**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

**к индивидуальному заданию**

**по дисциплине «Базы данных»**

**тема: «СОЗДАНИЕ ПРИЛОЖЕНИЯ»**

Вариант 15

|  |  |  |
| --- | --- | --- |
| Студент гр. 6304 |  | Рыбин А.С. |
| Преподаватель |  | Фомичева Т.Г. |

Санкт-Петербург

2018

**СОДЕРЖАНИЕ**

[**1. Исходная постановка задачи** 3](#_Toc532501594)

[**2.** **Проектирование БД** 5](#_Toc532501595)

[2.1. ER диаграмма 5](#_Toc532501596)

[*2.1.1.* *Сущности* 6](#_Toc532501597)

[*2.1.2.* *Связи между сущностями* 6](#_Toc532501598)

[2.2. Отношения 7](#_Toc532501599)

[*2.2.1.* *Получение отношений* 7](#_Toc532501600)

[*2.2.2.* *Распределение атрибутов* 9](#_Toc532501602)

[*2.2.3.* *НФБК* 10](#_Toc532501603)

[3. **Создание БД в MS Access** 11](#_Toc532501604)

[3.1. Структура таблиц 11](#_Toc532501605)

[3.2. Постоянные связи между таблицами 13](#_Toc532501606)

[3.3. Заполнение таблиц данными 14](#_Toc532501607)

[4. **Элементы БД** 18](#_Toc532501608)

[4.1. Описание экранных форм 18](#_Toc532501609)

[4.2. Описание запросов 26](#_Toc532501610)

[4.3. Описание отчетов 41](#_Toc532501611)

5. **Выводы**...……..……………………………………………………………..50

# Исходная постановка задачи

Пусть требуется создать программную систему, предназначенную для работника деканата. Такая система должна обеспечивать хранение сведений о группах и студентах, а также о результатах текущей сессии. Таким образом, для каждого студента должны храниться такие данные, как фамилия, имя отчество студента, номер его зачетной книжки, адрес постоянной прописки и адрес, по которому студент проживает, получает или нет стипендию, а также оценки, полученные в текущей сессии, и отметки о сданных зачетах. Сведения о группе – это номер группы, факультет, кафедра, специальность, к которым она относится, год формирования группы.

Работник деканата может вносить в БД следующие изменения:

* Удалить или добавить в базу студента;
* Поменять студенту номер группы, специальность, кафедру, номер зачетки;
* Занести оценки, полученные студентами на экзаменах по каждому предмету;
* По результатам сессии начислить стипендии студентам, не имеющим троек или иногородним студентам, которые имеют не более одной тройки. Право на 50 % повышение стипендии имеют студенты, получившие в сессию не более двух четверок, а на 100 % повышение – студенты, сдавшие сессию на все пятерки.

Работнику деканата могут потребоваться следующие сведения:

* Студенты, обучающиеся на определенной кафедре и не сдавшие хотя бы один экзамен, с указанием группы и предмета, по которому оценка отсутствует или равна 2?
* Средний балл студентов каждой группы указанного факультета?
* Средний балл по каждому предмету?
* Список студентов указанной кафедры, которые по итогам сессии могут получать стипендию?
* Список студентов, подлежащих отчислению (не сдано более двух предметов) по всему факультету.
* Количество студентов факультета, не получающих стипендию, получающих обычную стипендию, с 50 % надбавкой и со 100 % надбавкой.
* По какому предмету больше всего неудовлетворительных оценок?
* Какой предмет студенты сдали лучше, чем другие предметы?

Необходимо предусмотреть возможность получения отчета о студентах каждой группы факультета, получающих стипендию, с указанием оценок, полученных ими, и подсчетом среднего балла каждого студента, среднего балла группы и факультета. Следует также подсчитать в группе и на факультете общее количество студентов, количество студентов, получающих обычную стипендию, получающих повышенную стипендию и не получающих стипендию. Подготовить, используя шаблон и данные из БД, приказ на отчисление студентов, не сдавших более двух предметов.

# Проектирование БД

# ER диаграмма

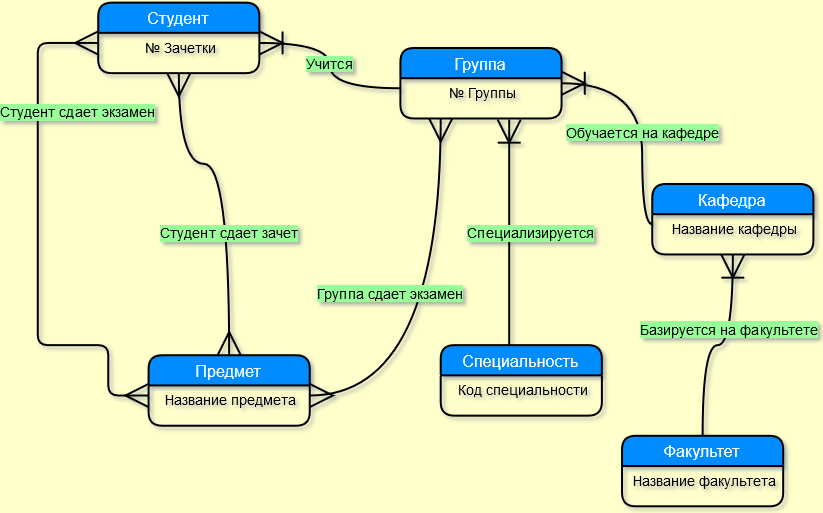


Рисунок 1 – ER диаграмма

# *Сущности*

При анализе предметной области можно выделить следующие сущности:

* **Студент** с идентификатором: Номер зачетной книжки и атрибутами: *ФИО, Адрес проп., Адрес прож., Иногородний*.
* **Группа** с идентификатором: Номер группы и атрибутом: *Год создания*.
* **Специальность** с идентификатором: Код специальности и атрибутами: *Название специальности, Квалификация*.
* **Предмет** с идентификатором: Название предмета и атрибутом *Количество часов*.
* **Кафедра** с идентификатором: Название кафедры и атрибутом *Название кафедры*.
* **Факультет** с идентификатором: Название факультета и атрибутом *Название Факультета*.

# *Связи между сущностями*

Между этими сущностями можно выделить следующий набор связей:

* **Учится** – связь «один ко многим» между сущностями **Студент** и **Группа**. Каждый студент учится в какой-нибудь группе, но только в одной. В каждой группе учится один и более студентов. Связь является обязательной с обеих сторон.
* **Студент сдает зачёт** – связь «многие ко многим» между сущностями **Студент** и **Предмет**. Каждый студент сдает один и более зачётов по какому-нибудь предмету. Каждый зачёт сдается одним и более студентов. Связь является обязательной с обеих сторон.
* **Студент сдает экзамен –** связь «многие ко многим» между сущностями **Студент** и **Предмет**. Каждый студент сдает один и более экзаменов по какому-нибудь предмету. Каждый предмет сдается одним и более студентов. Связь является обязательной с обеих сторон.
* **Группа сдает экзамен –** связь «многие ко многим» между сущностями **Группа** и **Предмет**. Каждая группа сдает один и более экзаменов по какому-нибудь предмету. Каждый экзамен сдается одной и более групп. Связь является обязательной с обеих сторон.
* **Специализируется –** связь «один ко многим» между сущностями **Специальность** и **Группа**. Каждая группа имеет одну специальность. К одной специальности может относиться несколько групп. Связь является обязательной с обеих сторон.
* **Обучается на кафедре –** связь один ко многим между сущностями **Кафедра** и **Группа**. Каждая группа принадлежит какой-нибудь одной кафедре. К каждой кафедре принадлежит одна и более групп. Связь является обязательной с обеих сторон.
* **Базируется на факультете –** связь «один ко многим» между сущностями **Факультет** и **Кафедра.** Каждая кафедра находится на каком-нибудь одном факультете. К каждому факультету принадлежит одна и более кафедр. Связь является обязательной с обеих сторон.

# Отношения

# *Получение отношений*

По правилам генерации отношений из ER-диаграмм получаем следующие отношения:

* Из связи **Учится** получаем отношения:
  + Студент (№ Зачетки, № Группы)
  + Группа (№Группы)
* Из связи **Студент** **сдает экзамен** получаем отношения:
  + Студент (№ Зачетки)
  + Предмет (Название предмета)
  + Сдача\_экзамена (№ Зачетки, Название предмета)
* Из связи **Студент сдает зачет** получаем отношения:
  + Студент (№ Зачетки)
  + Предмет (Название предмета)
  + Зачеты (№ Зачетки, Название предмета)
* Из связи **Группа сдает экзамен** получаем отношения:
  + Группа (№ Группы)
  + Предмет (Название предмета)
  + Экзамен (№ Группы, Название предмета)
* Из связи **Специализируется** получаем отношения:
  + Группа (№ Группы, Код специальности)
  + Специальность (Код специальности)
* Из связи **Обучается на кафедре** получаем отношения:
  + Группа (№ Группы, Кафедра)
  + Кафедра (Название кафедры)
* Из связи **Базируется на факультете** получаем отношения:
  + Кафедра (Название кафедры, Факультет)
  + Факультет (Название факультета)

# В результате получаем следующие отношения:

* Студент (№ Зачетки, № Группы)
* Группа (№Группы)
* Студент (№ Зачетки)
* Предмет (Название предмета)
* Сдача\_экзамена (№ Зачетки, Название предмета)
* Студент (№ Зачетки)
* Предмет (Название предмета)
* Зачет (№ Зачетки, Название предмета)
* Группа (№ Группы)
* Предмет (Название предмета)
* Экзамен (№ Группы, Название предмета)
* Группа (№ Группы, Код специальности)
* Специальность (Код специальности)
* Группа (№ Группы, Кафедра)
* Кафедра (Название кафедры)
* Кафедра (Название кафедры, Факультет)
* Факультет (Название факультета)

Оставим нужные отношения:

* Студент (№ Зачетки, № Группы)
* Группа (№ Группы, Название кафедры, Название факультета, Код специальности)
* Предмет (Название предмета)
* Специальность (Код специальности)
* Сдача\_экзамена (№ Зачетки, Название предмета)
* Зачет (№ Зачетки, Название предмета)
* Экзамен (№ Группы, Название предмета)

# *Распределение атрибутов*

Теперь разместим в этих отношениях пока еще не попавшие в них атрибуты предметной области. В результате получим следующие отношения:

* Студент (№ Зачетки, № Группы, ФИО, Адрес прож., Адрес проп., Иногородний)
* Группа (№ Группы, Название кафедры, Название факультета, Код специальности)
* Предмет (Название предмета, Количество часов)
* Специальность (Код специальности, Название специальности, Квалификация)
* Сдача\_экзамена (№ Зачетки, Название предмета, Оценка)
* Зачет (№ Зачетки, Название предмета, Зачет)
* Экзамен (№ Группы, Название предмета, Дата, Аудитория, Время)

Все атрибуты распределены. Далее следует рассмотреть отношения на предмет принадлежности к НФБК.

# *НФБК*

Видно, что Студент, Группа, Специальность, Предмет, Сдача\_экзамена, Зачеты и Экзамен находятся в НФБК. Проверим отношение Студент. В нем можно выделить следующие функциональные зависимости:

№ зачетки => ФИО

№ зачетки => Номер группы

№ зачетки => Адрес проп.

№ зачетки => Адрес прож.

№ зачетки => Иногородний

Получим отношение:

Студент (№ Зачетки, № Группы, ФИО, Адрес прож., Адрес проп., Иногородний)

В силу того, что факультет однозначно определяется кафедрой, имеются функциональные зависимости № группы => Факультет и № группы => Кафедра, поэтому их следовало бы вынести в отдельную таблицу. Однако для удобства последующей фильтрации студентов по факультетам и кафедре выделять эти отношения в самой БД не стоит.

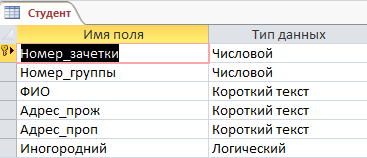
Код специальности однозначно определяет её название, значит, в качестве ключа отношения Специальность достаточно определить именно её код.

Проектирование можно считать законченным.

# Создание БД в MS Access

# Структура таблиц

* Таблица «Студент» содержит информацию о студентах:

Рисунок 2 – Структура таблицы «Студент»

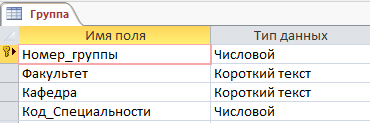
* Таблица «Группа» содержит информацию о группах:

Рисунок 3 – Структура таблицы «Студент»

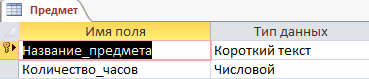
* Таблица «Предмет» содержит информацию о предметах, которые преподаются в университете:

Рисунок 4 – Структура таблицы «Предмет»

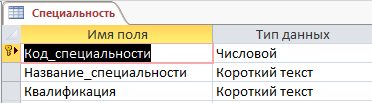
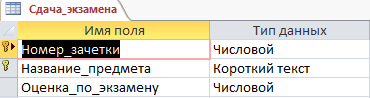
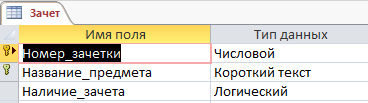
* Таблица «Специальность» содержит информацию о специальностях, существующих в университете:

Рисунок 5 – Структура таблицы «Специальность»

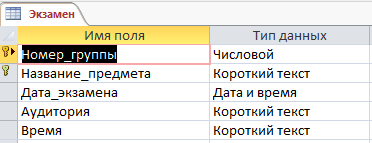
* Таблица «Сдача\_экзамена» содержит информацию о оценках, которые студенты получили за экзамены:

Рисунок 6 – Структура таблицы «Сдача\_экзамена»

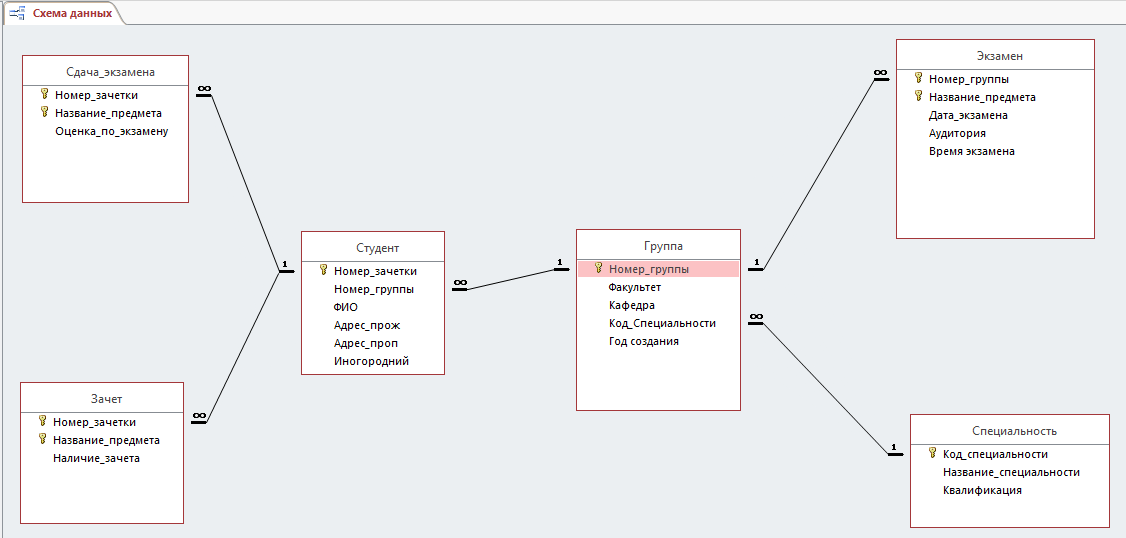
* Таблица «Зачет» содержит информацию о зачетах, которые получили или нет студенты:

Рисунок 7 – Структура таблицы «Зачет»

* Таблица «Экзамен» содержит расписание экзаменов для всех групп:

Рисунок 8 – Структура таблицы «Экзамен»

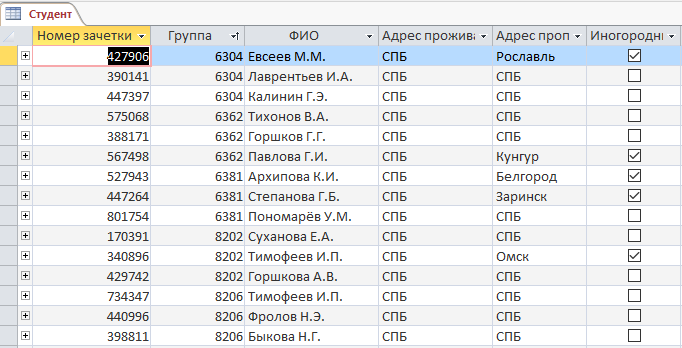
# Постоянные связи между таблицами

Рисунок 9 – Постоянные связи между таблицами

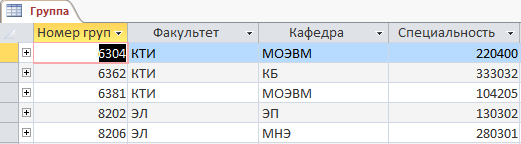
# Заполнение таблиц данными

Для отладки системы необходимо заполнить таблицы тестовыми данными.

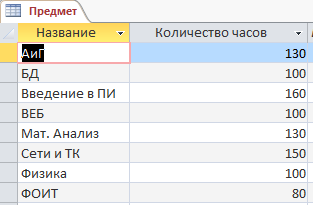
* Таблица «Студент»:

Рисунок 10 – Содержимое таблицы «Студент»

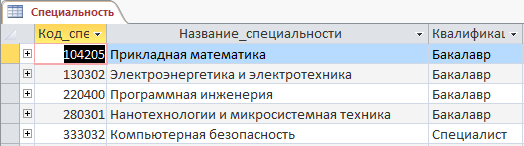
* Таблица «Группа»:

Рисунок 11 – Содержимое таблицы «Группа»

* Таблица «Предмет»:

Рисунок 12 – Содержимое таблицы «Предмет»

* Таблица «Специальность»:

Рисунок 13 – Содержимое таблицы «Специальность»

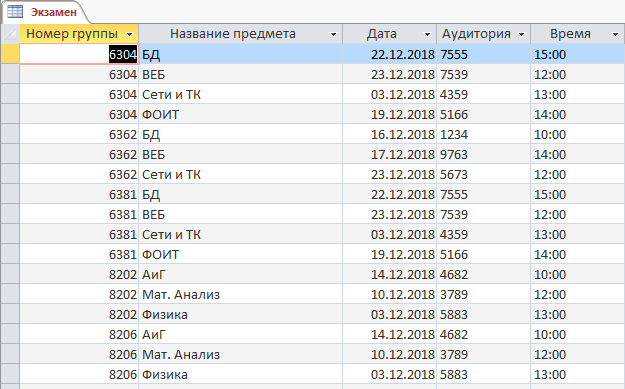
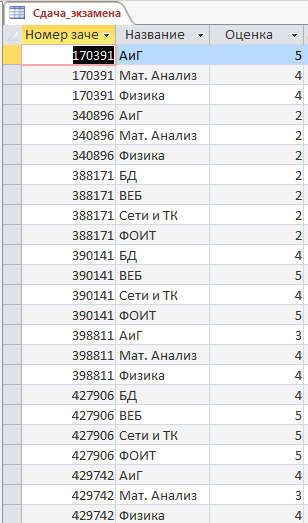
* Таблица «Экзамен»:

Рисунок 14 – Содержимое таблицы «Экзамен»

* Таблица «Сдача\_экзамена»:

Рисунок 15 – Содержимое таблицы «Сдача\_экзамена»

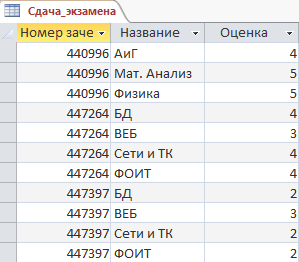
Рисунок 16 – Содержимое таблицы «Сдача\_экзамена»

Рисунок 17 – Содержимое таблицы «Сдача\_экзамена»

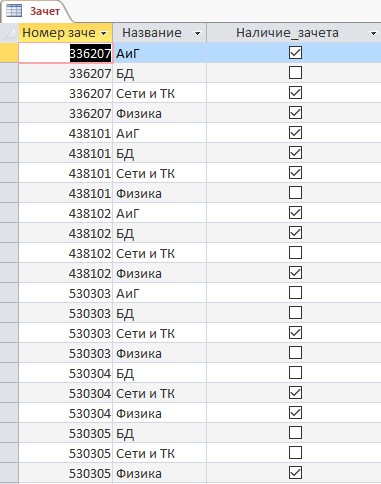
* Таблица «Зачет»:

Рисунок 18 – Содержимое таблицы «Зачет»

# Элементы БД

# Описание экранных форм

База данных содержит следующие формы:

**Главная** **форма** содержит следующие кнопки:

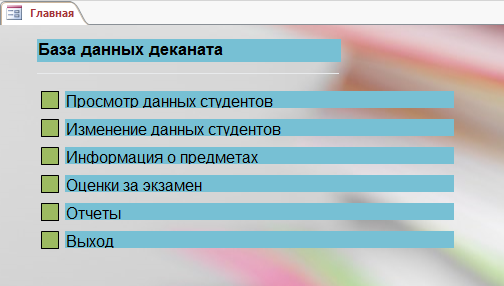
* Кнопка “Просмотр данных студентов” открывает форму “Студент”.
* Кнопка “Изменение данных студентов” открывает форму “Работа со студентами”.
* Кнопка “Информация о предметах” открывает форму “Предмет”.
* Кнопка “Оценки за экзамен” открывает форму “Сдача экзамена”.
* Кнопка “Отчеты” открывает кнопочную форму “Отчеты”.
* Кнопка “Выход” закрывает главную форму.

Рисунок 19 – Главная форма

**Форма Студент** (пункт меню “Просмотр данных студентов” в главной форме) выводит информацию о студентах, включающую в себя также и стипендию, которую они получают.

**Источник данных** – запрос “Для формы студент”.

**Кнопки**. Кнопки “КнопкаКАФПрименить” и “КнопкаФАКПрименить”, расположенные слева от списков вызывают макрос “ФильтрКафедры\_Факультета”, который в зависимости от значения группы переключателей (“Группа”) и значений в списках производит следующие действия:

*Условие отбора*: [Forms]![Студент]![Группа54]=1

*Применить фильтр*:

[Для формы студент].[Кафедра]=[Forms]![Студент]![ПолеКафедра] And [Для формы студент].[Двойки]>0

*Условие отбора*: [Forms]![Студент]![Группа54]=2

*Применить фильтр*:

[Двойки] Is Null And ([Тройки] Is Null Or ([Иногородний]=1 And ([Тройки] Is Null Or [Тройки]=1))) And [Для формы студент].[Кафедра]=[Forms]![Студент]![ПолеКафедра]

*Условие отбора* : [Forms]![Студент]![Группа54]=3

*Применить фильтр*:

[Двойки]>2 And [Для формы студент].[Факультет]=[Forms]![Студент]![ПолеФакультет]

Кнопка “Показать все” вызывает макрос “Показать все”, в котором содержится одна макрокоманда “ПоказатьВсеЗаписи”.

Кнопка “Выход” использует стандартную процедуру обработки события закрытия формы.

**Списки**. Два выпадающих списка “ПолеКафедра” и “ПолеФакультет”, расположенные слева от надписей “Кафедра” и “Факультет” соответственно (Источники данных – поля Кафедра и Факультет таблицы Группы) содержат списки кафедр и факультетов соответственно.

**Подчиненные формы**.

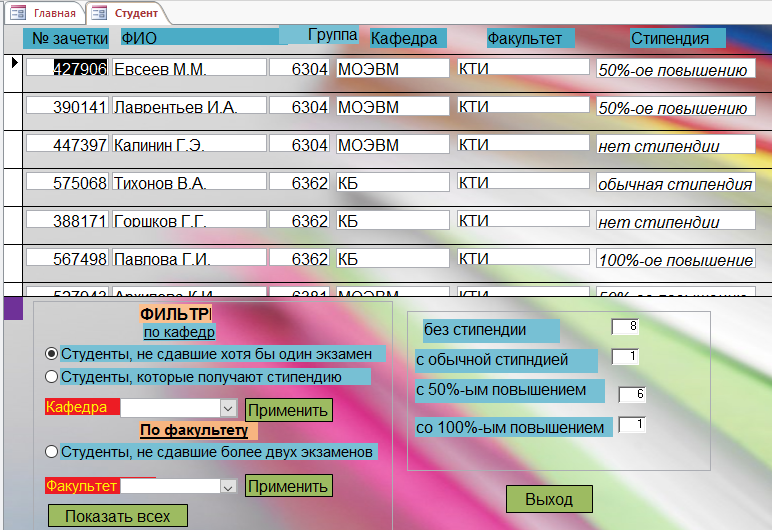
Форма содержит 4 подчиненные формы:

“Количество студентов без стипендии”

“подчиненная форма Количество студентов с 100%-ым повышением”

“подчиненная форма Количество студентов с 50%-ым повышением”

“подчиненная форма Количество студентов с обычной стипендией”

Рисунок 20 – Форма студент

**Форма Работа со студентом** (пункт меню “Изменение данных студента” в главной форме) позволяет добавлять и удалять студента в базу данных, а также изменять его данные.

**Источник данных** – таблица “Студент”.

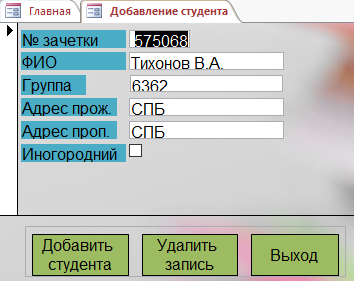
**Кнопка** “Добавить студента” открывает форму “Добавление студента”

**Кнопка** “Удалить запись” использует стандартную процедуру обработки удаления данных из таблицы.

**Кнопка** “Выход” использует стандартную процедуру обработки события закрытия формы.

**Форма Добавление студента** (открывается при нажатии на кнопку “Добавить студента” в форме “Работа со студентом”) позволяет добавить студента.

**Источник данных** – таблица “Студент”

Рисунок 21 – Форма «Работа со студентом»

**Кнопки.**

Кнопка “Добавить” вызывает макрос “Добавление студента” который осуществляет проверку на то, что все значения были введены в соответствующие поле, и, если это не было сделано, выводит сообщение об ошибке, при этом выполнение макроса приостанавливается до тех пор, пока не будут введены все значения. Например, проверка на то, что было введено значение в поле зачетки описано следующим образом:

*Условие*: [Forms]![Добавление студента]![Номер\_зачетки] Is Null

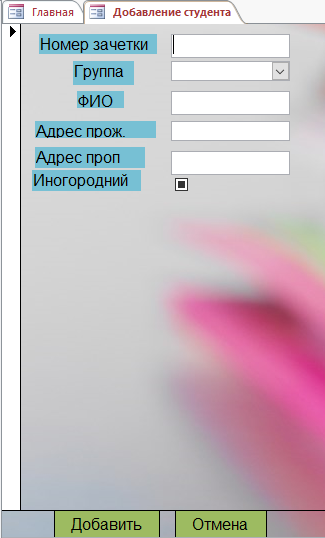
*Сообщение*: необходимо ввести номер зачетки!

*ОстановитьМакрос*

Если все значения были введены, то выполняется запуск запроса SQL:

INSERT INTO Студент VALUES (Номер\_зачетки, Номер\_группы, ФИО, адрес\_прож, Адрес\_проп, Иногородний)

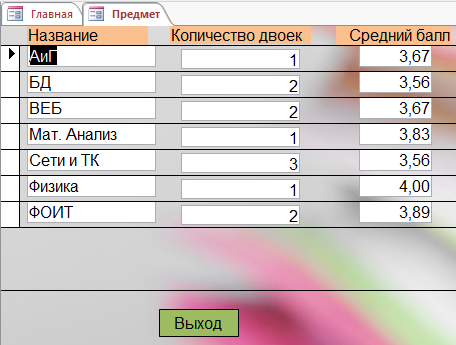
Кнопка “Выход” использует стандартную процедуру обработки события закрытия формы.

Рисунок 22 – Форма «Добавление студента»

**Форма Предмет** (пункт меню “Информация о предметах” в главной форме) показывает список всех предметов, а также средний балл и количество двоек по каждому из них.

**Источник данных** – запрос “Для формы предмет”

**Кнопки**. Кнопка “Выход” использует стандартную процедуру обработки события закрытия формы.

Рисунок 23 – Форма «Предмет»

**Форма Сдача экзамена** (пункт меню “Добавление экзамена” главной формы) позволяет добавить оценку по экзамену любому студенту.

Выпадающий **список** “Номер\_зачетки” (источник данных – поле Номер зачетки таблицы Студент) справа от надписи “Номер зачетки” содержит номера зачеток студентов.

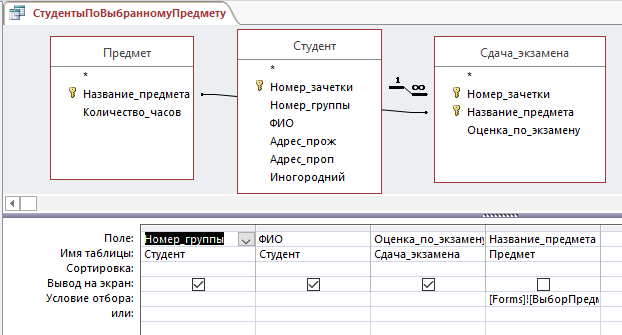
Выпадающий **список** “Название\_предмета” (источник данных – поле Название предмета таблицы Студент) справа от надписи “Название предмета” содержит названия предметов.

Выпадающий **список** “Оценка” содержит фиксированный набор значений: 2, 3, 4, 5.

**Кнопка** “Добавить” вызывает макрос “ДобавлениеОценкиПоЭкзамену”, который проверяет все ли значения введены в соответствующие поля, и, если это не так, выводит соответствующие сообщения, при этом выполнение макроса приостанавливается до тех пор, пока не будут введены все значения. Если все значения введены, то выполняется запуск запроса SQL:

INSERT INTO Сдача\_экзамена VALUES (Номер\_зачетки, Название\_предмета, Оценка)

Фильтрация студентов по предмету происходит при помощи запроса СтудентыПоВыбранному предмету:

Рисунок 24 – Запрос «СтудентыПоВыбранному предмету»

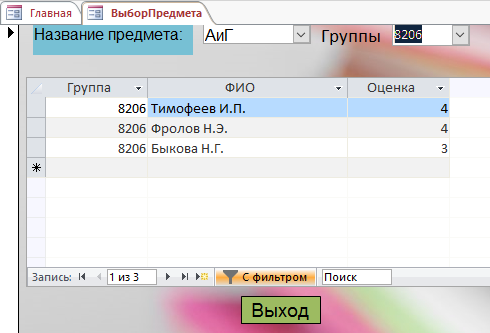
Отбор студентов по группе реализован с помощью запроса SQL Список\_Групп:

SELECT DISTINCT ""

FROM Группа;

UNION ALL SELECT DISTINCT Группа.Номер\_группы

from Группа;

Рисунок 25 – Форма «Сдача экзамена»

**Кнопочная форма Отчеты** (пункт меню “Отчеты” главной формы) позволяет выбрать нужный отчет для вывода на экран. Она содержит следующие кнопки:

Кнопка “Информация о студентах” открывает отчет «Информация о студентах»

Кнопка “Студенты с различной стипендией” открывает отчет “Количество СтудентовСРазнойСтипендией”

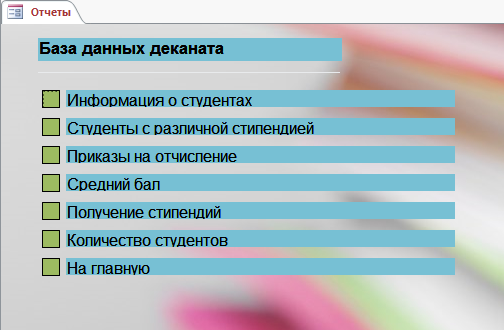
Кнопка “Приказы на отчисление” открывает отчет “Приказ на отчисление”

Кнопка “Средний балл” открывает отчет “Средний балл каждого студента, группы и факультета”

Кнопка “Получение стипендий” открывает отчет “Студенты, получающие стипендию”

Кнопка “Количество студентов” открывает отчет “Количество студентов”

Кнопка “На главную” осуществляет переход к главной форме.

Рисунок 26 – Форма «Отчеты»

# Описание запросов

**Формулировка запроса.** Вывести информацию о студенте, о номере группы, в которой он обучается, и количество всех его оценок (двоек, троек, четверок и пятерок)

**Запрос «Для формы студент»**

Используется как источник данных для формы “Студент”. Содержит таблицы “Студент” и “Группа” и запрос “\_Сколько оценок у студента”

**SQL-код запроса.**

SELECT Студент.\*, [\_Сколько оценок у студента].[\_Сколько двоек у студента].[Count-Оценка\_по\_экзамену1] AS Двойки, [\_Сколько оценок у студента].[\_Сколько троек у студента].[Count-Оценка\_по\_экзамену1] AS Тройки, [\_Сколько оценок у студента].[\_Сколько четверок у студента].[Count-Оценка\_по\_экзамену1] AS Четверки, [\_Сколько оценок у студента].[\_Сколько пятерок у студента].[Count-Оценка\_по\_экзамену1] AS Пятерки, Группа.Факультет, Группа.Кафедра

FROM Группа INNER JOIN (Студент INNER JOIN [\_Сколько оценок у студента] ON Студент.Номер\_зачетки = [\_Сколько оценок у студента].Номер\_зачетки) ON Группа.Номер\_группы = Студент.Номер\_группы;

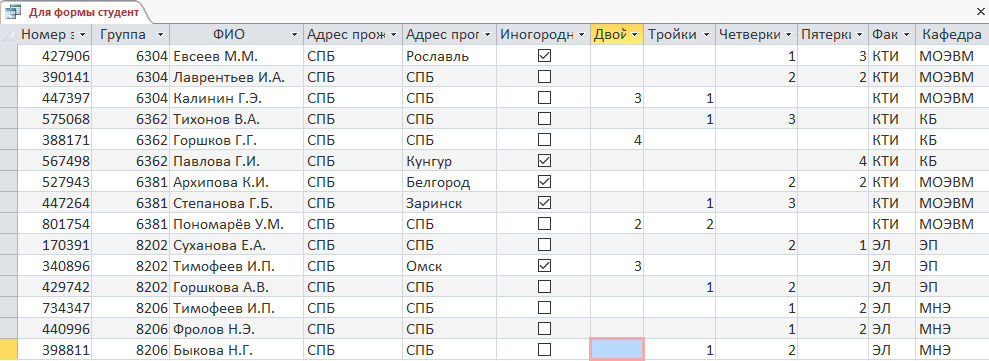
**Результат запроса.**

Рисунок 27 – Запрос «Для формы студент»

**Формулировка запроса.** Вывести номера зачеток студентов, и количество их оценок.

**Запрос «\_Сколько оценок у студента»**

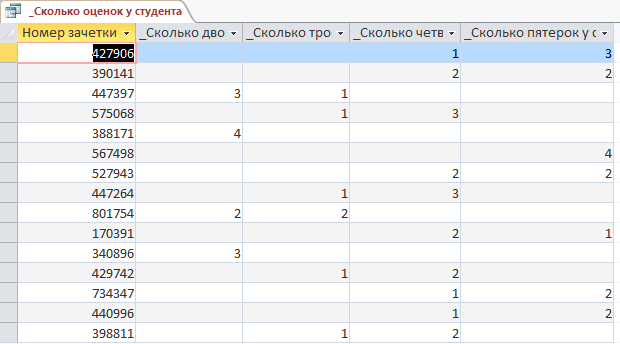
Вспомогательный запрос для запроса “Для формы студент”. Содержит таблицу “Студент” и запросы “\_Сколько двоек у студента”, “\_Сколько троек у студента ”, “\_Сколько четверок у студента”, “\_Сколько пятерок у студента ”

**SQL-код запроса.**

SELECT Студент.Номер\_зачетки, [\_Сколько двоек у студента].[Count-Оценка\_по\_экзамену1], [\_Сколько троек у студента].[Count-Оценка\_по\_экзамену1], [\_Сколько четверок у студента].[Count-Оценка\_по\_экзамену1], [\_Сколько пятерок у студента].[Count-Оценка\_по\_экзамену1]

FROM (((Студент LEFT JOIN [\_Сколько двоек у студента] ON Студент.Номер\_зачетки=[\_Сколько двоек у студента].Номер\_зачетки) LEFT JOIN [\_Сколько троек у студента] ON Студент.Номер\_зачетки=[\_Сколько троек у студента].Номер\_зачетки) LEFT JOIN [\_Сколько четверок у студента] ON Студент.Номер\_зачетки=[\_Сколько четверок у студента].Номер\_зачетки) LEFT JOIN [\_Сколько пятерок у студента] ON Студент.Номер\_зачетки=[\_Сколько пятерок у студента].Номер\_зачетки;

**Результат запроса.**

Рисунок 28 – Запрос «\_Сколько оценок у студента»

**Формулировка запроса.** Вывести номера зачеток студентов и количество двоек у каждого из них.

**Запрос «\_Сколько двоек у студента»**

Вспомогательный запрос для запроса “\_Сколько оценок у студента”. Содержит таблицу “Сдача экзамена”.

**SQL-код запроса.**

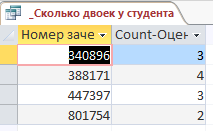
SELECT Сдача\_экзамена.Номер\_зачетки, Count(Сдача\_экзамена.Оценка\_по\_экзамену) AS [Count-Оценка\_по\_экзамену1]

FROM Сдача\_экзамена

GROUP BY Сдача\_экзамена.Номер\_зачетки, Сдача\_экзамена.Оценка\_по\_экзамену

HAVING (((Сдача\_экзамена.Оценка\_по\_экзамену)=2));

**Результат запроса.**

Рисунок 29 – Запрос «\_Сколько двоек у студента»

**Формулировка запроса.** Вывести номера зачеток студентов и количество троек у каждого из них.

**Запрос «\_Сколько троек у студента»**

Вспомогательный запрос для запроса “\_Сколько оценок у студента”. Содержит таблицу “Сдача экзамена”.

**SQL-код запроса.**

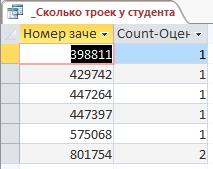
SELECT Сдача\_экзамена.Номер\_зачетки, Count(Сдача\_экзамена.Оценка\_по\_экзамену) AS [Count-Оценка\_по\_экзамену1]

FROM Сдача\_экзамена

GROUP BY Сдача\_экзамена.Номер\_зачетки, Сдача\_экзамена.Оценка\_по\_экзамену

HAVING (((Сдача\_экзамена.Оценка\_по\_экзамену)=3));

**Результат запроса.**

Рисунок 30 – Запрос «\_Сколько троек у студента»

**Формулировка запроса.** Вывести номера зачеток студентов и количество четверок у каждого из них.

**Запрос «\_Сколько четверок у студента»**

Вспомогательный запрос для запроса “\_Сколько оценок у студента”. Содержит таблицу “Сдача экзамена”.

**SQL-код запроса.**

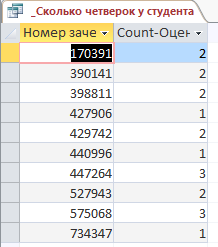
SELECT Сдача\_экзамена.Номер\_зачетки, Count(Сдача\_экзамена.Оценка\_по\_экзамену) AS [Count-Оценка\_по\_экзамену1]

FROM Сдача\_экзамена

GROUP BY Сдача\_экзамена.Номер\_зачетки, Сдача\_экзамена.Оценка\_по\_экзамену

HAVING (((Сдача\_экзамена.Оценка\_по\_экзамену)=4));

**Результат запроса.**

Рисунок 31 – Запрос «\_Сколько четверок у студента»

**Формулировка запроса.** Вывести номера зачеток студентов и количество пятерок у каждого из них.

**Запрос «\_Сколько пятерок у студента»**

Вспомогательный запрос для запроса “\_Сколько оценок у студента”. Содержит таблицу “Сдача экзамена”.

**SQL-код запроса.**

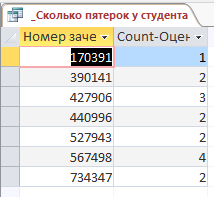
SELECT Сдача\_экзамена.Номер\_зачетки, Count(Сдача\_экзамена.Оценка\_по\_экзамену) AS [Count-Оценка\_по\_экзамену1]

FROM Сдача\_экзамена

GROUP BY Сдача\_экзамена.Номер\_зачетки, Сдача\_экзамена.Оценка\_по\_экзамену

HAVING (((Сдача\_экзамена.Оценка\_по\_экзамену)=5));

**Результат запроса.**

Рисунок 32 – Запрос «\_Сколько пятёрок у студента»

**Формулировка запроса.** Вывести названия предметов, средний балл и количество двоек по каждому из них.

**Запрос «Для формы предмет»**

Используется как источник данных для формы “Предмет”. Содержит таблицу “Сдача экзамена” и Запрос ”Сколько двоек по предмету”. В данном запросе выводятся все предметы, по которым студенты сдают экзамены.

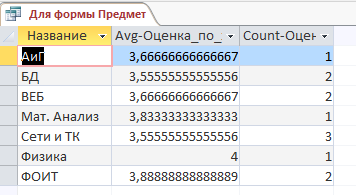
**SQL-код запроса.**

SELECT Сдача\_экзамена.Название\_предмета, Avg(Сдача\_экзамена.Оценка\_по\_экзамену) AS [Avg-Оценка\_по\_экзамену], [Сколько двоек по предмету].[Count-Оценка\_по\_экзамену1]

FROM Сдача\_экзамена LEFT JOIN [Сколько двоек по предмету] ON Сдача\_экзамена.Название\_предмета = [Сколько двоек по предмету].Название\_предмета

GROUP BY Сдача\_экзамена.Название\_предмета, [Сколько двоек по предмету].[Count-Оценка\_по\_экзамену1];

**Результат запроса.**

Рисунок 33 – Запрос «Для формы Предмет»

**Формулировка запроса.** Вывести названия предметов и количество двоек по каждому из них.

**Запрос Сколько двоек по предмету**

Вспомогательный запрос для запроса “Для формы предмет”. Содержит таблицу “Сдача\_экзамена”

**SQL-код запроса.**

SELECT Сдача\_экзамена.Название\_предмета, Count(Сдача\_экзамена.Оценка\_по\_экзамену) AS [Count-Оценка\_по\_экзамену1]

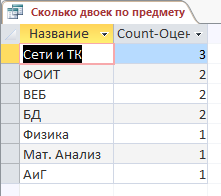
FROM Сдача\_экзамена

GROUP BY Сдача\_экзамена.Название\_предмета, Сдача\_экзамена.Оценка\_по\_экзамену

HAVING (((Сдача\_экзамена.Оценка\_по\_экзамену)=2))

ORDER BY Count(Сдача\_экзамена.Оценка\_по\_экзамену) DESC;

**Результат запроса.**

Рисунок 34 – Запрос «Сколько двоек по предмету»

**Формулировка запроса.** Вывести количество студентов, не получающих стипендии.

**Запрос «\_Количество студентов без стипендии»**

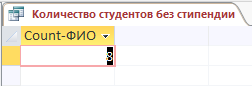
Используется для вспомогательный формы Количество студентов без стипендии” формы “Студент”. Содержит запрос “Студенты без стипендии”

**SQL-код запроса.**

SELECT Count([Студенты без стипендии].ФИО) AS [Count-ФИО]

FROM [Студенты без стипендии];

**Результат запроса.**

Рисунок 35 – Запрос «Количество студентов без стипендии»

**Формулировка запроса.** Вывести ФИО студентов, не получающих стипендию.

**Запрос «Студенты без стипендии»**

Вспомогательный запрос для запроса “\_Количество студентов без стипендии”. Содержит запрос “Для формы студент”

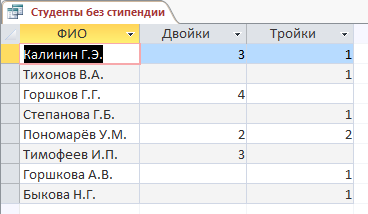
**SQL-код запроса.**

SELECT [Для формы студент].ФИО, [Для формы студент].Двойки, [Для формы студент].Тройки

FROM [Для формы студент]

WHERE ((([Для формы студент].Двойки) Is Null) AND (([Для формы студент].Тройки)>1)) OR ((([Для формы студент].Двойки) Is Null) AND (([Для формы студент].Тройки)>=1)) OR ((([Для формы студент].Двойки)>0));

**Результат запроса.**

****Рисунок 36 – Запрос «Студенты без стипендии»

**Формулировка запроса.** Вывести номер зачётки и ФИО студентов, получающих стипендию, с указанием номера их группы, факультета. Также вывести оценки по экзаменам, указывая названия предметов.

**Запрос для отчета: «Студенты, получающие стипендию»**

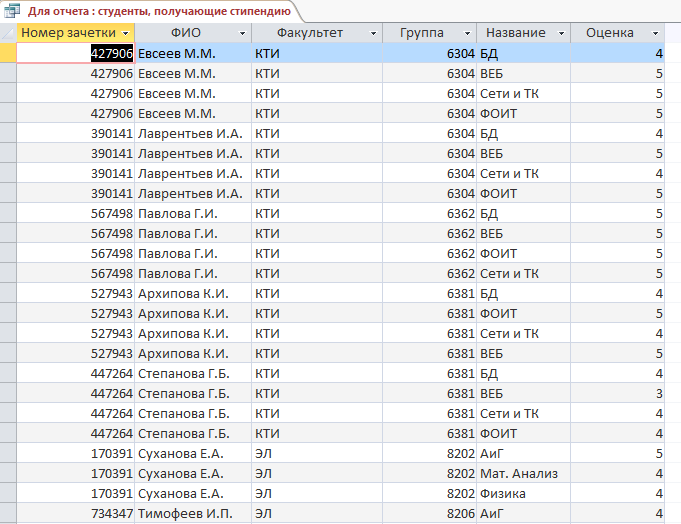
Используется для отчета “Студенты, получающие стипендию”. Содержит таблицы “Студент”, “Группа” и “Сдача\_Экзамена”, а также запрос “\_Сколько оценок у студента”.

**SQL-код запроса.**

SELECT Студент.Номер\_зачетки, Студент.ФИО, Группа.Факультет, Студент.Номер\_группы, Сдача\_экзамена.Название\_предмета, Сдача\_экзамена.Оценка\_по\_экзамену

FROM (Группа INNER JOIN (Студент INNER JOIN [\_Сколько оценок у студента] ON Студент.Номер\_зачетки = [\_Сколько оценок у студента].Номер\_зачетки) ON Группа.Номер\_группы = Студент.Номер\_группы) INNER JOIN Сдача\_экзамена ON Студент.Номер\_зачетки = Сдача\_экзамена.Номер\_зачетки

WHERE ((([\_Сколько оценок у студента].[\_Сколько двоек у студента].[Count-Оценка\_по\_экзамену1]) Is Null) AND (([\_Сколько оценок у студента].[\_Сколько троек у студента].[Count-Оценка\_по\_экзамену1]) Is Null) AND ((Студент.Иногородний)=False)) OR ((([\_Сколько оценок у студента].[\_Сколько двоек у студента].[Count-Оценка\_по\_экзамену1]) Is Null) AND (([\_Сколько оценок у студента].[\_Сколько троек у студента].[Count-Оценка\_по\_экзамену1])=1) AND ((Студент.Иногородний)<>0)) OR ((([\_Сколько оценок у студента].[\_Сколько двоек у студента].[Count-Оценка\_по\_экзамену1]) Is Null) AND (([\_Сколько оценок у студента].[\_Сколько троек у студента].[Count-Оценка\_по\_экзамену1]) Is Null) AND ((Студент.Иногородний)<>0));

Рисунок 37 – Запрос «Для отчета: студенты, получающие стипендию»

**Формулировка запроса.** Вывести количество студентов с обычной стипендией.

**Запрос «Количество студентов с обычной стипендией»**

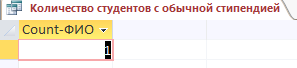
Используется для вспомогательной формы “\_Количество студентов с обычной стипендией” формы “Студент”. Содержит запрос “\_Студенты с обычной стипендией”

**SQL-код запроса.**

SELECT Count([Студенты с обычной стипендией].ФИО) AS [Count-ФИО]

FROM [Студенты с обычной стипендией];

**Результат запроса.**

Рисунок 38 – Запрос «Количество студентов с обычной стипендией»

**Формулировка запроса.** Вывести ФИО студентов, получающих обычную стипендию.

**Запрос «Студенты с обычной стипендией»**

Вспомогательный запрос для запроса “\_Количество студентов без c обычной стипендией”. Содержит запрос “Для формы студент”

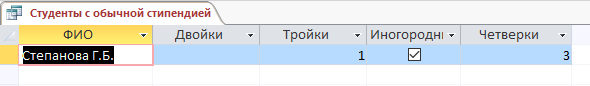
**SQL-код запроса.**

SELECT [Для формы студент].ФИО, [Для формы студент].Двойки, [Для формы студент].Тройки, [Для формы студент].Иногородний, [Для формы студент].Четверки

FROM [Для формы студент]

WHERE ((([Для формы студент].Двойки) Is Null) AND (([Для формы студент].Тройки)=1) AND (([Для формы студент].Иногородний)<>0)) OR ((([Для формы студент].Двойки) Is Null) AND (([Для формы студент].Тройки) Is Null) AND (([Для формы студент].Иногородний)<>0) AND (([Для формы студент].Четверки)>2)) OR ((([Для формы студент].Двойки) Is Null) AND (([Для формы студент].Тройки) Is Null) AND (([Для формы студент].Иногородний)=False) AND (([Для формы студент].Четверки)>2));

**Результат запроса.**

Рисунок 39 – Запрос «Студенты с обычной стипендией»

**Формулировка запроса.** Вывести количество студентов, получающих стипендию с 50%-ой надбавкой.

**Запрос «Количество студентов с 50%-ым повышением»**

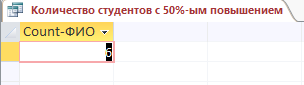
Используется для вспомогательный формы “Количество студентов с 50%-ым повышением” формы “Студент”. Содержит запрос “Студенты с 50%-ым повышением”

**SQL-код запроса.**

SELECT Count([Студенты с 50%-ным повышением].ФИО) AS [Count-ФИО]

FROM [Студенты с 50%-ным повышением];

**Результат запроса.**

****Рисунок 40 – Запрос «Количество студентов с 50% повышением»

**Формулировка запроса.** Вывести ФИО студентов, получающих стипендию с 50%-ой надбавкой.

**Запрос Студенты с 50%-ым повышением**

Вспомогательный запрос для запроса “Количество студентов c 50%-ым повышением”. Содержит запрос “Для формы студент”

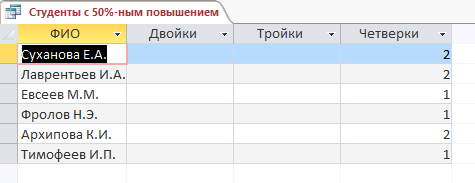
**SQL-код запроса.**

SELECT [Для формы студент].ФИО, [Для формы студент].Двойки, [Для формы студент].Тройки, [Для формы студент].Четверки

FROM [Для формы студент]

WHERE ((([Для формы студент].Двойки) Is Null) AND (([Для формы студент].Тройки) Is Null) AND (([Для формы студент].Четверки)<=2 And ([Для формы студент].Четверки)>0));

**Результат запроса.**

Рисунок 41 – Запрос «Студенты с 50% повышением»

**Формулировка запроса.** Вывести количество студентов, получающих стипендию со 100%-ой надбавкой.

**Запрос «Количество студентов со 100%-ым повышением»**

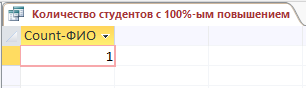
Используется для вспомогательный формы “Количество студентов со 100%-ым повышением” формы “Студент”. Содержит запрос “Студенты со 100%-ным повышением”

**SQL-код запроса.**

SELECT Count([Студенты с 100%-ным повышением].ФИО) AS [Count-ФИО]

FROM [Студенты с 100%-ным повышением];

**Результат запроса.**

Рисунок 42 – Запрос «Количество студентов с 100% повышением»

**Формулировка запроса.** Вывести ФИО студентов, получающих стипендию со 100%-ой надбавкой.

**Запрос «Студенты со 100%-ным повышением»**

Вспомогательный запрос для запроса “Количество студентов cо 100%-ым повышением”. Содержит запрос “Для формы студент”

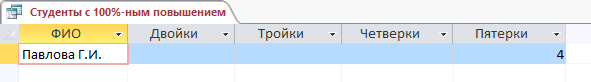
**SQL-код запроса.**

SELECT [Для формы студент].ФИО, [Для формы студент].Двойки, [Для формы студент].Тройки, [Для формы студент].Четверки, [Для формы студент].Пятерки

FROM [Для формы студент]

WHERE ((([Для формы студент].Двойки) Is Null) AND (([Для формы студент].Тройки) Is Null) AND (([Для формы студент].Четверки) Is Null) AND (([Для формы студент].Пятерки)>0));

**Результат запроса.**

Рисунок 43 – Запрос «Студенты с 100%-ным повышением»

**Формулировка запроса.** Вывести количество студентов, не получающих стипендию, получающих обычную стипендию, с 50% и 100%-ыми надбавками.

**Запрос «КоличествоСтудентовСРазнойСтипендией»**

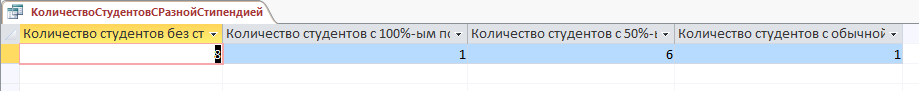
Используется для отчета “КоличествоСтудентовСРазнойСтипендией”. Содержит запросы “Количество студентов без стипендии”, “Количество студентов с обычной стипендией”, “Количество студентов с 50%-ым повышением”, “Количество студентов со 100%-ым повышением”.

**SQL-код запроса.**

SELECT [Количество студентов без стипендии].[Count-ФИО], [Количество студентов с 100%-ым повышением].[Count-ФИО], [Количество студентов с 50%-ым повышением].[Count-ФИО], [Количество студентов с обычной стипендией].[Count-ФИО]

FROM [Количество студентов без стипендии], [Количество студентов с 100%-ым повышением], [Количество студентов с 50%-ым повышением], [Количество студентов с обычной стипендией];

**Результат запроса.**

Рисунок 44 – Запрос «КоличествоСтудентовСРазнойСтипендией»

**Формулировка запроса.** Вывести ФИО студента, номер группы, в которой он учится и факультет, к которому относится группа.

**Запрос «Для отчета количество студентов»**

Используется для отчета “Количество студентов”. Содержит таблицы “Студент” и “Группа”.

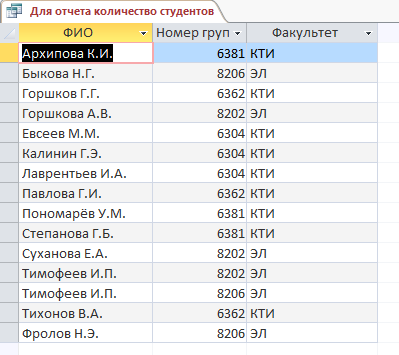
**SQL-код запроса.**

SELECT Студент.ФИО, Группа.Номер\_группы, Группа.Факультет

FROM Группа INNER JOIN Студент ON Группа.Номер\_группы = Студент.Номер\_группы

GROUP BY Студент.ФИО, Группа.Номер\_группы, Группа.Факультет;

**Результат запроса.**

Рисунок 45 Запрос «Для отчета количество студентов»

**Формулировка запроса.** Вывести ФИО студентов, которые получили больше 2-х двоек по экзамену, а также вывести количество не сданных экзаменов.

**Запрос «Для отчета: Приказ на отчисление»**

Используется для отчета “Приказ на отчисление”. Содержит запрос “Для формы студент”.

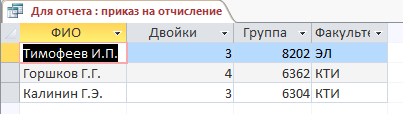
**SQL-код запроса.**

SELECT [Для формы студент].ФИО, [Для формы студент].Двойки

FROM [Для формы студент]

WHERE ((([Для формы студент].Двойки)>2));

**Результат запроса.**

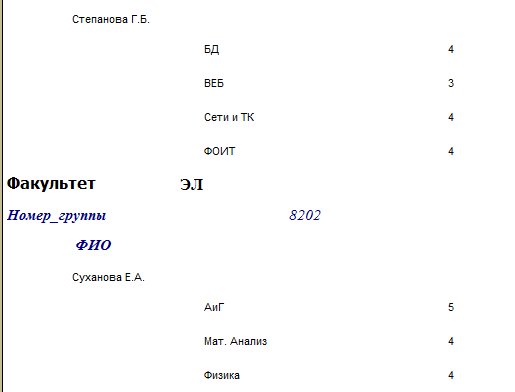
Рисунок 46 – Запрос «Для отчета: Приказ на отчисление»

# Описание отчетов

Отчёт **«Информация о студентах»** состоит из четырех внедрённыхотчётов: **«Количество студентов», «Студенты с различной стипендией», «Средний балл», «Студенты, получающие стипендию».**

Рисунок 47 – Отчет «Студенты, получающие стипендию»

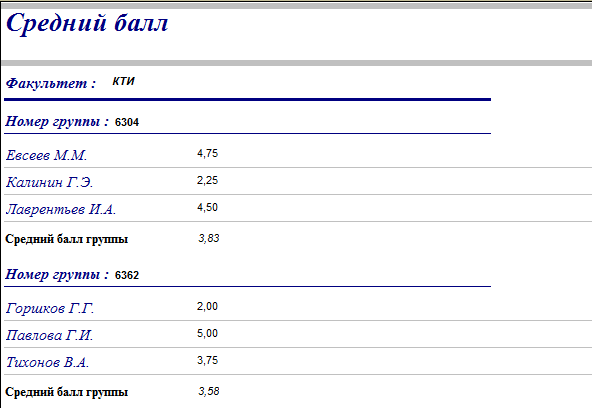
Рисунок 48 – Отчет «Студенты, получающие стипендию»

Рисунок 49 - Отчет «Студенты, получающие стипендию»

**Источник данных -** Для отчета: студенты, получающие стипендию

**Наличие группировки и/или сортировки.** Группировка производится по полям “Факультет” и “Номер группы”, а сортировка по полю “ФИО”.

**Формулы и области расположения вычисляемых полей.** Вычисляемых полей нет.

Рисунок 50 – Отчет «Средний бал»

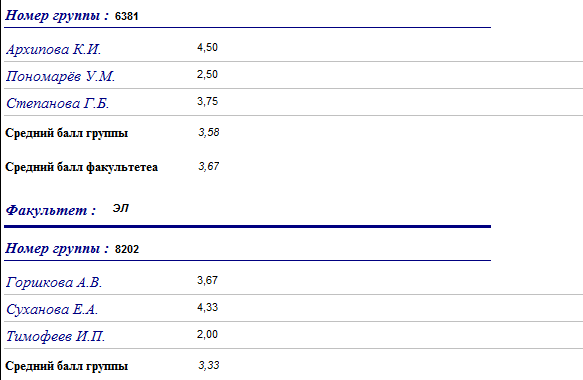
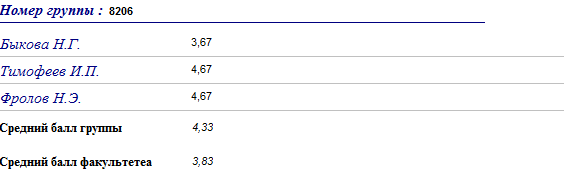
Рисунок 51 – Отчет «Средний бал»

Рисунок 52 – Отчет «Средний бал»

**Источник данных –** связанные таблицы “Студент”, “Группа” и “Сдача\_экзамена”

**Наличие группировки и/или сортировки.** Группировка и сортировка производятся по полям “Факультет”, ”Номер группы” и “ФИО”.

**Формулы и области расположения вычисляемых полей.**

Вычисляемое поле “СреднийБаллСтудента” (Формула : Avg([Оценка\_по\_экзамену])) находится в заголовке группы “ФИО”

Вычисляемое поле “СреднийБаллГруппы” (Формула :  
Avg([Оценка\_по\_экзамену]) находится в примечании группы “Номер группы”

Вычисляемое поле “СреднийБаллФакультета” (Формула : Avg([Оценка\_по\_экзамену])) находится в примечании группы “Факультет”

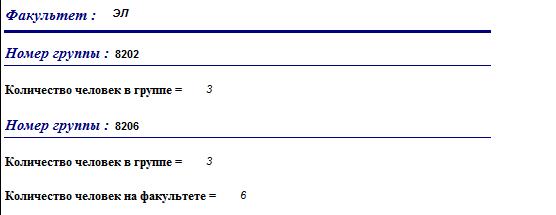
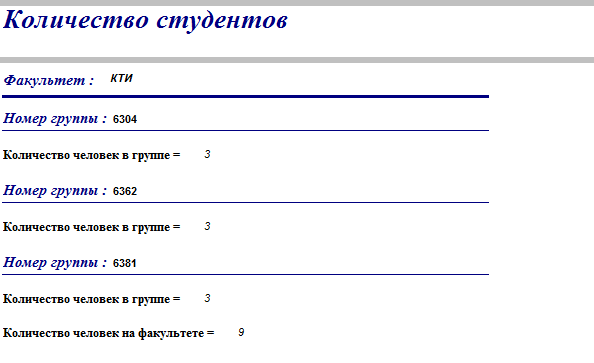
Рисунок 53 – Отчет «Количество студентов»

Рисунок 54 –Отчет «Количество студентов»

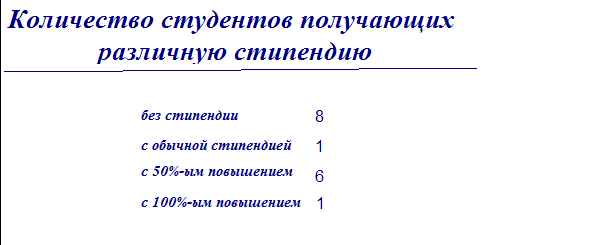
**Источник данных** – запрос “Для отчета количество студентов”

**Наличие группировки и/или сортировки.** Группировка производится по полям “Факультет” и “Номер группы”, сортировка – по полю “Номер группы”.

**Формулы и области расположения вычисляемых полей.**

Вычисляемое поле “СчетГруппы” (Формула : Count([ФИО])) находится в примечании группы “Номер\_группы”.

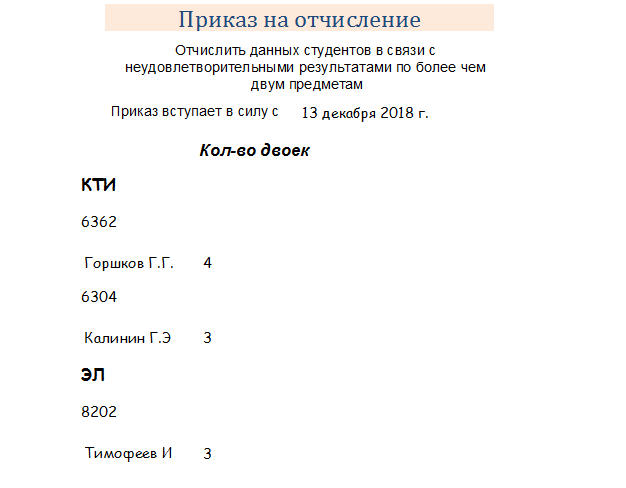
Вычисляемое поле “СчетФакультета” (Формула : Count([ФИО])) находится в примечании группы “Факультет”.

Рисунок 55 – Отчет «КоличествоСтудентовСРазнойСтипендией»

**Источник данных –** запрос “КоличествоСтудентовСРазнойСтипендией”

**Наличие группировки и/или сортировки.** Группировки и сортировки нет.

**Формулы и области расположения вычисляемых полей.** Вычисляемых полей нет.

Рисунок 56 – Отчет «Приказ на отчисление»

**Источник данных –** запрос “Для отчета: приказ на отчисление”

**Наличие группировки и/или сортировки.** Группировки нет, сортировка – по полю ФИО.

**Формулы и области расположения вычисляемых полей.** Вычисляемых полей нет.

**Выводы**

В результате выполнения данного индивидуального задания была создана программная система, предназначенная для работника деканата. Данная система обеспечивает хранение сведений о группах и студентах, а также о результатах текущей сессии.

Работник деканата может вносить в БД следующие изменения:

* Удалить или добавить в базу студента;
* Поменять студенту номер группы, специальность, кафедру, номер зачетки;
* Занести оценки, полученные студентами на экзаменах по каждому предмету;
* По результатам сессии начислить стипендии студентам

Работник деканата может получить следующие сведения:

* Студенты, обучающиеся на определенной кафедре и не сдавшие хотя бы один экзамен, с указанием группы и предмета, по которому оценка отсутствует или равна 2
* Средний балл студентов каждой группы указанного факультета
* Средний балл по каждому предмету
* Список студентов указанной кафедры, которые по итогам сессии могут получать стипендию
* Список студентов, подлежащих отчислению (не сдано более двух предметов) по всему факультету.
* Количество студентов факультета, не получающих стипендию, получающих обычную стипендию, с 50 % надбавкой и со 100 % надбавкой.

Предусмотрена возможность получения отчета о студентах каждой группы факультета, получающих стипендию, с указанием оценок, полученных ими, и подсчетом среднего балла каждого студента, среднего балла группы и факультета. Также подсчитывается в группе и на факультете общее количество студентов, количество студентов, получающих обычную стипендию, получающих повышенную стипендию и не получающих стипендию.

Возможно получение приказа на отчисление студентов, не сдавших более двух предметов.