

Чтобы создать связь надо привести указатель мыши на поле, нажать левую кнопку мыши и, удерживая её нажатой, протащить на соответствующее поле другой таблицы. Появится окно **Изменение связей**. Восстановите связь между таблицами «Группы» и «Студенты» (рис.30).

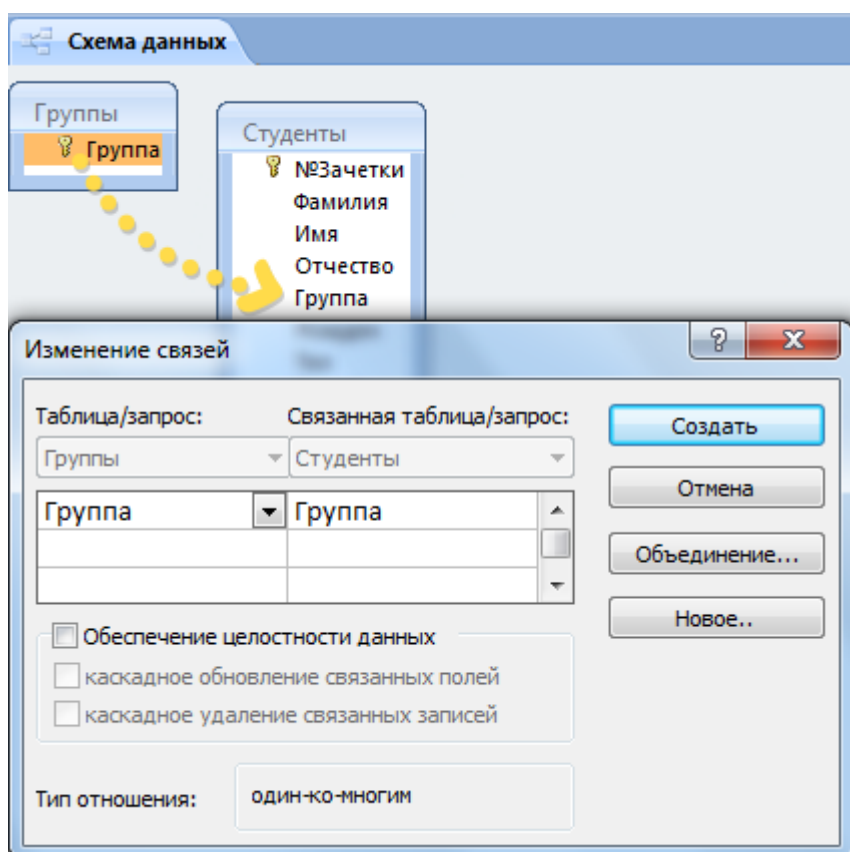


Рисунок 30. Создание связи по полю «Группа» между таблицами «Группы» и «Студенты».

В окне **Изменение связей** можно задавать **Обеспечение целостности** или отказываться от него, Кроме того, можно включать или отключать такие дополнительные возможности, как **Каскадное обновление связанных полей** и **Каскадное удаление связанных записей**.

Вопросы по теме «Схема данных»

1. Что такое **Схема данных**? Для чего она используется? Что такое **список полей** в схеме данных? Как связываются таблицы? Можно ли связывать таблицы уже заполненные значениями?
2. Что такое **межтабличная связь**? Какие существуют **типы** межтабличных связей? Что такое **линия связи** в схеме данных? Каковы требования к связываемым полям?
3. Что такое **первичный ключ** и **внешний ключ**?
4. Что такое **главная** и **подчиненная** таблица? Может ли главная таблица одновременно быть и подчиненной таблицей? Какова очередность заполнения главной и подчиненной таблиц? Чем это вызвано?

Целостность данных

Обеспечение целостности данных нам нужно для того, чтобы одна и та же информация в связанных таблицах была представлена одинаково. Например, человек может при вводе ошибиться и ввести в одном случае код группы **1721**, а в другом код группы **1721_**. Это разные коды с точки зрения Access. Обеспечение целостности страхует нас от таких ошибок.

Зададим **Обеспечение целостности**. Для этого надо выделить линию связи, например, между полем «Группа» таблицы «Группы» и полем «Группа» таблицы «Студенты», и правой кнопкой мыши вызвать контекстное меню, в котором выбрать "Изменить связь". Появится окно «Изменение связей». Поставим галочку для обеспечения целостности (рис. 31).

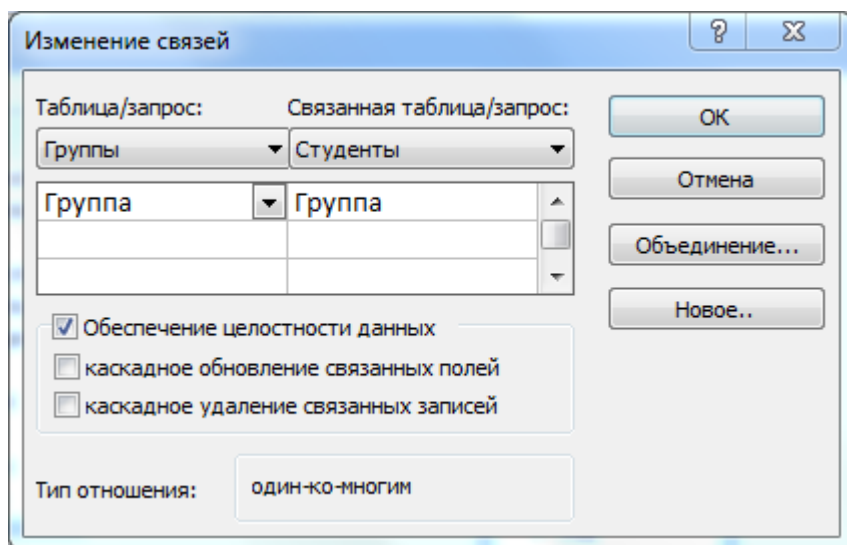


Рисунок 31. Окно «Изменение связей». Включено Обеспечение целостности.

После обеспечения целостности для всех связей получим такую схему данных (рис. 32).

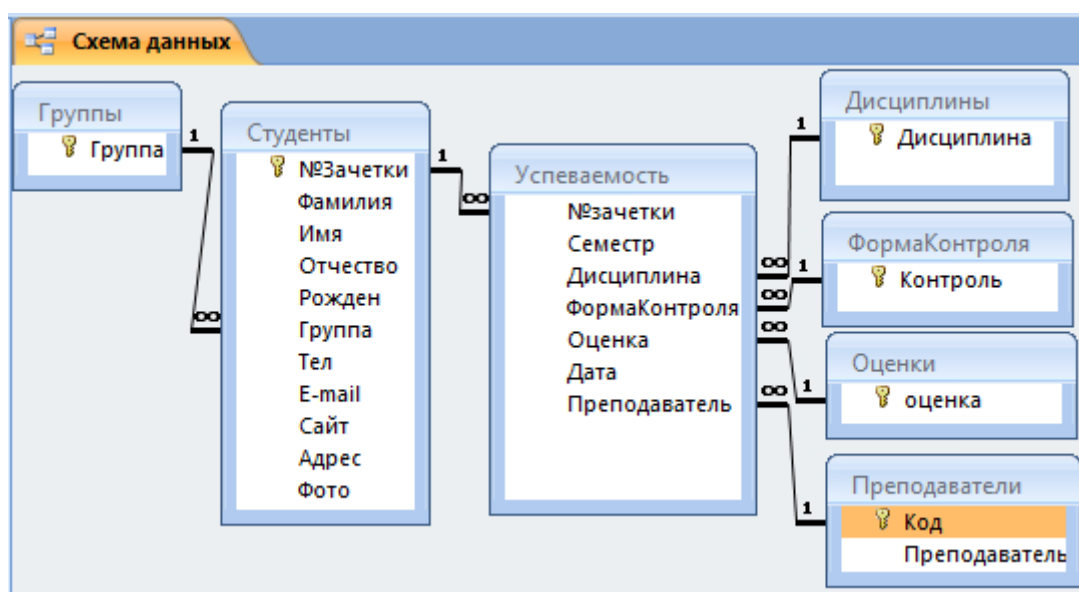


Рисунок 32. Схема данных с обеспечением целостности.

У концов линий связи появятся знаки "1" и "∞". Таблицы на стороне "1" называются **главными**, а на стороне "∞" – **подчиненными** или связанными. Например, таблица «Группы» – главная, а таблица «Студенты» – подчиненная. **Тип связи** между ними "**один-ко-многим**". Это обозначает, что в одной группе может быть много студентов, а может и ни одного. Бывают еще типы связи "**один-к-одному**" и "**многие-ко-многим**". Например, таблица «Успеваемость» обеспечивает связь "многие-ко-многим". Один студент сдает экзамены по многим дисциплинам, одну дисциплину сдают многие студенты.

Правила целостности

Правила целостности гарантируют нам, что **первичный ключ** (например, поле «Группа» таблицы «Группы») не будет содержать повторяющихся значений, а **внешний ключ** (в нашем примере поле

«Группа» таблицы «Студенты») будет содержать только те значения, которые ранее были введены в соответствующий первичный ключ.

Самостоятельно проверьте, как работает **Обеспечение целостности**. Для этого, надо открыть какую-нибудь таблицу, например «Группы», и попробовать изменить или удалить код какой-нибудь группы. Если у этой группы есть подчиненные записи в таблице Студенты, то Access не позволит нам это сделать. Действительно, как же можно удалять группу, если в ней есть студенты. А вот если студентов в группе нет, тогда можно и удалить код группы, и изменить его. А теперь откроем подчиненную таблицу «Студенты» и попробуем изменить код группы у какого-либо студента, если такой группы нет в главной таблице Группы, то Access не позволит нам это сделать, в то же время перевести студента из одной группы в другую можно. Поэкспериментируйте.

Каскадное обновление связанных полей

Задайте у линии связи таблицы Группы **Каскадное обновление связанных полей** (рис. 33)

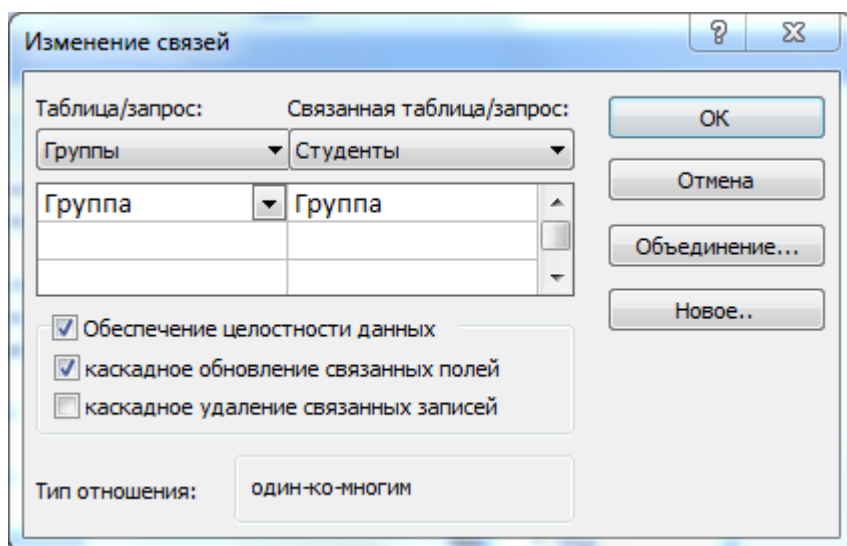


Рисунок 33. Каскадное обновление связанных полей.

А теперь откройте таблицу «Группы» и измените код какой-нибудь группы. Access позволит нам это сделать, при этом код группы в таблице «Студенты» также изменится.

Каскадное удаление связанных записей

Задайте у линии связи таблицы «Группы» **Каскадное удаление связанных записей** (рис. 34).

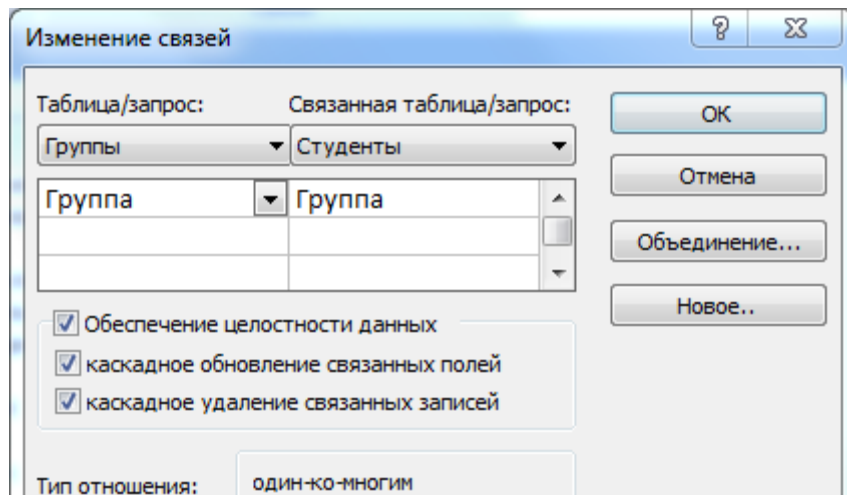


Рисунок 34. Каскадное удаление связанных записей.

Если вы откроете таблицу «Группы» и попытаетесь удалить запись с кодом какой-либо группы, то Access позволит нам это сделать, но при этом будут удалены все студенты этой группы, так что пользоваться этой возможностью надо осторожно.

Вопросы по теме «Целостность данных»

1. Что такое **межтабличная связь**? Какие существуют **типы** межтабличных связей? Что такое **линия связи** в схеме данных? Каковы требования к связываемым полям?
2. Что такое **первичный ключ** и **внешний ключ**?
3. Что такое **главная** и **подчиненная** таблица? Может ли главная таблица одновременно быть и подчиненной таблицей? Какова очередность заполнения главной и подчиненной таблиц? Чем это вызвано?
4. Что такое **целостность данных**? Для чего она используется? Проанализируйте схему данных. Как конкретно обеспечивается целостность данных для связанных таблиц?
5. Что такое **Каскадное обновление связанных полей**?
6. Что такое **Каскадное удаление связанных записей**?
7. Почему надо тщательно продумать и создать схему базы данных до ввода данных в таблицы?

Запросы

Запросы используются для выборки данных из таблиц или других запросов, для изменения данных, для вычислений. В меню "Создание" представлены два основных способа создания запросов: **Мастер запросов** и **Конструктор запросов**. При создании запроса появится меню Конструктор, в котором можно указать **тип запроса**. Существует несколько типов запросов: "Выборка" (тип запроса по умолчанию), "Создание таблицы", "Добавление", "Обновление", "Удаление" и др. Есть несколько режимов просмотра запроса, основными режимами являются "Режим таблицы" и "Конструктор".

Запрос на выборку

С помощью Мастера запросов (Простой запрос) создайте **запрос на выборку**. Включите в запрос две таблицы «Группы» и «Студенты», выберите в запрос поля Группа, №Зачетки, Фамилия, Имя (рис. 35). Сохраните запрос с именем "Список студентов". Проверьте запрос в работе.

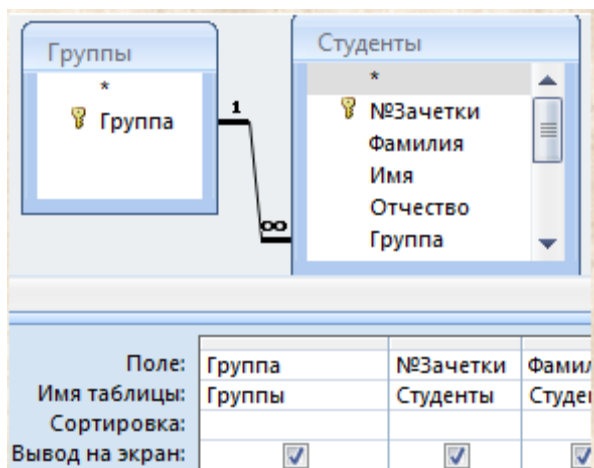


Рисунок 35. Запрос на выборку.

Запрос с параметром

С помощью Конструктора запросов создайте **запрос с параметром** на выборку студентов указываемой группы. Включите в запрос две таблицы «Группы» и «Студенты», выберите в запрос

поля Группа, №Зачетки, Фамилия, Имя. В **условии отбора** по полю «Группа» введите подходящий текст в квадратных скобках, например, **[Введите группу]** (рис. 36). Проверьте запрос в работе.

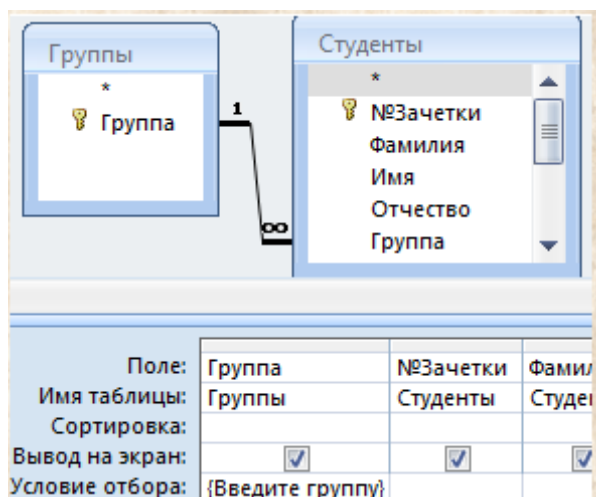


Рисунок 36. Запрос с параметром.

Запрос «без подчиненных»

С помощью Мастера запросов (**запрос "без подчиненных"**) узнайте, в каких группах нет студентов. У нас в группе с кодом 9999 нет студентов. Выполните запрос, он покажет группу, в которой нет студентов. Обратите внимание, что по полю «Группа» в **условии отбора** Мастер запросов поставил выражение **Is Null** (рис. 37).

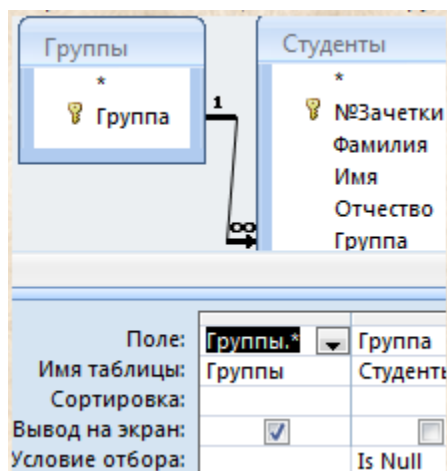


Рисунок 37. Запрос без подчиненных.

Запрос «повторяющиеся записи»

С помощью Мастера запросов (**запрос "повторяющиеся записи"**) узнайте, есть ли повторяющиеся записи в таблице "Успеваемость". Для этого выберите в запрос все поля из таблицы "Успеваемость". Если у нас нет повторяющихся записей в таблице "Успеваемость", то запрос покажет их отсутствие. Для проверки работы запроса введите в таблицу запись, которая точно повторяет какую-либо запись. Обязательно должны быть заполнены поля «№зачетки» и «Преподаватель», другие поля могут быть и не заполнены. После этого повторите запрос, повторяющаяся запись будет найдена и показана. Это ошибочная запись, так как нет необходимости дважды фиксировать одни и те же результаты, это может нарушить отчетность. **Самостоятельно** ознакомьтесь со структурой запроса в конструкторе, он достаточно громоздкий.

Запрос на основе нескольких таблиц

С помощью Мастера запросов (**простой запрос**) получите данные об успеваемости студентов на основе трех таблиц: "Группы", "Студенты", "Успеваемость". Включите в запрос поля: Группа, №Зачетки, Фамилия, Имя, Семестр, Дисциплина, ФормаКонтроля, Оценка, Дата, Преподаватель. Дайте запросу имя "**Успеваемость Студентов**" (рис. 38).

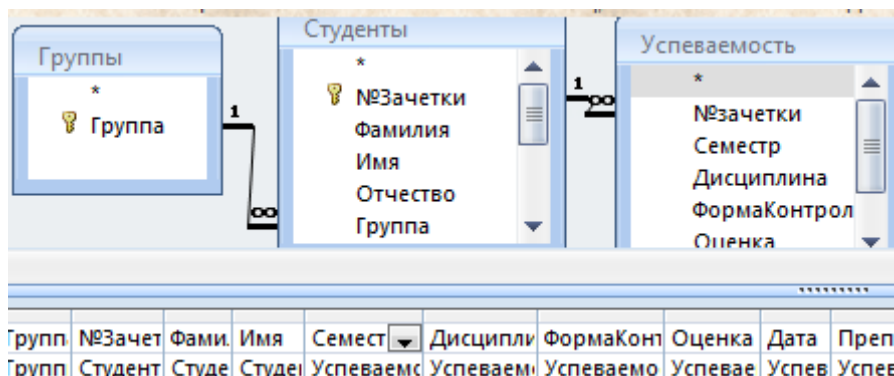


Рисунок 38. Запрос на основе нескольких таблиц.

Выполните запрос "**Успеваемость Студентов**".

Этот запрос нами подготовлен для следующего задания по созданию перекрестного запроса, так как перекрестный запрос не может быть построен на основе нескольких таблиц, а только на основе одной таблицы или одного запроса.

Перекрестный запрос

С помощью Мастера запросов (**перекрестный запрос**), на основе предварительно специально нами подготовленного запроса "Успеваемость студентов", постройте перекрестную таблицу со сведениями об успеваемости студентов по учебным дисциплинам. Этот запрос должен показать нам в качестве заголовков строк - Дисциплины, в качестве заголовков столбцов - Фамилии, а на пересечении строк и столбцов - оценки. В конструкторе наш перекрестный запрос будет выглядеть примерно так (рис. 39).

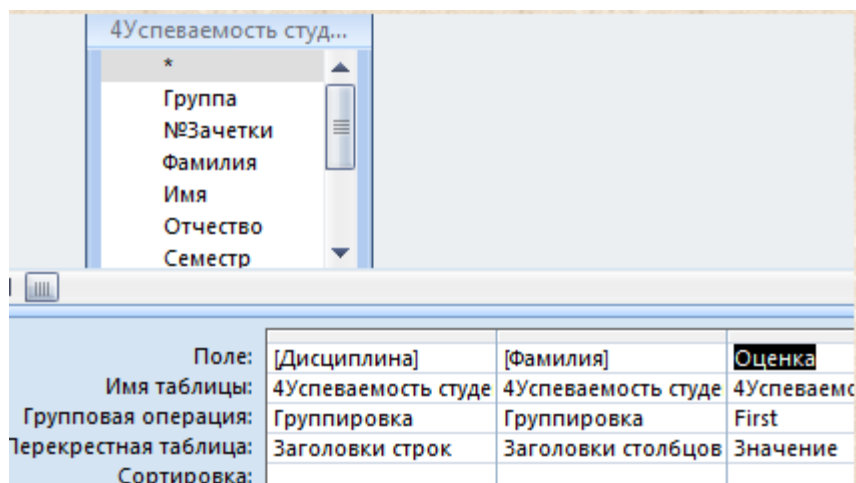


Рисунок 39. Перекрестный запрос в Конструкторе.

Выполните запрос. Должно получиться примерно так (рис. 40).

Дисциплина	Иванов	Петров
Иностранный язык в сфер		
Информатика		
История госуд. и права за		
История отеч. государства		
ИТ в юридической деятел	зачтено	зачтено

Рисунок 40. Перекрестный запрос.

Вопросы по теме «Запросы»

1. Что такое **запрос на выборку**? Способы создания запросов.
2. **Типы запросов**: запрос на выборку, запрос на обновление, запрос на добавление, запрос на удаление, запрос на создание таблицы, перекрестный запрос.
3. **Режимы работы с запросом**: Режим таблицы, Сводная таблица, Сводная диаграмма, Режим SQL, Конструктор.
4. **Бланк запроса**? Для чего он используется? Можно ли изменять межтабличные связи в бланке запроса?
5. **Источник данных для запроса**? Позволяет ли запрос использовать в качестве источника данных другие запросы, несколько таблиц или запросов?
6. Что такое **условие отбора** для поля в запросе?
7. Что такое **запрос с параметром**?
8. Что такое **перекрестный запрос**?
9. Какую информацию позволяет получить "**Запрос без подчиненных**"?
10. Какую информацию позволяет получить "**Запрос Повторяющиеся записи**"?

Вычисления в запросах. Построитель выражений.

Запрос с группировкой

Создадим в Конструкторе **запрос с группировкой** для подсчета количества студентов в группах. Включим в запрос таблицы «Группы» и «Студенты». Из них выберем два поля «Группа» и «№Зачетки». Далее надо показать в запросе строку **Групповая операция**, которая обычно скрыта. Для этого можно в меню Конструктор нажать кнопку **Итоги**. В строке Групповая операция для поля «Группа» задать функцию **Группировка**, а для поля «№Зачетки» - функцию **Count**, что обозначает количество. Таким способом можно производить группировку и подсчет количества записей в запросах (рис. 41).

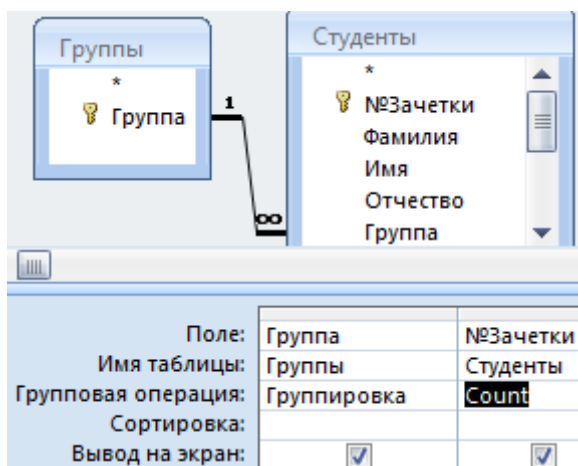


Рисунок 41. Запрос с группировкой.

Выполните запрос. Результат на рис. 42.

Группа	Count-№Зач
1721	4
1761	1

Рисунок 42. Результат выполнения запроса с группировкой.

Запрос с условием отбора по полю с типом данных Дата

Создадим запрос с условием отбора студентов, родившихся в некоем промежутке дат (рис. 43).

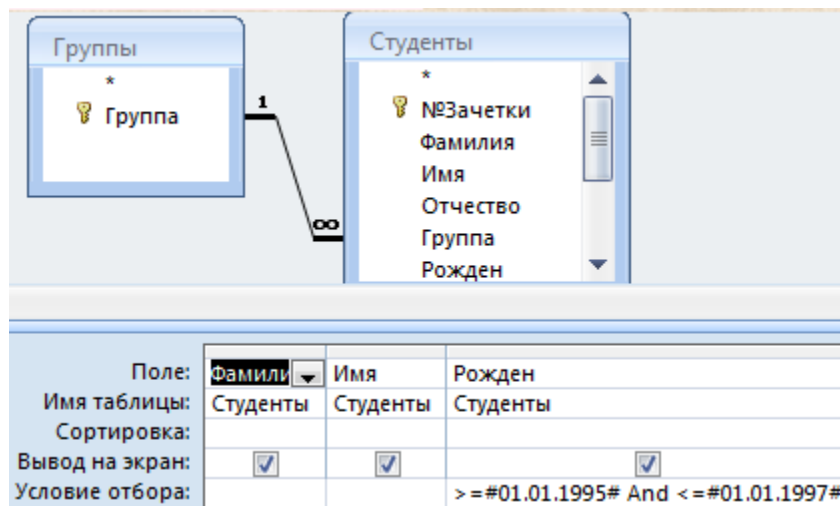


Рисунок 43. Запрос с условием отбора по полю с типом данных Дата.

В условии отбора для поля «Рожден» записано выражение:

`>=#01.01.1995# And <=#01.01.1997#`

Обратите внимание, что даты взяты в символы #.

Запрос с вычисляемым полем

Создадим в Конструкторе **запрос с вычисляемым полем**, в котором отображается Фамилия и инициалы (рис. 42).

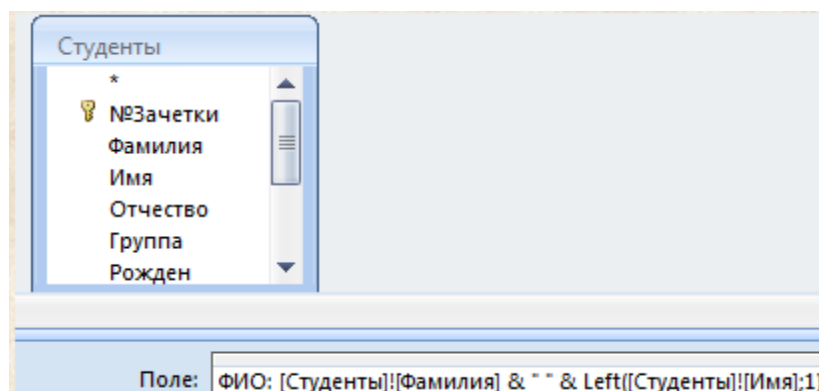


Рисунок 44. Запрос с вычисляемым полем.

Добавим в запрос таблицу «Студенты». В поле введем выражение:

`ФИО: [Студенты]![Фамилия] & " " & Left([Студенты]![Имя];1)`

Поясним, что обозначает это выражение.

ФИО - это название вычисляемого поля, оно заканчивается двоеточием.

[Студенты]![Фамилия] - это поле Фамилия из таблицы Студенты.

& - это символ конкатенации, он объединяет текстовые строки.

" " - это пробел в двойных кавычках.

Left([Студенты]![Имя];1) - это встроенная функция Left.

У функции два аргумента: первый – поле Имя из таблицы Студенты, второй – число 1, которое указывает, сколько символов взять из имени, считая слева. То есть эта функция вернет нам первую букву имени.

Построитель выражений

Для записи таких выражений лучше всего использовать **Построитель выражений**. Для вызова Построителя можно щелкнуть правой кнопкой мыши по вычисляемому полю в запросе и выполнить команду **Построить**. Появится окно Построитель выражений (рис. 45).

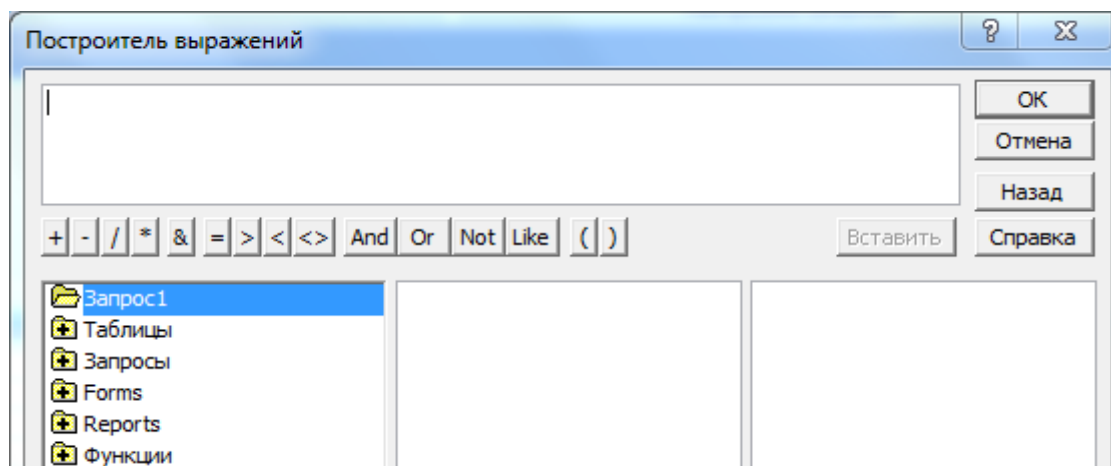


Рисунок 45. Построитель выражений.

Раскройте таблицы по знаку плюс, выберите таблицу «Студенты», выберите поле «Фамилия» и нажмите кнопку «Вставить» (рис. 46).

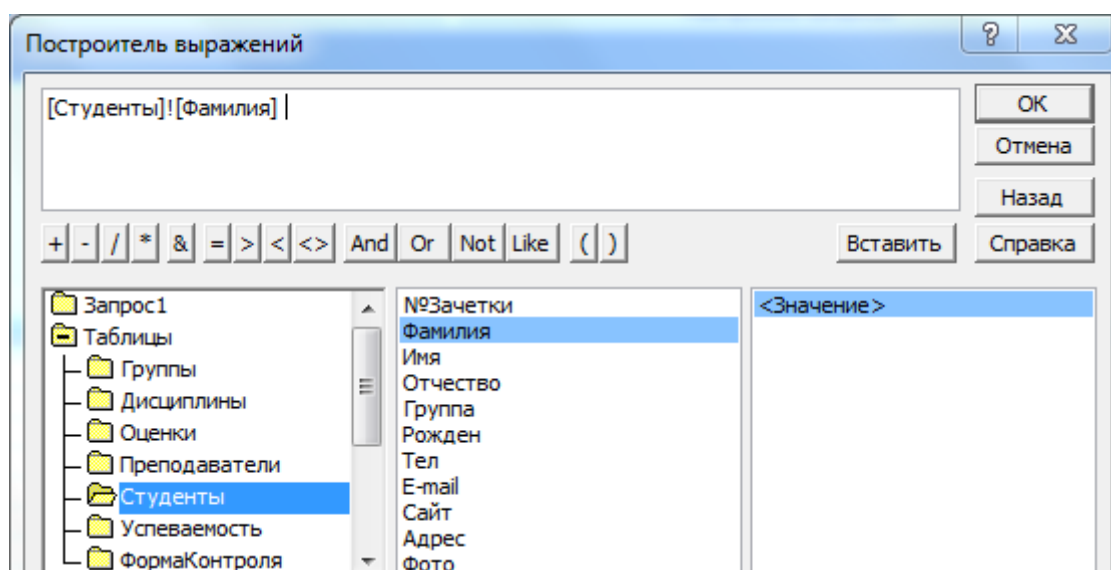


Рисунок 46. Построение выражения [Студенты]![Фамилия].

Введите знак конкатенации **&**, нажав соответствующую кнопку на панели инструментов Построителя выражений, затем введите вручную пробел в двойных кавычках " ", введите ещё один знак **&** (рис. 47).

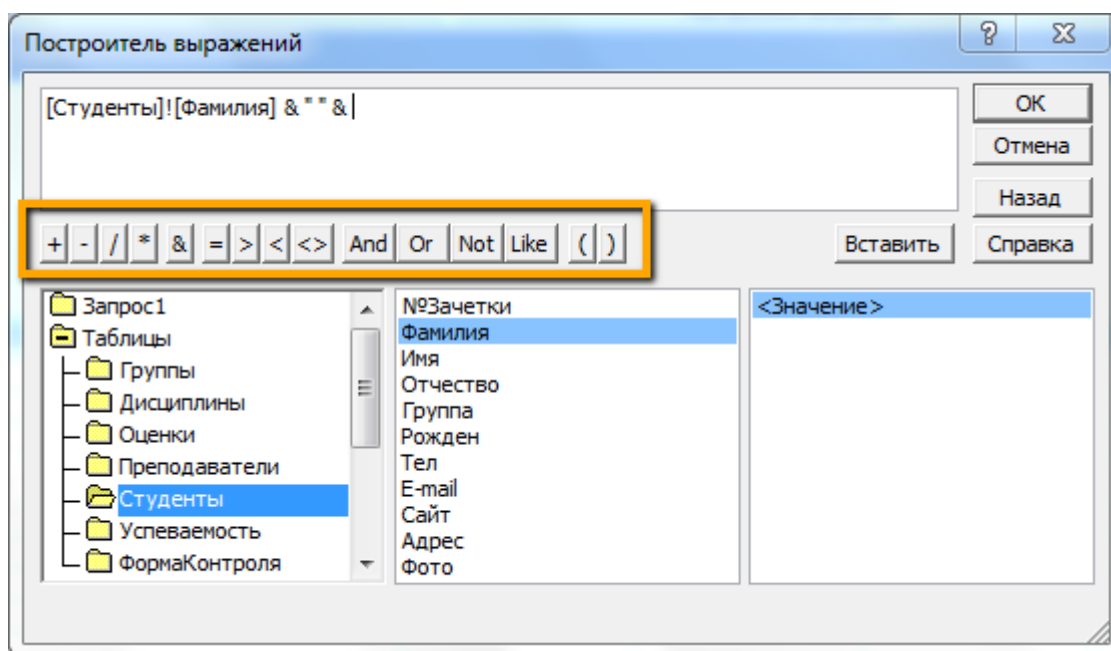


Рисунок 47. Панель инструментов Построителя выражений. Добавление пробела & " " &

Выберите «Встроенные функции», затем категорию «Текстовые», затем функцию Left. Нажмите кнопку «Вставить» (рис. 48).

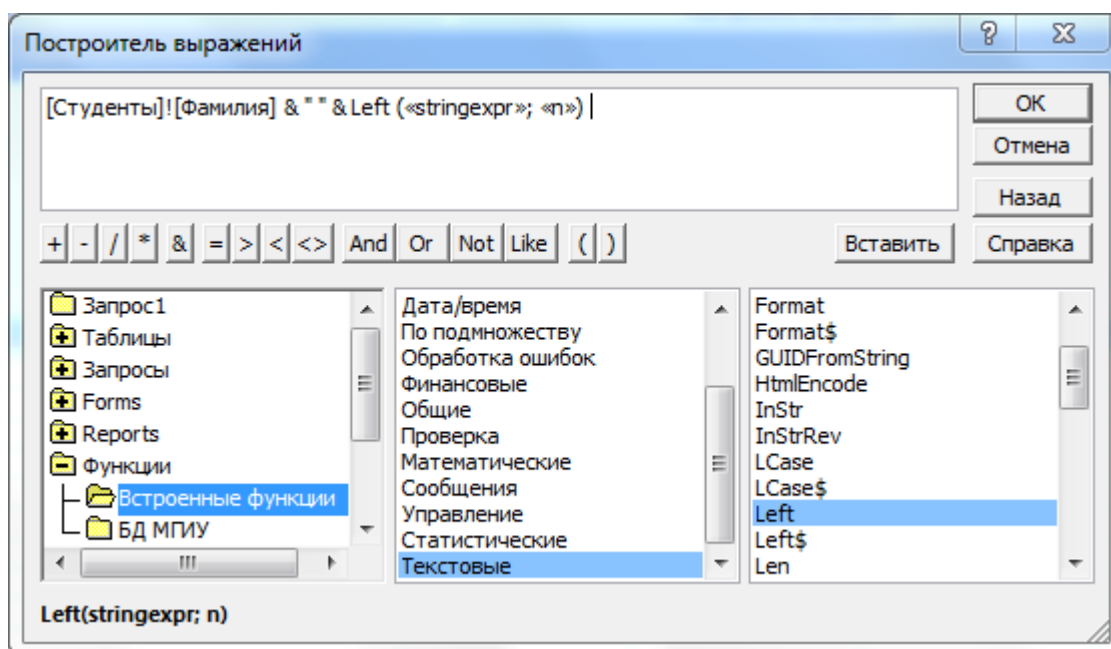


Рисунок 48. Вставка функции Left в Построителе выражений.

У функции Left заменим первый аргумент **«stringexpr»** на выражение **[Студенты]![Имя]**, для этого выделим первый аргумент, затем выберем таблицу «Студенты, выберем поле «Имя» и нажмем кнопку «Вставить» (рис. 49).

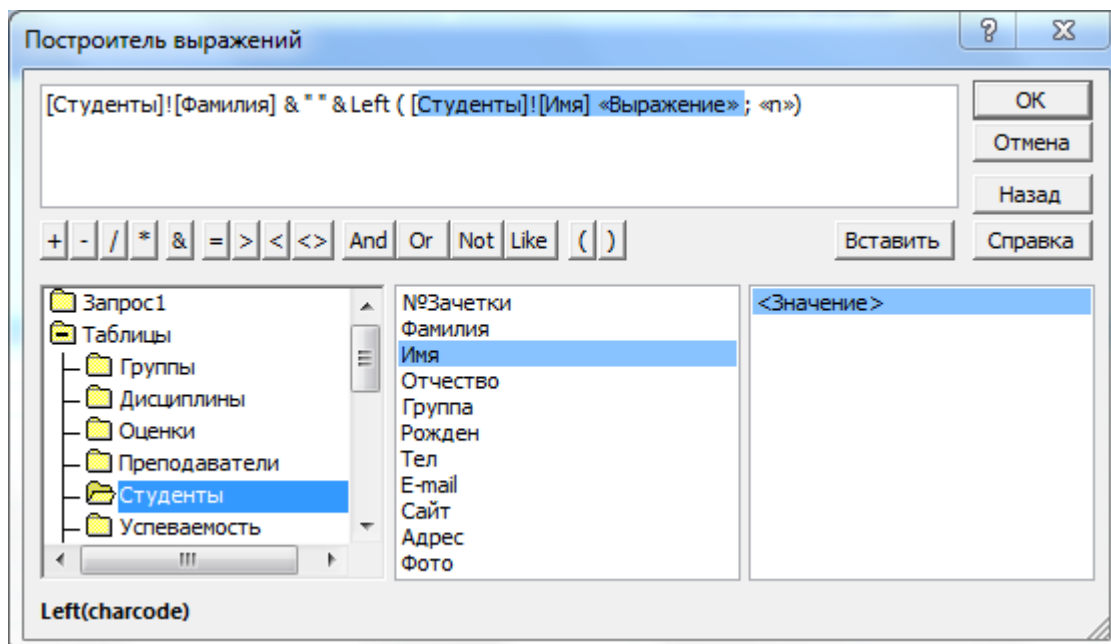


Рисунок 49. Замена первого аргумента функции Left на выражение [Студенты]![Имя].

Вручную удалите текст «Выражение», а вместо второго аргумента «п» введите цифру 1. Окончательный вид построенного выражения на рис. 50.

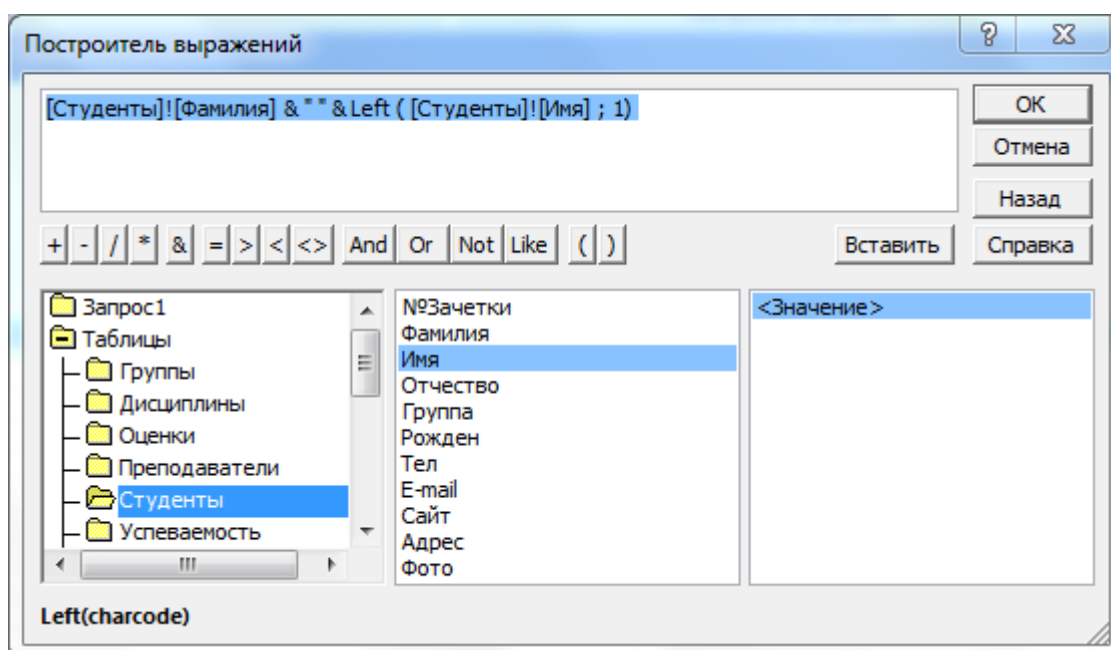


Рисунок 50. Окончательный вид построенного выражения в Построителе выражений.

Нажмите кнопку ОК. Построенное выражение появится в вычисляемом поле в виде:

Выражение1: [Студенты]![Фамилия] & " " & Left([Студенты]![Имя];1)

Остается только заменить текст «Выражение1» на «ФИО». Запрос с вычисляемым полем построен. Проверьте его в работе.

Вопросы по теме «Вычисления в запросах»

1. Что такое **условие отбора** для поля? Выражение в условии отбора. Условия отбора с логическими операторами **And** и **Or**?
2. Что такое **вычисляемое поле** в запросе?
3. Что такое **выражение**?
4. Что такое **Построитель выражений**? Как им пользоваться?
5. Как провести **сортировку записей** в запросе?
6. Как провести **группировку записей** в запросе и подсчитать количество записей?

Формы

Формы предназначены для удобной работы с базой данных. Формы можно подразделить на формы ввода, изменения и просмотра данных и формы управляющие. Формы создаются на основе таблиц и запросов. В меню Создание представлены различные способы создания форм. Рассмотрим создание форм с помощью Мастера форм и с помощью Конструктора.

Форма «Студенты»

С помощью Мастера на основе таблицы Студенты создайте форму для ввода персональных данных о студенте. Отформатируйте форму по своему вкусу. Добавьте в форму фото. Дайте форме имя «Студенты» (рис. 51).

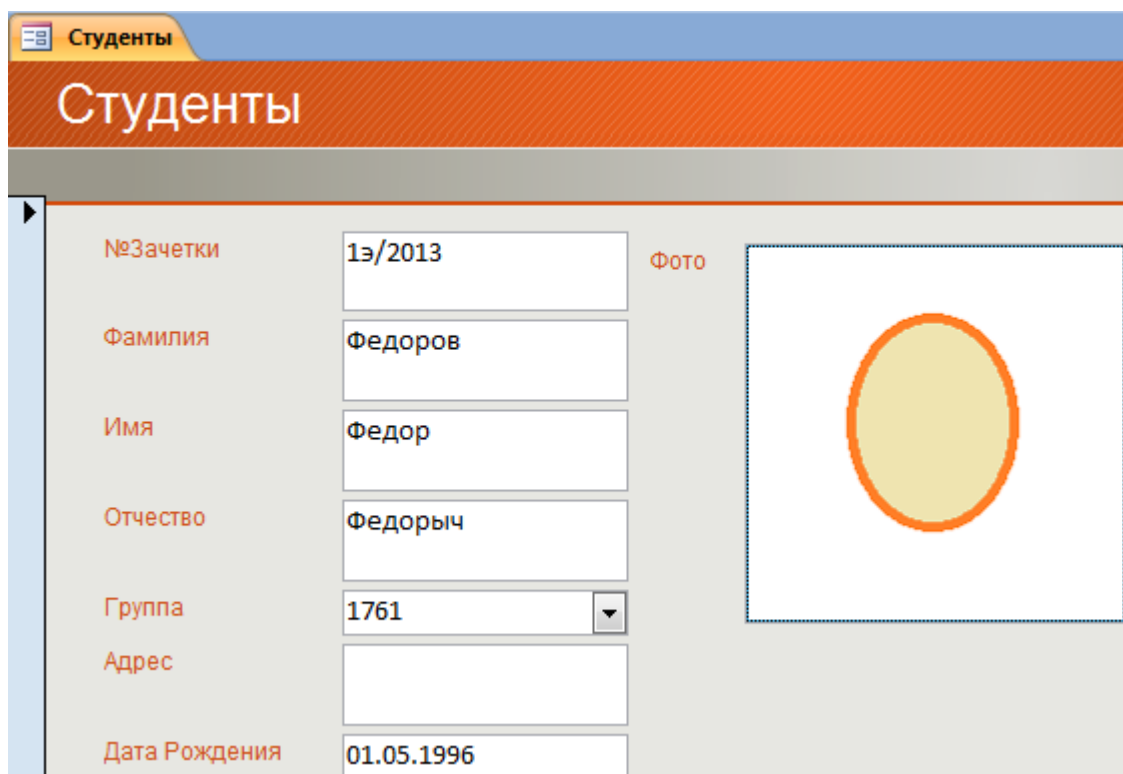


Рисунок 51. Форма «Студенты».

Форма «Успеваемость»

Создадим форму для учета успеваемости студентов. С помощью **Мастера форм** создадим форму с подчиненными формами на основе трех таблиц: "Группы", "Студенты", "Успеваемость", выбирая из этих таблиц необходимые поля (рис. 52). Дайте форме имя «Успеваемость».

Успеваемость

Успеваемость студентов Персональные данные студентов

Группа: 1721

Студенты

№Зачетки	Фамилия	Имя	Отчество
1ю/2013	Иванов	Иван	Иванович
2ю/2013	Петров	Петр	Петрович
*			

Записи: 1 из 2

Успеваемость

Семестр	Дисциплина	Форма	Оценка	Дата	Преподаватель
1	Иностранный язык	Зачет			
1	Логика	Зачет		26.12.2013	Безгинов А.Н.
2	Правоохранительные органы	Экзамен	2		Качаун Е.Я.
1	Русский язык и литература	Зачет	незачтено		
1	Теория государства и права	Экзамен	неявка		Качаун Е.Я.

Записи: 1 из 14

Рисунок 52. Форма «Успеваемость».

Обратите внимание, что будет создано три формы: одна главная и две подчиненных, а отображаться будет объединенная форма. Отформатируйте её вначале в режиме макета, а затем в режиме конструктора.

Опробуйте работу с формой: просмотрите информацию по группе, по студенту, добавьте новую группу, нового студента, добавьте данные о своей успеваемости.

Элементы управления в форме

В форму можно вставлять **элементы управления** для удобства работы. На рисунке 52 в форме «Успеваемость» есть кнопка "Персональные данные студентов". Она создана для перехода в форму «Студенты». Создадим и мы такую кнопку.

Откройте форму «Успеваемость» в Конструкторе. Проверьте, что в меню Конструктор нажата кнопка "Использовать мастера", она поможет нам в создании кнопки (рис. 53).

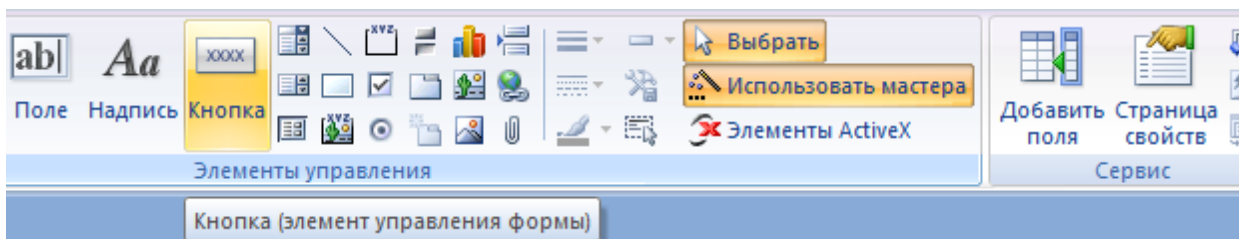


Рисунок 53. Элементы управления в меню Конструктор.

Нажмите элемент **Кнопка** и проведите мышкой там, где планируете создать кнопку. Откроется **Мастер создания кнопок**, которому надо указать, что мы хотим открыть форму «Студенты». Форма «Успеваемость» в Конструкторе представлена на рис. 54.

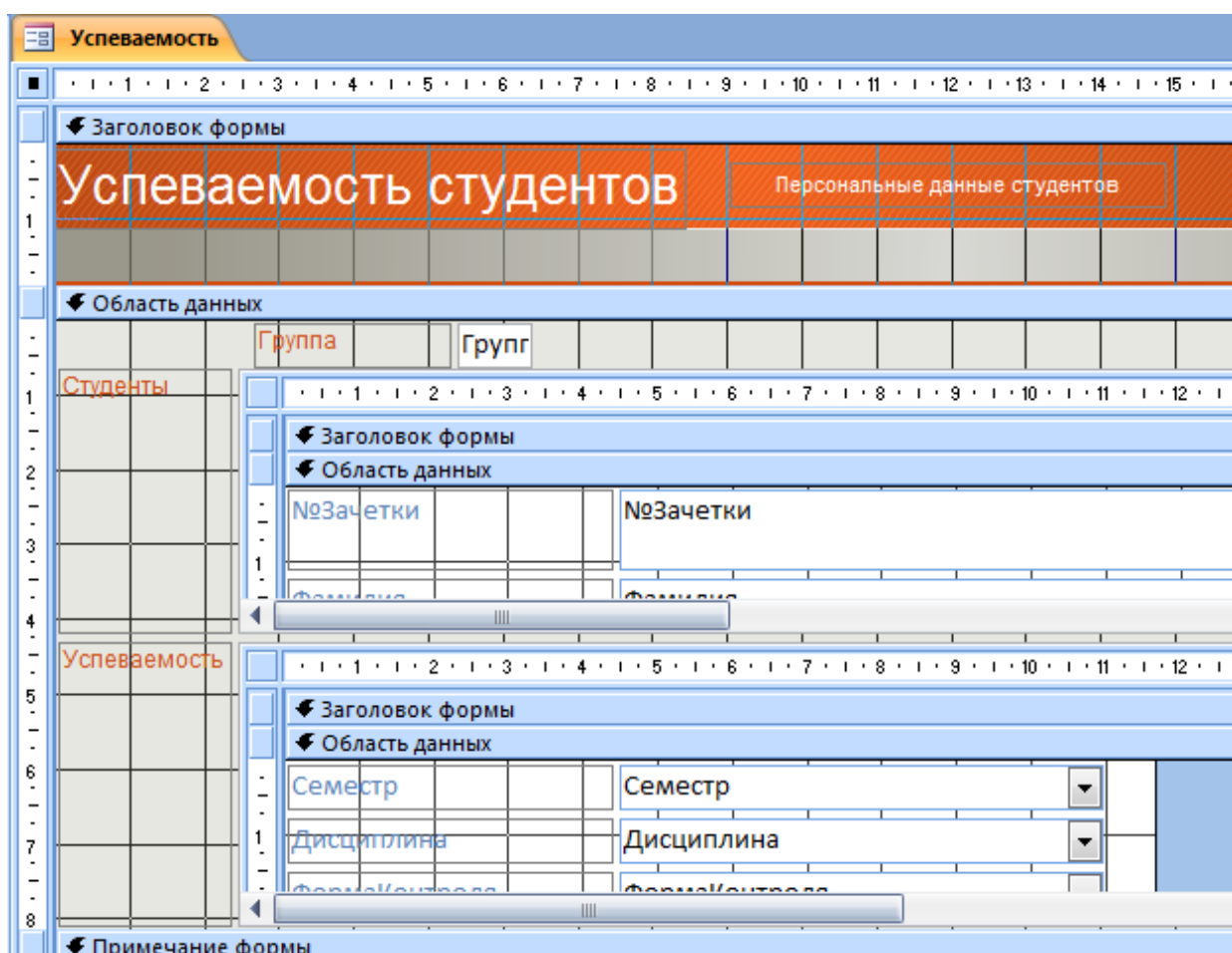


Рисунок 54. Форма «Успеваемость» в Конструкторе.

Обратите внимание на структуру формы в режиме конструктора. Форма состоит из **разделов**: **Заголовок** и **Примечание формы**, **Колонтитулы** страницы, **Область данных**. В разделах Заголовок и Примечание формы отображается общая информация для всей формы, в колонтитулах страницы отображается информация для страницы (если форма состоит из страниц), в разделе Область данных отображаются собственно данные.

Перейдите в режим формы и опробуйте кнопку "Персональные данные студентов".

Вопросы по теме «Формы»

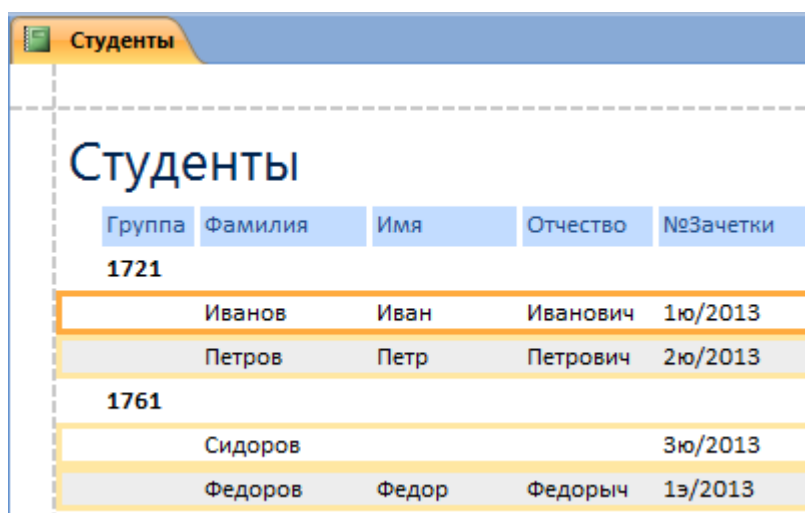
1. Что такое **форма ввода-вывода**? Для чего используются формы? Преимущества использования форм по сравнению с таблицами?
2. **Способы создания форм**: Мастер форм, Конструктор форм?
3. **Режимы работы с формами**: Режим формы, Режим макета, Режим Конструктор?
4. **Разделы формы**: Заголовок и примечание, Колонтитулы, Область данных.
5. **Элементы управления** в форме.
6. **Страница свойств** формы и её элементов. Свойство **Источник записей** формы?
7. **Список полей** формы?
8. Что такое подчиненная форма?

Отчеты

Отчеты – объекты базы данных, предназначенные для создания печатных документов. Отчеты создаются на основе таблиц и запросов. Основными способами создания отчетов являются Мастер отчетов и Конструктор отчетов. Основными режимами просмотра отчета являются: Преставление отчета, Режим макета, Конструктор, Предварительный просмотр.

Отчет Студенты

С помощью Мастера отчетов создайте отчет на основе таблицы "Студенты", включив в отчет поля №зачетки, Фамилия, Имя, Отчество, Группа. Добавьте уровень группировки по полю «Группа». Задайте сортировку по фамилии, имени, отчеству. Отчет будет примерно как на рис. 55.



Группа	Фамилия	Имя	Отчество	№Зачетки
1721	Иванов	Иван	Иванович	1ю/2013
	Петров	Петр	Петрович	2ю/2013
1761	Сидоров			3ю/2013
	Федоров	Федор	Федорыч	1э/2013

Рисунок 55. Отчет «Студенты», сделанный с помощью мастера.

Наша задача – преобразовать отчет, чтобы была нумерация студентов внутри групп, было подсчитано количество студентов в группах и всего. Отчет в режиме просмотра должен выглядеть примерно как на рис. 56.



№	№Зачетки:	Фамилия:	Имя:	Отчество:
1	1ю/2013	Иванов	Иван	Иванович
2	2ю/2013	Петров	Петр	Петрович
3	3ю/2013	Сидоров		

Количество студентов в группе: 3

Группа: 1761

№	№Зачетки:	Фамилия:	Имя:	Отчество:
1	1э/2013	Федоров	Федор	Федорыч

Количество студентов в группе: 1

Количество студентов всего: 4

Страница 1 из 1

Рисунок 56. Отчет «Студенты» в режиме просмотра.

Форматирование отчета «Студенты» в режиме макета

Откройте отчет «Студенты» в режиме макета. В меню **Формат** нажмите кнопку **Группировка**. Внизу макета отчета появится раздел Группировка, сортировка и итоги (рис. 57).

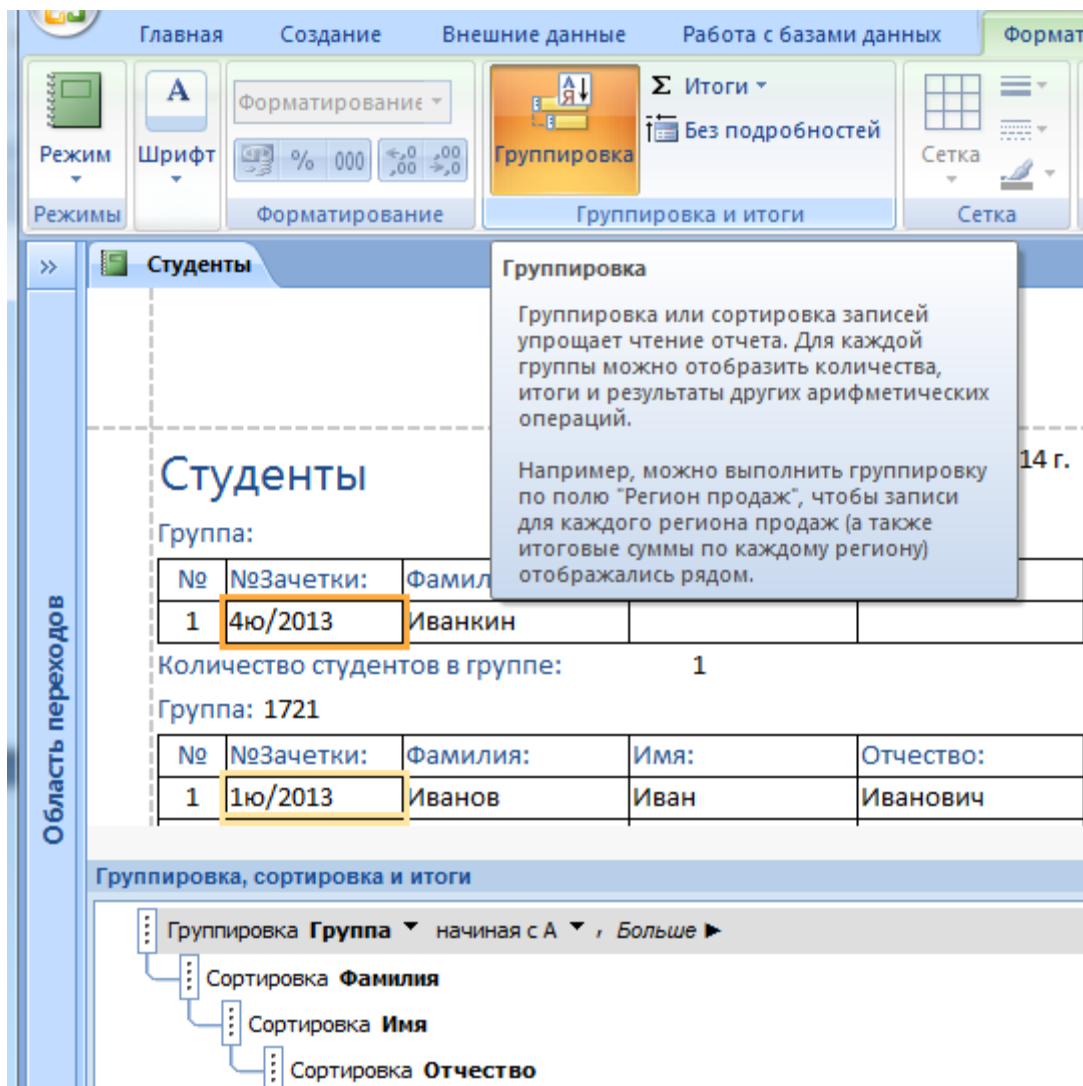


Рисунок 57. Отчет «Студенты» в режиме макета. Группировка, сортировка и итоги.

Сгруппируйте записи по группам студентов, подсчитайте количество студентов по группам, подсчитайте общее количество студентов, отсортируйте записи по фамилии, имени, отчеству.

Форматирование отчета «Студенты» в Конструкторе

Откройте отчет Студенты в режиме конструктора (рис. 58).

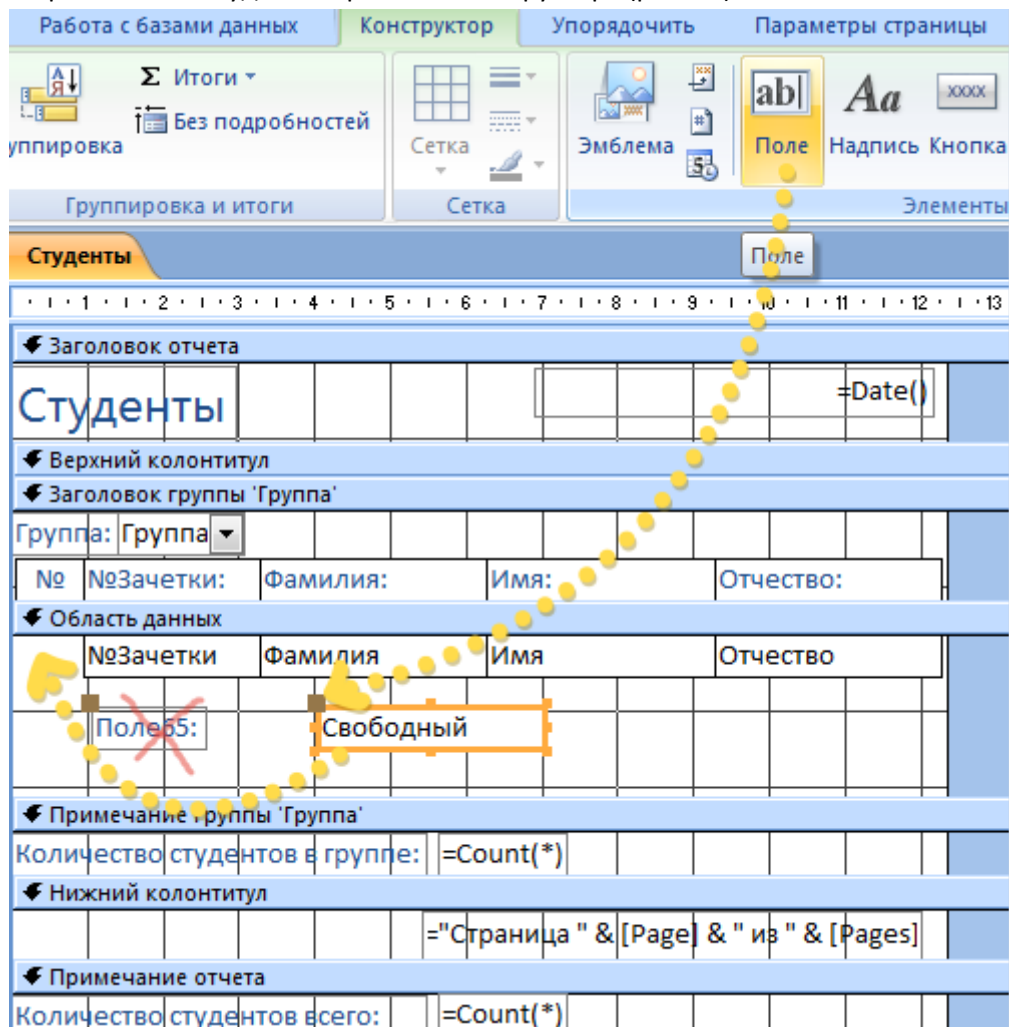


Рисунок 58. Отчет «Студенты» в Конструкторе.

Пронумеруем студентов внутри групп. Для этого слева от №зачетки освободите место, передвинув всю строку вправо и увеличив высоту раздела Область данных, и вставьте в туда **свободное поле**, а **надпись** к свободному полю удалите.

Выделите свободное поле, затем откройте **Страницу свойств** свободного поля и на вкладке **Данные** укажите (рис. 59): Данные =1; Сумма с накоплением **Для группы**.

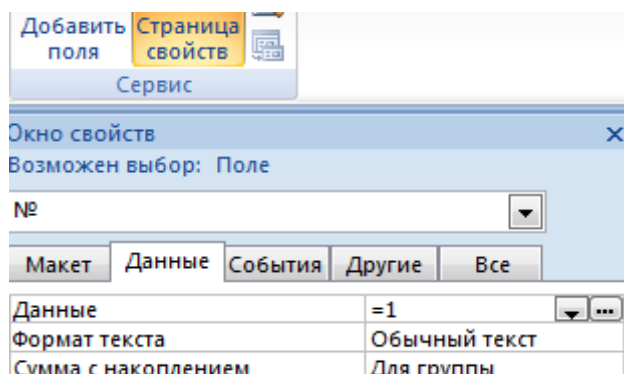


Рисунок 59. Страница свойств поля, вкладка Данные.

Результат форматирования должен выглядеть примерно как на рис. 60.

Студенты											
Заголовок отчета											
Студенты										=Date()	
Верхний колонтитул											
Заголовок группы 'Группа'											
Группа: Группа											
№	№Зачетки:	Фамилия:		Имя:		Отчество:					
Область данных											
=1	№Зачетки	Фамилия		Имя		Отчество					
Примечание группы 'Группа'											
Количество студентов в группе:										=Count(*)	
Нижний колонтитул											
										="Страница " & [Page] & " из " & [Pages]	
Примечание отчета											
Количество студентов всего:										=Count(*)	

Рисунок 60. Отчет «Студенты» в Конструкторе с нумерацией студентов внутри групп.

Отчет «Успеваемость»

С помощью Мастера отчетов на основе таблиц "Студенты" и "Успеваемость" создайте отчет "Успеваемость студентов" примерно как на рис. 50.

Успеваемость студентов		
Группа	1721	
2ю/2013	Петров	Петр Петрович
Семестр	1	
ФормаКонтроля	Зачет	
Иностранный язык в сфере юриспруденции		
История госуд. и права заруб. стран		
ИТ в юридической деятельности	зачтено	Ткаченко Л.А.
Логика		
Русский язык и культура речи		
Философия		
ФормаКонтроля	Экзамен	
Информатика	4	Ткаченко Л.А.

Рисунок 61. Отчет «Успеваемость студентов».

Отформатируйте отчет в режиме макета:

- 1) Задайте группировку по Группе, №зачетки, Семестру, Форме контроля.
- 2) Задайте сортировку по Дисциплине.

Вопросы по теме «Отчеты»

1. Что такое **отчет**? Способы создания отчетов: Мастер отчетов, Конструктор отчетов.
2. **Режимы работы с отчетом**: Представление отчета, Предварительный просмотр, Режим макета, Конструктор.
3. **Разделы отчета**: Заголовок, Примечание, Заголовок группы, Примечание группы, Верхний колонтитул, Нижний колонтитул, Область данных.
4. Что такое **свободное поле** и **вычисляемое поле** в отчете? Для чего используются?
5. Как **пронумеровать страницы** отчета?
6. Как **сортировать записи** в отчете?
7. Как **группировать записи** в отчете? Как сделать несколько уровней группировки записей в отчете?
8. Как **пронумеровать записи** в отчете? Как пронумеровать записи внутри группы в отчете?
9. Как **подсчитать количество записей** в отчете? Как подсчитать количество записей внутри группы в отчете? В каком разделе отчета следует поместить вычисляемое поле для подсчёта количества записей в группе?
10. Что делать если появляется предупреждение, что некоторые данные в отчете могут не выводиться?
11. Какие элементы управления могут быть добавлены в отчет? Как отличить элемент **Надпись** от элемента **Поле** в макете отчёта?

Управляющая форма

Наша задача – быстро и удобно получить список студентов выбранной группы.

Запрос «Студенты группы»

Чтобы сформировать список студентов создайте **запрос** на основе таблицы "Студенты". Включите в запрос следующие поля: Группа, Незачетки, Фамилия, Имя, Отчество. Дайте запросу имя "Студенты группы" (рис. 62).

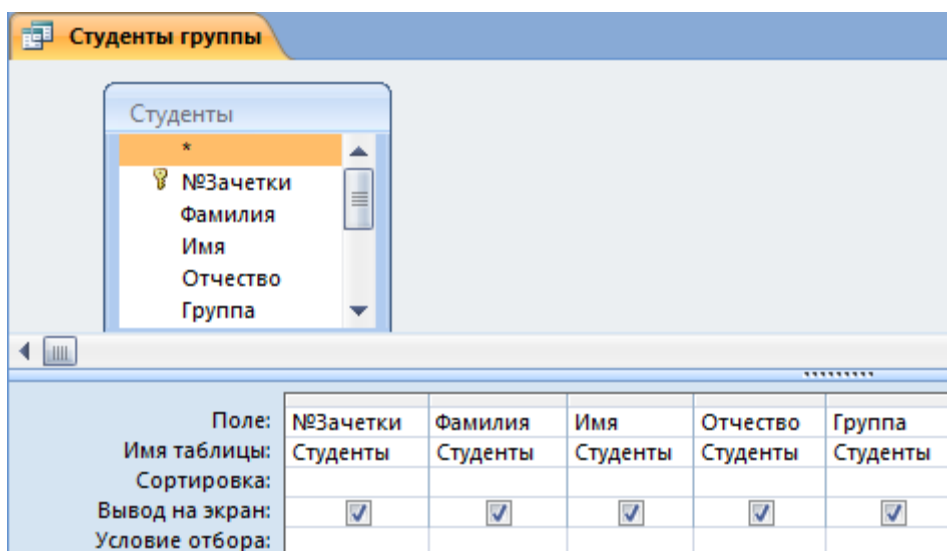


Рисунок 62. Запрос «Студенты группы» в Конструкторе.

Этот запрос покажет нам студентов всех групп. Чтобы выбрать студентов определенной группы, надо задать условие отбора по полю «Группа». Можно сделать запрос с параметром, например, записать в условии отбора по полю «Группа» выражение **[Введите код группы]**. Но это не очень удобно, надо помнить код группы и правильно его ввести. Удобнее сделать управляющую форму, в которой

создать поле со списком, которое поможет нам выбрать код группы из таблицы «Группы». Этот код группы надо будет подставить в условие отбора. Покажем, как это можно сделать.

Форма «Управляющая»

С помощью Конструктора форм создайте свободную (пустую) форму. Сохраните её, дав её имя "Управляющая". В управляющей форме создайте **поле со списком**, которое будет использовать значения из поля «Группа» таблицы «Группы», а также создайте **кнопку** для открытия запроса "Студенты группы" (рис. 63).

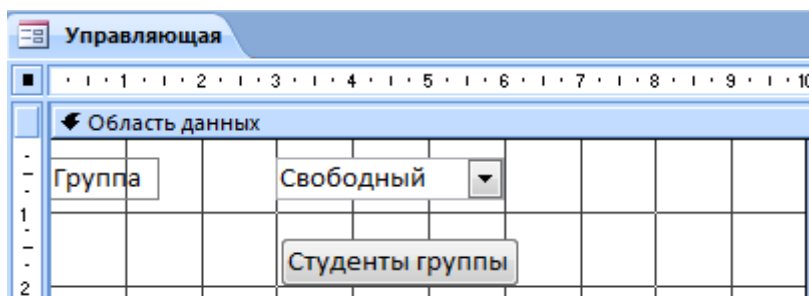


Рисунок 63. Форма «Управляющая». Поле со списком «Группа». Кнопка «Студенты группы».

Выделите поле со списком и на странице свойств дайте ему имя "Выберите группу" (рис. 64).

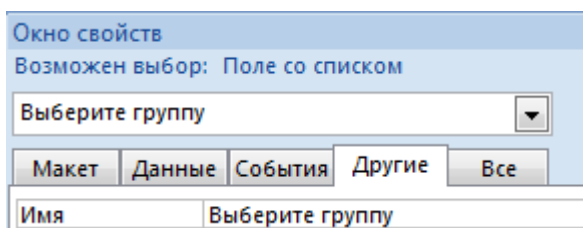


Рисунок 64. Задание имени полю со списком.

Организация связи управляющей формы с запросом «Студенты группы»

В запросе "Студенты группы" в условии отбора для поля "Группа" (рис.65) с помощью **Построителя выражений** создайте ссылку на поле со списком "Выберите группу" из формы "Управляющая":

=[Forms]![Управляющая]![Выберите группу]

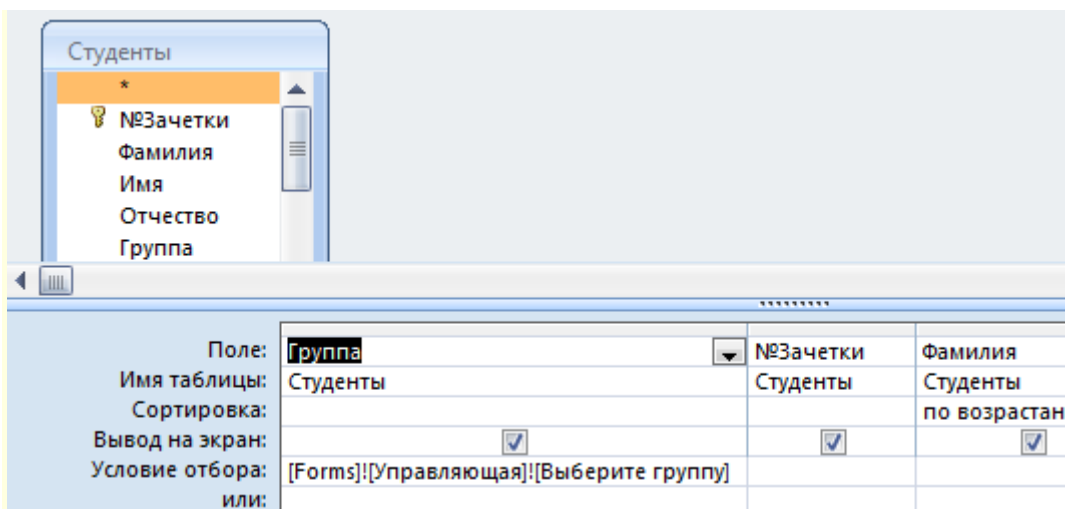


Рисунок 65. Организация связи формы «Управляющая» с запросом «Студенты группы».

Проверьте работу управляющей формы. Для этого откройте форму «Управляющая», выберите какую-либо группу с помощью поля со списком «Группа», нажмите кнопку «Студенты группы». Откроется запрос со списком студентов выбранной группы.

Самостоятельная работа

Попробуйте создать управляющую форму, которая бы была единым центром доступа ко всем объектам базы данных по учету успеваемости студентов. Примером может послужить следующая управляющая форма (рис. 66).

Управляющая форма

Ввод и просмотр данных

Персональные данные студентов

Успеваемость студентов

Дисциплины

Типы оценок

Преподаватели

Группы

Получение отчетов

Группа

Дисциплина

Экзаменационная ведомость незаполненная

Экзаменационная ведомость с оценками

Списки групп студентов

Успеваемость студентов

Получение запросов

Список студентов

Количество студентов в группах

Сводная ведомость успеваемости

Группы без студентов

Повторы записей в Успеваемости

Возраст студентов

[Сайт СПФ МГИУ](#)

Выйти из приложения

Рисунок 66. Пример управляющей формы базы данных «Успеваемость студентов».

Вопросы по теме Управляющая форма

1. Что такое управляющая **форма**? Преимущества использования управляющих форм?
2. Что такое **свободная форма**? Что такое свойство **Источник записей** формы?
3. Что такое **элемент управления**? Какие элементы можно поместить в форму?
4. Элементы управления: **Надпись, Кнопка, Поле, Поле со списком**. Их назначение? Что такое **присоединенное** поле и **свободное** поле, в чем их отличие? Можно ли использовать свободное поле для вычислений?
5. Кнопка **Использовать мастера**? Для чего используется?
6. Что такое **страница свойств**? Свойства элементов управления: свойство **Имя**, свойство **Данные**.
7. Разделы формы: Заголовок и примечание формы, Колонтитулы, Область данных.
8. **Список полей** формы?
9. Как организовать связь управляющей формы с запросом?

Контрольные вопросы

1. Системы управления базами данных. Назначение, возможности.
2. Базы данных. Назначение. Преимущества.
3. Понятие реляционной базы данных. Её преимущества.
4. Основные этапы проектирования баз данных.
5. Объекты базы данных, их назначение и взаимосвязь.
6. Таблица. Её структурные элементы. Способы создания и работы с ней.
7. Поле таблицы. Типы данных поля. Свойства поля.
8. Понятие ключевого поля.
9. Поле со списком в таблице. Свойства поля со списком.
10. Запись в таблице. Сортировка и фильтрация записей в таблице.
11. Схема данных. Нарисуйте схему данных и поясните, что на ней изображено.
12. Первичный и внешний ключ. Правила целостности для связанных полей
13. Связи между таблицами. Типы связей.
14. Главная и подчиненная таблицы. Очередность их заполнения
15. Целостность данных. Преимущества целостной базы данных.
16. Запрос. Типы запросов. Конструирование запроса.
17. Сортировка и отбор записей в запросе. Построитель выражений.
18. Вычисления в запросе. Вычисляемые поля в запросе. Построитель выражений.
19. Условия отбора в запросе. Формирование запросов с операциями AND и OR.
20. Сортировка и группировка записей в запросе.
21. Форма. Назначение. Структура формы. Способы создания. Работа с формой.
22. Элементы управления, используемые в форме. Построитель выражений.
23. Отчет. Структура отчета. Способы создания и форматирования.
24. Нумерация записей в отчете.
25. Группировка и сортировка записей в отчете.
26. Элементы управления, используемые в отчете. Построитель выражений.
27. Управляющая форма. Назначение. Свойство Источник записей формы.
28. Элементы управления, используемые в управляющей форме. Построитель выражений.
29. Защита информации в базе данных.

Контрольные задания

Задание 01

Создать базу данных "Расписание занятий".

Календарь на семестр. Расписание пар. Аудитории. Группы студентов. Учебные дисциплины и количество аудиторных часов в семестре. Преподаватели. Подсчитать количество часов для групп по дисциплинам, для преподавателей по дисциплинам, для аудиторий. Показать расписание для группы, для преподавателя, для аудитории.

Задание 02

Создать базу данных "Абитуриенты"

Учет абитуриентов, поступающих в университет из разных населенных пунктов на выбранные специальности. Известен средний школьный балл каждого абитуриента. Справки о количестве абитуриентов из разных населенных пунктов, количестве абитуриентов, выбравших ту или иную специальность. Справки о количестве абитуриентов - отличников, хорошистов, троечников. Отчет о

количественном составе абитуриентов. Списки абитуриентов, выбравших ту или иную специальность. Поздравление абитуриента с днем рождения.

Задание 03

Создать базу данных "Телефонная книга"

Университет состоит из нескольких факультетов, на каждом факультете несколько групп. В каждой группе несколько студентов. У студентов известны домашние адреса, номера телефонов. Выдать отчет Телефонная книга студентов. По фамилии студента определить его номер телефона и адрес. По номеру телефона определить фамилию и адрес студента. По запросу методиста представить список студентов выбранной группы, список студентов из выбранного населенного пункта (с телефонами и адресами).

Задание 04

Создать базу данных "Учебные планы"

Список дисциплин. Список специальностей. Список семестров. Количество часов (всего, лекции, практикум, саморабота) на дисциплину в семестре. Формы контроля (зачет, дифзачет, экзамен). Учебный план по специальностям на все семестры.

Задание 05

Создать базу данных "Сотрудники".

Университет состоит из нескольких отделов. Имеется список сотрудников университета. Известен уровень образования сотрудников. Известны даты рождения, адреса и телефоны сотрудников. Справки и отчеты о количестве сотрудников в целом по университету, в каждом отделе, а также количество сотрудников с высшим или средним образованием. По запросу представить список сотрудников выбранного отдела, список сотрудников из выбранного населенного пункта (с телефонами и адресами). Конкретному сотруднику выдать отчет-поздравление с днем рождения.

Задание 06

Создать базу данных "Выпускники".

Университет готовит студентов по нескольким специальностям. Имеется список выпускников. Институт поддерживает связи со своими выпускниками, которые сообщают о месте и характере своей работы. Справки и отчеты о количестве выпускников всего по институту, а также трудоустроенных и безработных. Узнать, какие специальности наиболее востребованы на рынке труда.

Задание 07

Создать базу данных "Преподаватели".

Список кафедр университета. Список преподавателей университета, каждый преподаватель входит в состав какой-либо кафедры. Список преподаваемых учебных дисциплин. Каждый преподаватель может вести преподавание по нескольким дисциплинам. Одну и ту же дисциплину могут преподавать несколько преподавателей. Преподаватель может иметь или не иметь ученую степень (кандидат наук, доктор наук). Подсчитать общее количество преподавателей, количество преподавателей по учебным дисциплинам, количество преподавателей с учеными степенями. Представить список учебных дисциплин, по которым в институте нет преподавателей. Представить список преподавателей, ведущих ту или иную дисциплину. Представить список преподавателей с учеными степенями. Представить список кафедр, список и количество преподавателей кафедры.

Задание 08

Создать базу данных "Библиотека".

Библиотека ведет пополнение и учет библиотечного фонда, а также выдачу книг читателям. Книги распределены по рубрикам (экономика, юриспруденция, психология, информатика и т.д.). Учет книг. Учет читателей. Учет выдачи книг читателям. Подсчитать общее количество книг всего в библиотеке и по каждой рубрике. По заявке читателя узнать имеется ли книга в наличии или у кого она на руках. По заявке читателя получить список книг по данной рубрике, определенного автора, издательства или года выпуска. По запросу библиотекаря узнать какие книги числятся за читателем.

Задание 09

Создать базу данных "Издательский отдел".

Издательский отдел издает учебные материалы преподавателей университета. Ведется список преподавателей института, список кафедр института. Каждый преподаватель входит в состав какой-либо кафедры. Ведется перечень изданных учебных материалов с указанием названия, авторов, года издания, количества страниц. Учебные материалы подразделяются по типам (учебники, учебные пособия, конспекты лекций, практикумы и т.д.) и по учебным дисциплинам (математика, информатика, социология и т.д.). Учет авторов, кафедр, изданий. Перечень изданий, сгруппированный по кафедрам и учебным дисциплинам. Количество изданий кафедр и института за год, за последние 5 лет. Перечень и количество изданий по типам учебных материалов за определенный период. Перечень и количество изданий автора за определенный период.

Задание 10

Создать базу данных "Информационный отдел".

Информационный отдел накапливает и хранит электронные учебные материалы (издания) преподавателей университета. Ведется список преподавателей, список кафедр университета. Каждый преподаватель входит в состав какой-либо кафедры. Ведется перечень электронных учебных материалов с указанием названия, авторов, года издания. Электронные учебные материалы подразделяются по типам (учебники, практикумы, тесты и т.д.), по учебным дисциплинам (математика, информатика, социология и т.д.). Учет авторов, изданий. Перечень изданий, сгруппированный по кафедрам и учебным дисциплинам. Количество изданий кафедр и университета, за год, за последние 5 лет. Перечень и количество изданий по типам учебных материалов за определенный период. Перечень и количество изданий автора за определенный период.

Заключение

Практикум Access дает базовые знания и навыки разработки учебной базы данных на примере хорошо знакомой студенту предметной области, связанной с учебным процессом. Решена сквозная задача разработки базы данных учета успеваемости студентов. В практикуме не освещены вопросы создания макросов и модулей. Желющие углубить свои знания в области баз данных могут обратиться к учебным ресурсам, часть из которых представлена ниже.