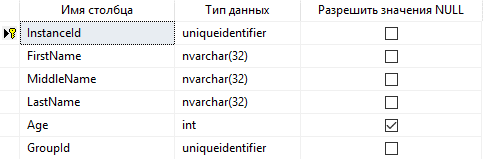
## 1. Сущности

За СУБД был взят “Microsoft SQLEXPRESS 2014”, а визуальная среда работы с ним “SQLServer 2014 Management Studio».

Рассмотрим все используемые элементы и их связи:

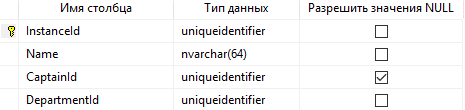
1. Таблица *students* (Таблица 2.1):



*Таблица 2.1*

В данной таблице хранятся все студенты. Данная таблица имеет внешний ключ *GroupId*, который связан с таблицей groups с полем *InstanceId*.

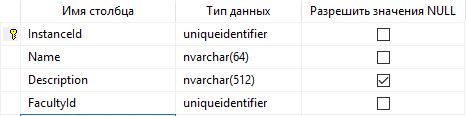
1. Таблица *groups* (Таблица 2.2):



*Таблица 2.2*

В данной таблице хранятся все группы. Данная таблица имеет два внешних ключа: *CaptainId*, который связан с таблицей *student* полем *InstanceId*; *DepartmentId* – связан с таблицей *departments* по полю *InstanceId*.

1. Таблица *departments* (Таблица 2.3):



*Таблица 2.3*

В данной таблице хранятся все кафедры. Данная таблица имеет внешний ключ *FacultyId* на таблицу *faculties* по полю *InstanceId*.

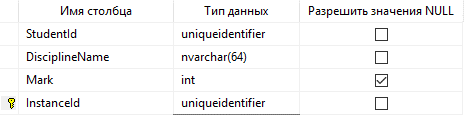
1. Таблица *faculties* (Таблица 2.4)



*Таблица 2.4*

*В данной таблице хранятся все факультеты.*

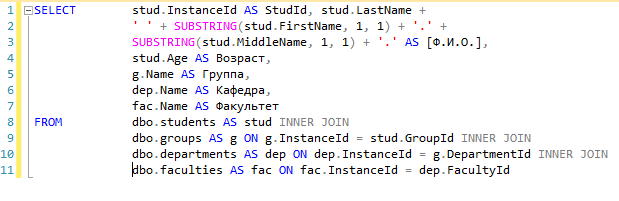
1. Таблица *performance*(Таблица 2.5)



*Таблица 2.5*

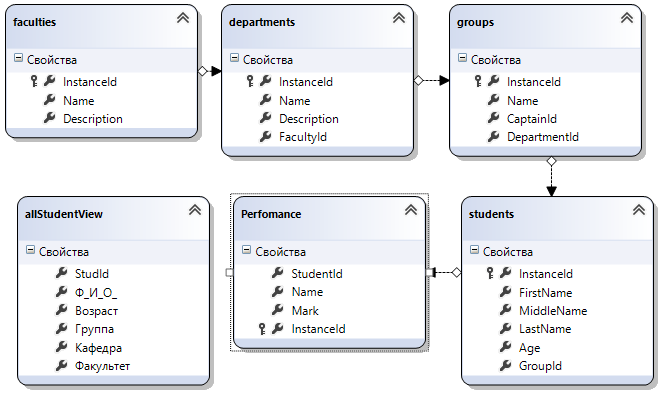
В данной таблице ведется учет успеваемости студентов. Данная таблица имеет внешний ключ *StudentId* на таблицу *students* по полю *InstanceId*.

1. Представление *allStudentView*, которое формируется из запроса, указанного ниже (Рис 2.1):



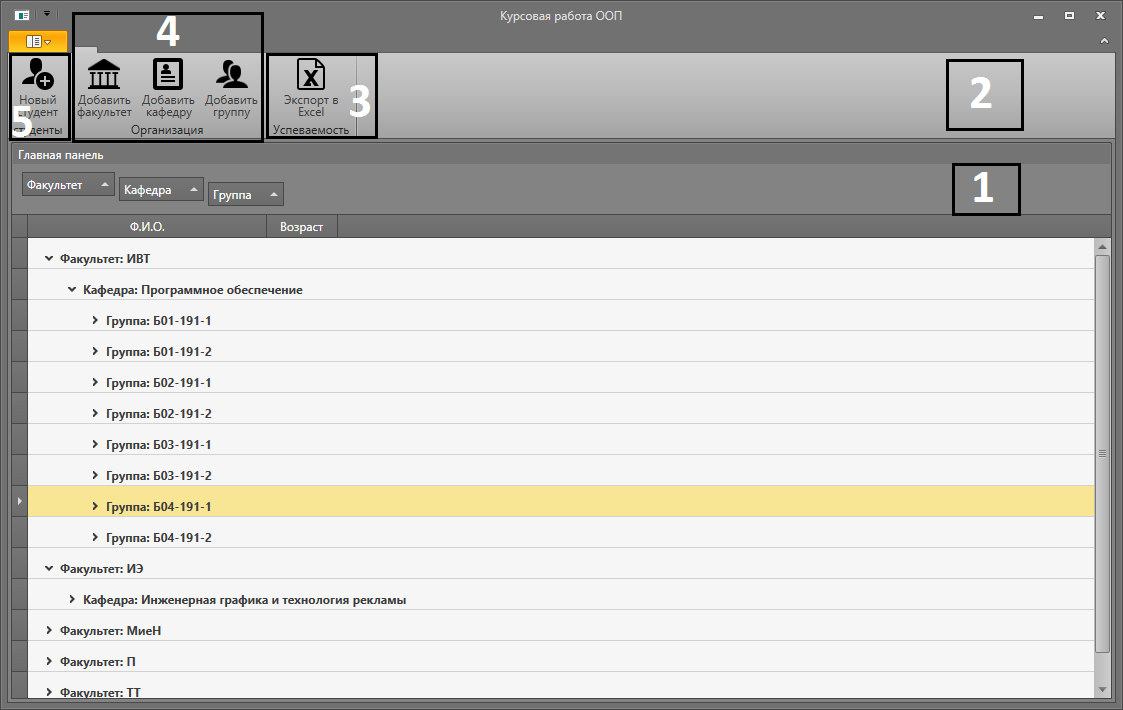
*РИС 2.1*

## 2. Диаграмма сущностей



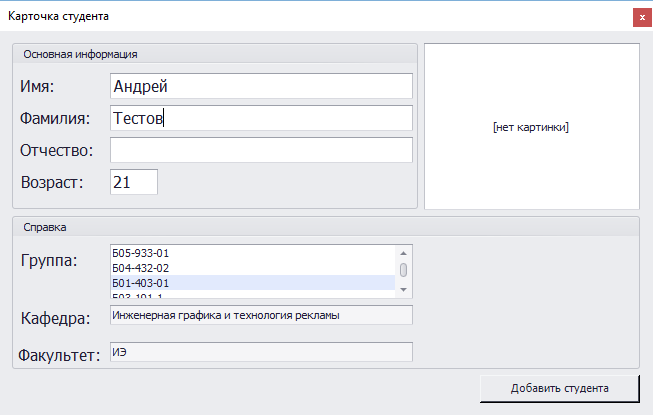
Ниже представлен интерфейс программы, с который пользователь видеть при открытии программы. Можно заметить, что в таблице, в которой отображаются все студенты, автоматически проведена группировка по факультетам, кафедрам, группам (Рис 5.1):

3. Прототипы экранных форм

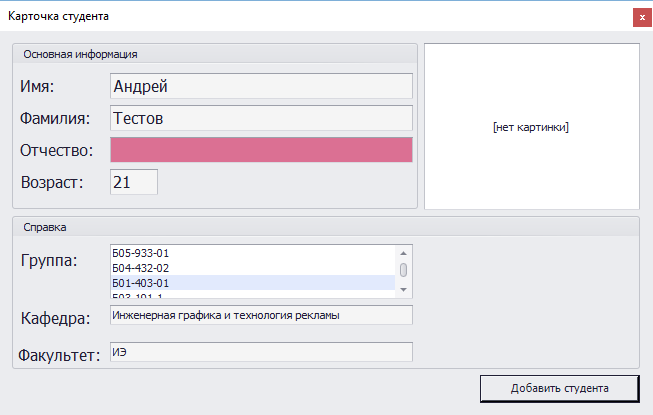


1 – Главная таблица отображаемых данных; 2 – Панель управления кнопками; 3 – группа кнопок «успеваемость»; 4 – группа кнопок «организация»; 5 – группа кнопок «студенты».

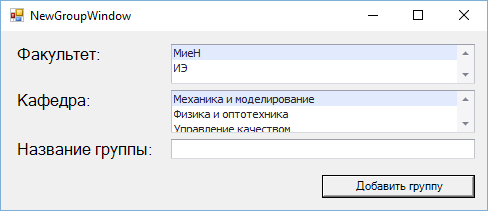
Для примера добавим нового студента. Для этого на панели управления кнопками, в группе «студенты», нажимаем на кнопку новый студент. У нас открывается следующее окно (Рис 5.2):

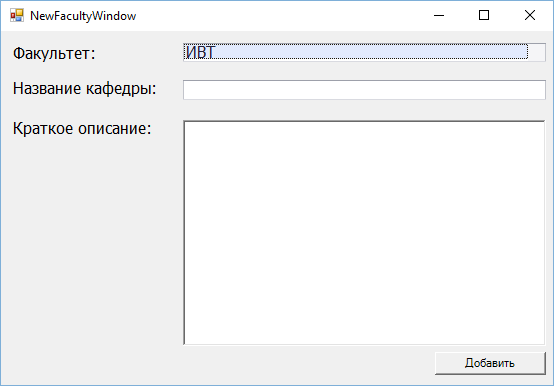


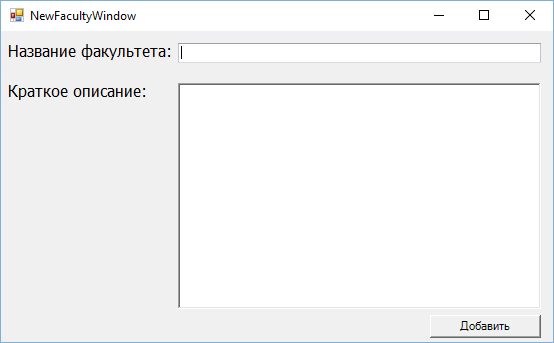
При выборе группы из списка, кафедра и факультет автоматически подгружаются. В данном примере намерено не заполняем поле «Отчество» и нажимаем на кнопку «Добавить студента». Поле «Отчество» было окрашено в красный цвет, что подсказывает о необходимости заполнить это поле (Рис 5.3). За функцию проверки введённых данных отвечает класс *Validator*.



Также имеется с помощью визуального интерфейса добавлять новую группу (Рис 5.4), кафедру (Рис 5.5), факультет (Рис 5.6). Все эти окна проверяются одним классом *Validator*.







1. Действия в системе

|  |  |  |  |
| --- | --- | --- | --- |
| Название действия | Краткое описание | Входная инф. | Выходная инф. |
|  |  |  |  |
| Добавление нового студента | Добавляет нового студента в систему | ФИО, Возраст, Группа, Кафедра, Факультет | Успех операции |
| Удаление студента | Удаляет студента из таблицы | Id | Успех операции |
| Добавление новой группы | Добавляет группу в таблицу групп | Название группы,  Родительская кафедра,  Родительский факультет | Успех операции |
| Добавление новой кафедры | Добавляет кафедру в таблицу кафедр | Название кафедры, Описание кафедры, Родительский факультет | Успех операции |
| Удаление кафедры | Удаляет кафедры из таблицы кафедр | Id | Успех операции |
| Добавление нового факультета | Добавление нового факультета в таблицу факультетов | Название факультета, описание факультета | Успех операции |
| Удаление факультета | Удаление факультета из таблицы | Id | Успех операции |
| Добавить/изменить оценку студенту | Работает с оценкой | Id студента, id предмета,  Значение оценки | Сообщение об ошибке/успехе |
| Добавление/удаление предмета | Добавление/удаление записи в таблице предметов | Id, наименование | Сообщение об ошибке/успехе |
| Получение списка оценок | Получает таблицу оценок у стеднта | Id студентв | Таблица с предметами и оцеками |

1. Иерархическая структура работ (ИСР)  
   * + 1. Разработка технического задания

1.1. Определение сроков разработки

1.2. Определение принципа работ

1.3. Определение требований к системе

1.4. Определение бюджета проекта

* + - 1. Проектирование базы данных

2.1. Создание диаграммы «Сущность-связь»

2.2. Создание базы данных

2.3. Заполнение первоначальных данных

* + - 1. Проектирование интерфейса

3.1. Пользовательский интерфейс

3.1.1. Форма «Главного окна»

3.1.2. Форма «Карточка студента»

3.1.3. Форма «Новая группа»

3.1.4. Форма «Новая кафедра»

3.1.5. Форма «Новый факультет»

3.1.6. Форма «Представление студентов»

3.1.7. Форма «Оценки студентов»

3.2. Интерфейс администратора

3.2.1. Форма «Окно администратора»

4. Разработка программной реализации

4.1. Добавление данных

4.2. Валидация

4.3. Удаление данных

4.4. Аналитика и выгрузка данных

5. Настройка и запуск сервера

6. Тестирование

6.1. Тестирование пользовательской части

6.2. Тестирование администраторской части

6.3. Тестирование сервера

6.4. Тестирование базы данных

6.5. Нагрузочное тестирование

7. Подготовка документации

8. Размещение готового приложения