|  |  |  |
| --- | --- | --- |
| Лабораторная работа №6  Взаимодействие с БД из приложения | Ф.И.О. | Авдеев И. А. |
| Группа | ИВТ-263 |
| Преподаватель | Соколов А.А. |
| Дата сдачи |  |

Задание на лабораторную работу

1. Выбрать язык программирования (**любой**) для реализации лабораторной работы и в дальнейшем курсовой работы (в протоколе указать, какой)

2. Изучить технологии (библиотеки) для работы с выбранной СУБД и **реализовать простейшее приложение** для подключения к собственной БД в СУБД на выбранном языке программирования и с помощью выбранных библиотек для подключения (в протоколе указать, каких)

3. В качестве интерфейса взаимодействия с пользователем на данной лабораторной работе достаточно реализовать **консольное приложение или простую веб-страницу** с элементарной вёрсткой, однако можно реализовать настольное (оконное) приложение, полноценное веб-приложение с приличной вёрсткой, мобильное приложение и т.п.

4. **Исходный код программы коммитить**в репозиторий (если есть), приложить ссылку в протокол

5. Использование ООП и паттернов проектирования желательно, но не обязательно. Как минимум необходимо **разделить уровень работы с интерфейсом и уровень взаимодействия с БД.**

6. Реализовать **CRUD (получение, добавление, редактирование и удаление данных)**для**2-х таблиц** вашей базы данных, одна из которых должна быть связана с первой внешним ключом.

**Присоединённые данные должны отображаться в виде атрибутов с данными из другой таблицы, а не ID внешних ключей.** Т.е. в запросах использовать **JOIN**. Например, если выводите игру, в которой есть жанр, издатель, разработчик, нужно выводить не ID жанра, ID издателя и ID разработчика, а наименование жанра, наименования издателя и разработчика и, возможно, другие присоединённые атрибуты (город разработчика, возрастное ограничение жанра и т.п.).

Однако, при добавлении или изменении данных **можно вводить ID.** Но если планируете и при вводе и обновлении данных тоже указывать атрибуты с данными из другой таблицы (например, текстовые, в частности, наименование разработчика), а вам при добавлении и обновлении нужны ID, то перед выполнением основного запроса нужно сделать **запрос поиска ID по введённым данным**, а потом этот ID использовать в основном запросе добавления или обновления.

7. Реализовать получение результатов от**2 аналитических запросов** (с передачей параметров в запрос, полученных от пользователя) из **функциональных требований.**Здесь, аналогично, **не нужно выводить ID внешних ключей,**а атрибуты с контентом (данными)

8. **Протестировать**логику взаимодействия с БД, с помощью проверки данных в БД убедиться, что все функции приложения выполняются корректно. Методы, в которых содержатся запросы к БД должны возвращать корректные данные, должна корректно производиться запись, создаваться корректные связи между сущностями БД

9. В протокол включить скриншоты с результатами работы приложения и**описание используемых команд консольного приложения**(что нужно ввести для добавления, редактирования, вызова аналитических запросов и т.п.), общее описание алгоритмов работы приложения

10. Демонстрацию готового приложения можно проводить на ноутбуке как обычно. Если для этого планируется использовать компьютер в аудитории, то для избежания возможных затруднений с наличием нужной СУБД можно использовать **SQLite**. Свою БД создать в ней в виде файла и работать из приложения с этим файлом с помощью библиотеки для SQLite

11. **Предварительно протестировать своё приложение на компьютере** в аудитории  перед лабораторной работой (если планируется показывать на нём) во избежание возможных проблем с запуском приложения

11. предварительно протестировать своё приложение на компьютере в аудитории









