**Задание 3. Простая БД**

Создайте клиентское приложение для работы с БД (СУБД).

**Требования**

* Разделение представления и модели (данных и методов работы с ними).
* Одна таблица с 4+ полями.
* Собственный формат БД или SQL (QSLite или серверная БД)
* Добавление, проверка, изменение, удаление данных
* Поиск, сортировка (как минимум по одному полю).
* Документация (в коде) описывающая формат данных в файле (если используется).
* **Требования к GUI**:
  + вывод данных в таблицу (но не хранение в этом компоненте!)
  + меню приложения
  + панель инструментов
  + шрифт и цветовая палитра отличные от задаваемых по умолчанию.
  + иконка приложения
  + Всплывающая подсказка или подсказка в строке состояния для элементов интерфейса. ◦ Информация о разработчике. ◦ *Дополнительно (выполните минимум 2 пункта, макс. оценка на экзамене 3, если не выполнено это условие)*: ▪ горячие клавиши ▪ цветовое кодирование данных в таблице ▪ использование элементов интерфейса (флажок, числовое поле ввода и т.п.) в таблице ▪ хранение изображений в БД ▪ использовать как минимум одно модальное окно ▪ Автоматическое сохранение БД через заданные интервалы времени ▪ краткая справка

Вопросы 1. Что такое представление и модель? 2. Как представлена модель в программе? Как происходит проверка данных? 3. Какие исключительные ситуации могут возникнуть во время работы программы? 4. Что такое SOLID? Соблюдаются ли эти принципы в вашей программе?

Ссылки - Считывание данных из файла CSV и их представление через QstandardItemModel evileg.com/ru/post/158 - raw.githubusercontent.com/VetrovSV/OOP/master/Qt\_lec2.pdf - Пример на Qt: github.com/VetrovSV/OOP/tree/master/examples/Qt/database - DataGrid и BindingList в C#: github.com/VetrovSV/OOP/tree/master/examples/WpfApp\_DataGrid - Qt StyleSheet ◦ doc.qt.io/qt-5/stylesheet-examples.html ◦ doc.qt.io/qt-5/stylesheet-reference.html ◦ Пример: github.com/VetrovSV/OOP/blob/master/examples/Qt/database/style.qss