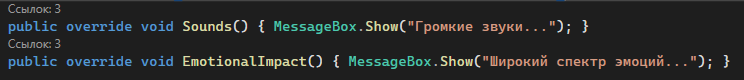


Рисунок 4 — Перекрывающие методы подкласса EntertainmentFilm

1. Научные фильмы (ScientificFilm).

Свои данные: Field, Audience.

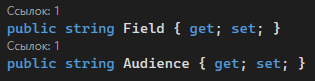


Рисунок 5 — Данные подкласса ScientificFilm

Свои собственные методы: Educate(), Motivate().

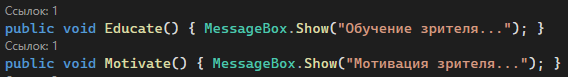


Рисунок 6 — Собственные методы подкласса ScientificFilm

Перекрывающие методы: Sounds(), EmotionalImpact().

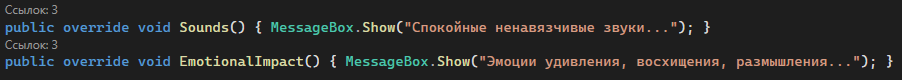


Рисунок 7 — Перекрывающие методы подкласса ScientificFilm

Если в классе есть метод, который объявлен виртуальным, то в нем создается ссылка на виртуальную таблицу именно этого класса, а также создаются ссылки на таблицы в подклассах. Таким образом, когда программа считывает описание метода, она подставляет реализацию именно того класса, который был нужен. Так достигается полиморфизм.

В итоге получилась следующая структура.

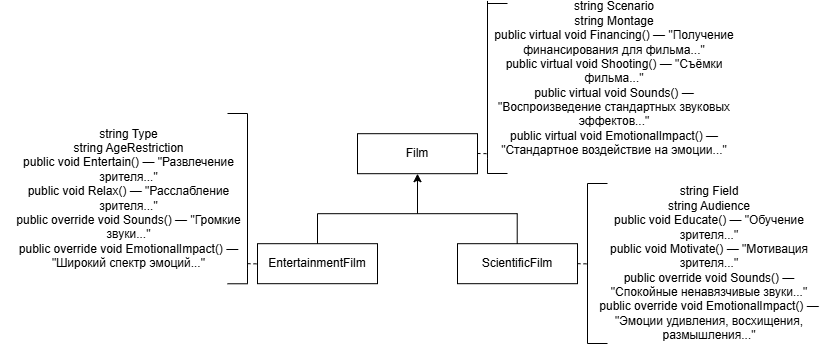


Рисунок 8 — Структура классов

Работа с базой данных.

Для взаимодействия с базой данных PostrgeSQL используется библиотека Npgsql.

Операции с базой данных.

1. Подключение к базе данных.

Перед выполнением любых операций с БД необходимо установить соединение. Это делается с помощью строки подключения и конструкции using, которая автоматически закрывает соединение после выполнения операций.

1. Добавление данных в базу данных.

Чтобы добавить новый фильм в базу данных создаётся объект класса EntertainmentFilm или ScientificFilm; формируется SQL-запрос с параметрами для безопасной вставки; выполняется команда ExecuteNonQuery().

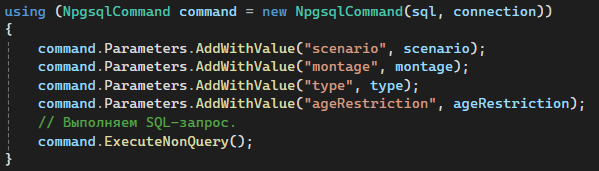


Рисунок 9 — Пример добавления в БД для развлекательного фильма

1. Обновление данных в базе данных.

Для изменения существующей записи: необходимо получить объект из БД (например, по ID); обновить его свойства; выполнить SQL-запрос UPDATE.

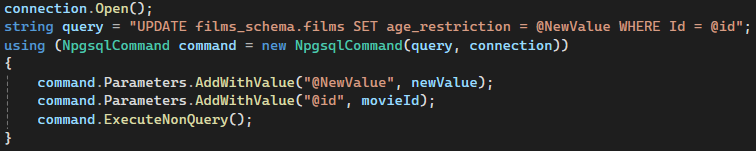


Рисунок 10 — Пример обновление данных в БД для развлекательного фильма

1. Выбор данных из базы данных.

Для получения данных из БД используется метод ExecuteReader(), который возвращает результат запроса.

1. Удаление данных из базы данных.

Чтобы удалить запись, используется запрос DELETE.

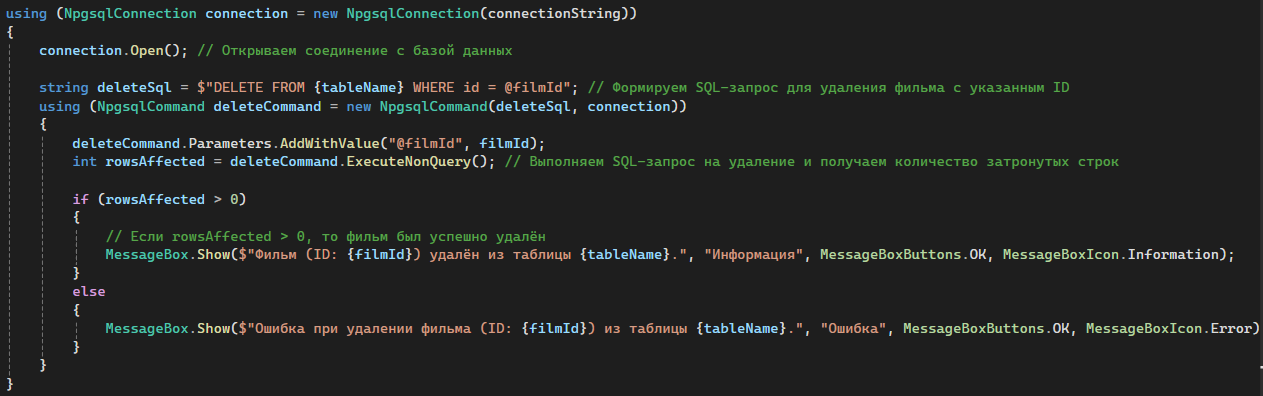


Рисунок 11 — Удаление записи из БД

**Вывод:** в ходе выполнения данной лабораторной работы была разработана объектно-ориентированная структура классов, включающая родительский класс Фильмы и два подкласса — Развлекательные и Научные. Были закреплены ключевые принципы ООП, такие как инкапсуляция, наследование и полиморфизм. Каждый класс содержит собственные данные, методы, а также переопределённые виртуальные функции, демонстрирующие различное поведение подклассов.

Дополнительно была реализована работа с базой данных с использованием библиотеки Npgsql. Созданные классы отображаются на таблицы БД, реализованы операции добавления, изменения и выборки данных.