

ЛАБОРАТОРНАЯ РАБОТА № 10

Тема: **УСТАНОВЛЕНИЕ СВЯЗЕЙ МЕЖДУ ТАБЛИЦАМИ**

1 ЦЕЛЬ РАБОТЫ

Научиться:

- устанавливать связи между таблицами;
- удалять межтабличные связи;
- просматривать связанные записи.

2 ОБЪЁМ ЛАБОРАТОРНОЙ РАБОТЫ И ПОРЯДОК ЕЁ ВЫПОЛНЕНИЯ

Данная лабораторная работа выполняется в течение двух часов. Изучение теоретического материала идёт одновременно с выполнением практических заданий, для закрепления которых нужно выполнить упражнения для самостоятельной работы. Результаты выполнения лабораторной работы необходимо сохранить в указанной преподавателем папке.

3 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ. ТЕХНОЛОГИЯ ВЫПОЛНЕНИЯ РАБОТЫ

Создайте в режиме дизайна новую таблицу **Успеваемость**, принадлежащую базе данных **Студенты**. Для этого:

1. Запустите программу LibreOffice Base (Приложения – Офис – LibreOffice Base).

2. В окне Мастер базы данных на первом шаге выберите команду **Открыть файл существующей базы данных**. В раскрывающемся списке **Последние:** выберите **Студенты**. Если имя вашей базы данных не отображается в списке **Последние:**, то нажмите кнопку **Открыть** и в появившемся окне укажите путь к созданной ранее базе данных **Студенты** и нажмите кнопку **Открыть**.

3. Выберите в левом секции пиктограмму **Таблицы** и в правой секции щелкните по команде **Создать таблицу в режиме дизайна...**

4. Самостоятельно в **Конструкторе таблицы** опишите все поля новой таблицы, пользуясь данными из [таблицы 10.1](#).

5. Сохраните новую таблицу с именем **Успеваемость**. Если вы всё сделали правильно, то в секции **Таблицы** вы увидите два объекта **Анкета** и **Успеваемость** ([рисунок 10.1](#)).

Таблица 10.1 – Данные для макета таблицы

Название поля	Тип поля	Длина	
Номер группы	Текст [VARCHAR]	5	
Номер зачётной книжки	Текст [VARCHAR]	5	Первичный ключ
Фамилия	Текст [VARCHAR]	12	
Имя	Текст [VARCHAR]	10	
Инженерная графика	Вещественное [REAL]		
Химия	Вещественное [REAL]		
Теоретическая механика	Вещественное [REAL]		
Информатика	Вещественное [REAL]		

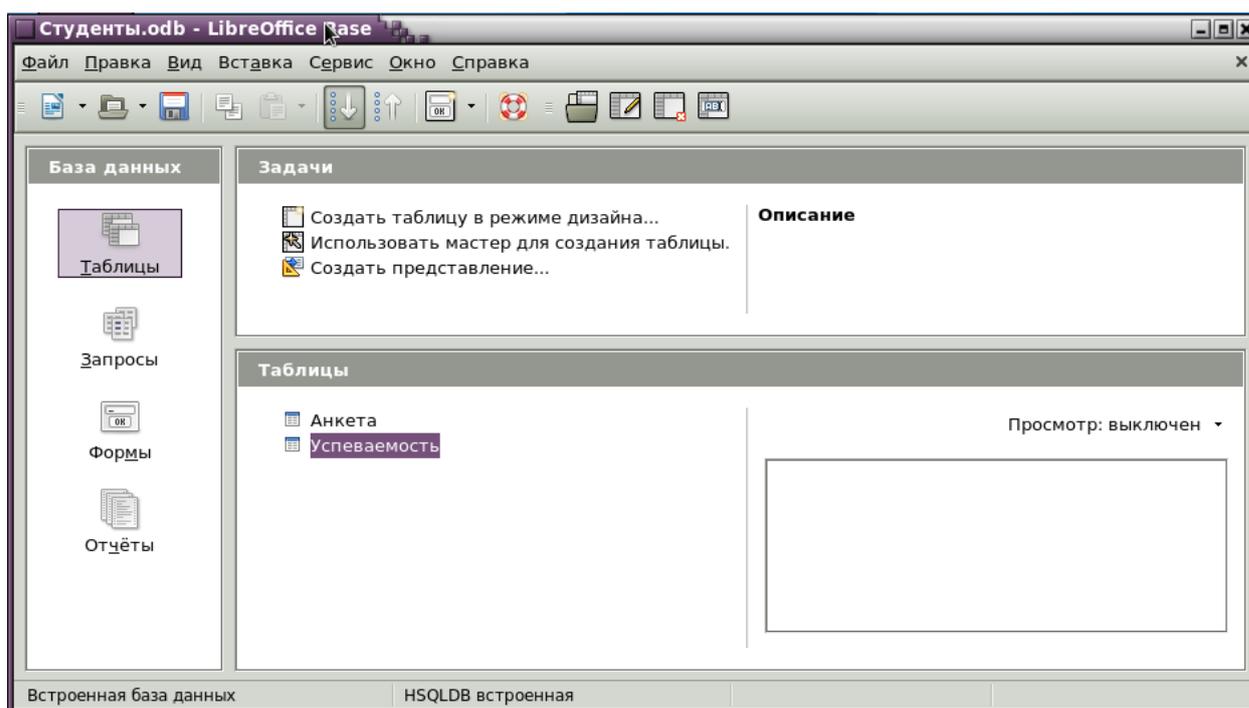


Рисунок 10.1 – Две таблицы содержащиеся в базе данных

6. Откройте таблицу **Успеваемость** и занесите в неё записи из [таблицы 9.6](#) и [таблицы 9.7](#).
7. Также в таблицу **Успеваемость** занесите оценки студентов в соответствии с [таблицей 10.2](#).
8. Закройте таблицу **Успеваемость**, сохранив данные.

Таблица 10.2 – Оценки студентов

Фамилия	Инженерная графика	Химия	Теоретическая механика	Информатика
Акопян	3	2	2	2
Васюшкина	3	4	3	5
Литвинов	5	5	5	5
Портнова	3	4	4	4
Рыковский	4	4	3	5
Бражников	5	5	4	5
Ефимов	4	2	2	3
Коптева	4	5	3	4
Нестеров	3	2	3	2
Сотченко	4	4	5	5
Ткаченко	3	2	2	3
Замковая	3	4	5	5
Кишко	4	3	4	4
Лазарев	3	4	4	4
Скринникова	4	5	4	5

3.1 Установление связи между таблицами

Теперь, когда ваша база данных состоит из двух таблиц, нужно сообщить программе LibreOffice Base, как эти таблицы связаны друг с другом. Позднее LibreOffice Base будет использовать эти связи в запросах и отчетах.

Типы связей

Существует три типа межтабличных связей: *один-к-одному*, *один-ко-многим* и *многие-ко-многим*.

Если две таблицы связаны соотношением *один-к-одному*, то каждой записи в первой таблице соответствует не более чем одна запись во второй таблице, и наоборот, каждой записи во второй таблице соответствует не более чем одна запись в первой таблице. Например, и в таблице **Анкета**, и в таблице **Успеваемость** имеется поле **Номер зачётной книжки**, содержащее уникальные данные. Если эти две таблицы связать посредством поля **Номер зачётной книжки**, то между ними установится связь *один-к-одному*.

В более общей связи *один-ко-многим* каждой записи в первой таблице может соответствовать несколько записей во второй таблице, однако любой записи во второй таблице соответствует только одна запись в первой таблице. Например, в группе могут учиться много студентов, но каждый студент учится только в одной группе.

Связь *многие-ко-многим* означает, что каждая запись в первой таблице может быть связана со многими записями во второй таблице, а каждая запись

во второй таблице – со многими записями в первой таблице. Например, у каждого студента может быть несколько преподавателей, а у каждого преподавателя может быть много студентов. Современные СУБД не позволяют создавать связи типа *многие-ко-многим* напрямую. Однако в реальной жизни такие связи встречаются очень часто, поэтому их реализуют через вспомогательные таблицы, увязывая несколько таблиц связями типа *один-ко-многим*.

Установление связи

Установим связь между таблицами **Анкета** и **Успеваемость** посредством их общего поля **Номер зачётной книжки**. Для этого:

1. Убедитесь в том, что поле **Номер зачетной книжки** является ключевым и в таблице **Анкета**, и в таблице **Успеваемость**.
2. Выполните команду меню **Сервис – Связи...** В результате открылось новое окно **Конструктор связей** и одновременно с ним открылось диалоговое окно **Добавить таблицы** ([рисунок 10.2](#)).

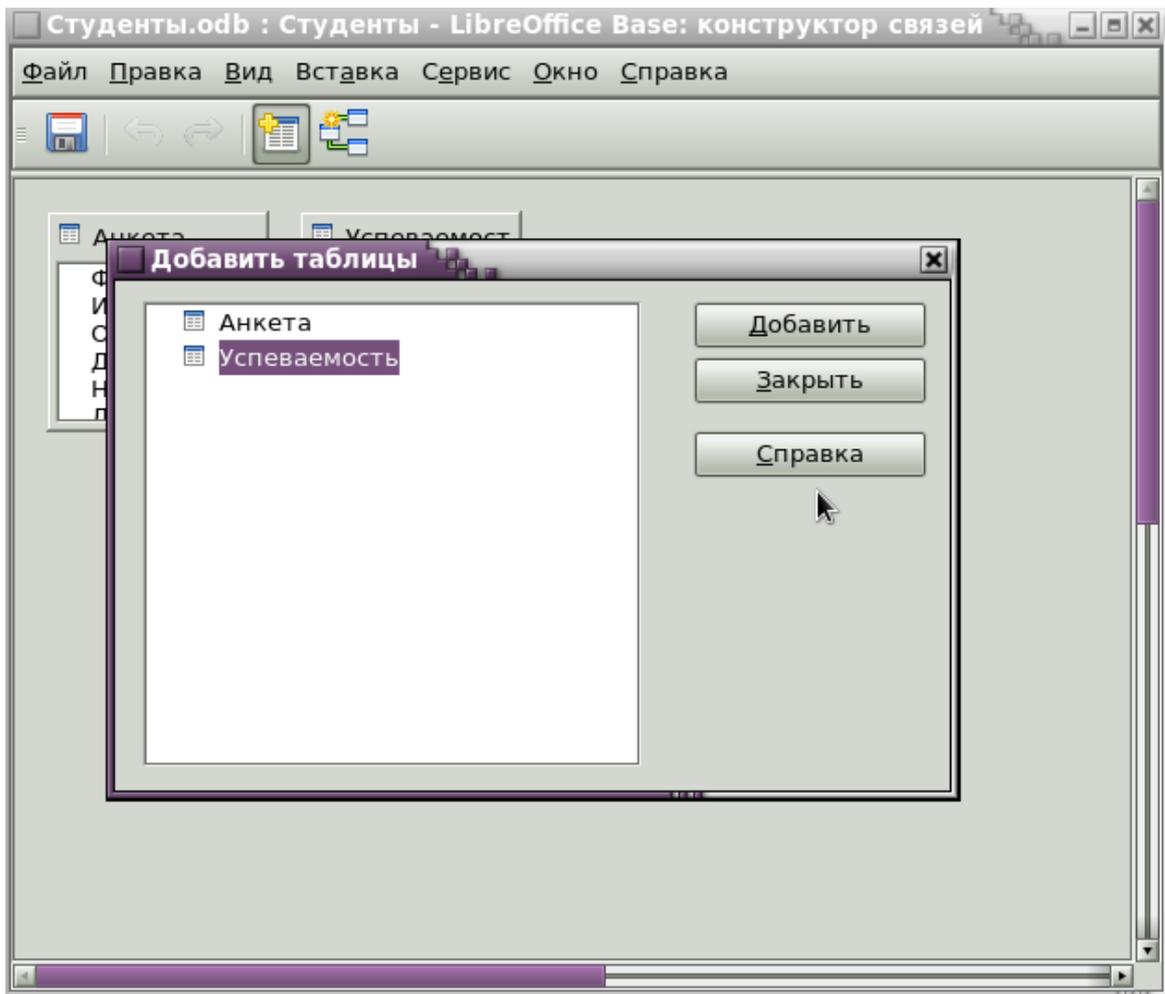


Рисунок 10.2 – Окно Конструктор связей и окно Добавить таблицы

3. Добавьте по одной таблицы **Анкета** и **Успеваемость** в окно **Конструктор связей**. Для этого:

- в окне **Добавить таблицы** выделите таблицу **Анкета** и щелкните по кнопке **Добавить**;

- затем выделите таблицу **Успеваемость** и щелкните также по кнопке **Добавить**.

- закройте окно **Добавить таблицы**, щелкнув по кнопке **Закреть**.

4. Увеличьте размер окон **Анкета** и **Успеваемость** так, чтобы названия всех полей были видны полностью.

5. Обычно одна из таблиц условно считается главной, другая подчиненной, пусть **Анкета** будет главной таблицей, а **Успеваемость** – подчиненной. Тогда, чтобы установить связь, нужно в окне **Конструктор связей** в главной таблице **Анкета** выделить поле **Номер зачётной книжки** и перетащить его на одноименное поле **Номер зачётной книжки** подчиненной таблицы **Успеваемость**. Сделайте это. В окне между таблицами должна появиться линия связи с указанием типа связи в данном случае один-к-одному (1–1) ([рисунок 10.3](#)). Если линия связи не появилась, попробуйте установить её от подчинённой таблицы **Успеваемость** к главной **Анкета**. В **LibreOffice Base** допустимо установление линии связи между таблицами от подчинённой таблицы к главной.

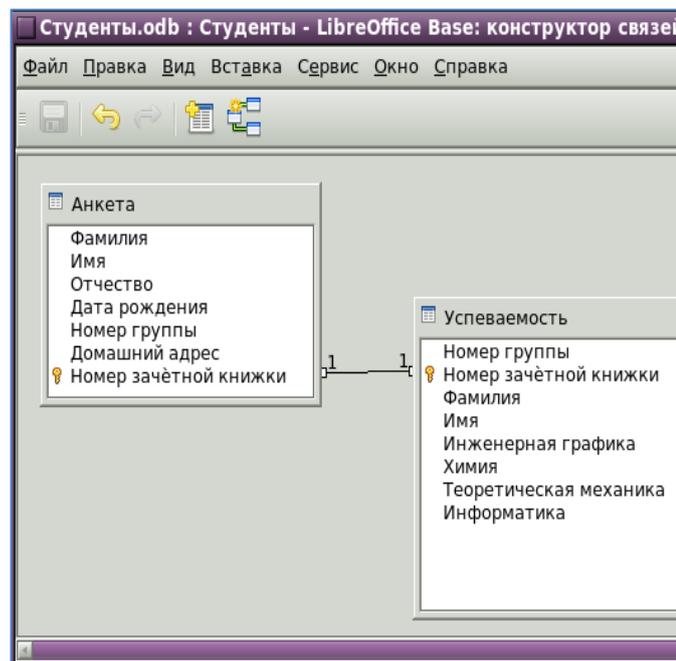


Рисунок 10.3 – Конструктор связей

6. Помимо создания связи между таблицами необходимо выполнить настройки параметров межтабличной связи. Для этого:

- в окне **Конструктор связей** правой кнопкой мыши щелкните по линии связи и выберите в контекстном меню команду **Правка...**

- откроется диалоговое окно **Связи** ([рисунок 10.4](#)), в этом окне уста-

новите **обновить каскадно** и **удалить каскадно**, подтвердите изменения нажав кнопку **Да**.

Связи

Таблицы

Успеваемость Анкета

Поля

Успеваемость	Анкета
Номер зачётной книжки	Номер зачетной книжки

При обновлении

не изменять
 обновить каскадно
 установить NULL
 установить умолчание

При удалении

не изменять
 удалить каскадно
 установить NULL
 установить умолчание

Да Отменить Справка

Рисунок 10.4 – Настройка параметров межтабличной связи

Параметры межтабличной связи:

Не изменять — указывается, что любое изменение первичного ключа не должно затрагивать остальные внешние ключевые поля.

Обновить каскадно — происходит обновление всех внешних ключевых полей при изменении соответствующего первичного ключа (обновление каскадом).

Установить NULL — если соответствующий первичный ключ изменен, этот параметр используется для задания значения "ПУСТО" всем внешним ключевым полям. Это значение указывает, что поле пустое.

Установить умолчание — если соответствующий первичный ключ изменен, этот параметр используется для задания значения по умолчанию всем внешним ключевым полям. При создании соответствующей таблицы значение по умолчанию внешнего ключевого поля определяется при задании

свойств данного поля.

Аналогичный смысл имеют параметры удаления.

Если установить переключатель **обновить каскадно**, то LibreOffice Base позволит изменять номера зачетных книжек студентов в главной таблице. При этом автоматически будут изменяться эти же номера в подчиненной таблице.

Если разрешить **удалить каскадно**, то вы сможете удалять записи в главной таблице. При этом автоматически будут удаляться соответствующие записи в подчиненной таблице.

7. Закройте окно **Конструктор связей**. В ответ на предложение сохранить изменения щелкните по кнопке **Да**.

8. Вызовите на экран таблицу **Успеваемость**. Измените номер зачетной книжки студента Литвинова В.Н., например, на 00555. Сохраните изменения.

9. Откройте таблицу **Анкета**. Найдите запись о Литвинове В.Н. Вы видите, что номер зачетной книжки этого студента автоматически изменился. Почему? Потому что при связывании таблиц **Анкета** и **Успеваемость** вы разрешили обновить каскадно связанные поля. Закройте окно таблицы **Анкета**.

10. В таблице **Успеваемость** в столбце **Номер зачётной книжки** вместо номера 00555 введите прежнее значение, т.е. 00859. Сохраните изменения.

11. Убедитесь в том, что в таблице **Анкета** произошли соответствующие изменения.

12. Закройте окна таблиц **Анкета** и **Успеваемость**.

3.2 Удаление связи между таблицами

Чтобы удалить связь между таблицами **Анкета** и **Успеваемость**, выполните следующие действия:

1. Выполните команду меню **Сервис – Связи...** чтобы снова открыть окно **Конструктор связей**.

2. Щелкните правой кнопкой мыши по линии, соединяющей две таблицы.

3. В появившемся контекстном меню выберите команду **Удалить**.

4. Линия связи удаляется без подтверждения.

5. Свяжите снова таблицы **Анкета** и **Успеваемость** посредством их общего поля **Номер зачётной книжки** еще раз. Установите переключатели обновить каскадно и удалить каскадно. Закройте и сохраните изменения в Конструкторе связей.

4 ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

1. В режиме дизайна создайте таблицу **Кураторы**, принадлежащую базе данных **Студенты** ([рисунок 10.5](#)). Заполните эту таблицу данными из [таблицы 10.3](#).

Таблица 10.3 – Кураторы

Номер группы	Фамилия	Имя	Отчество
МП-11	Матвейкина	Жанна	Владимировна
МП-12	Самойлова	Татьяна	Филипповна
МП-13	Самойлова	Ирина	Валентиновна

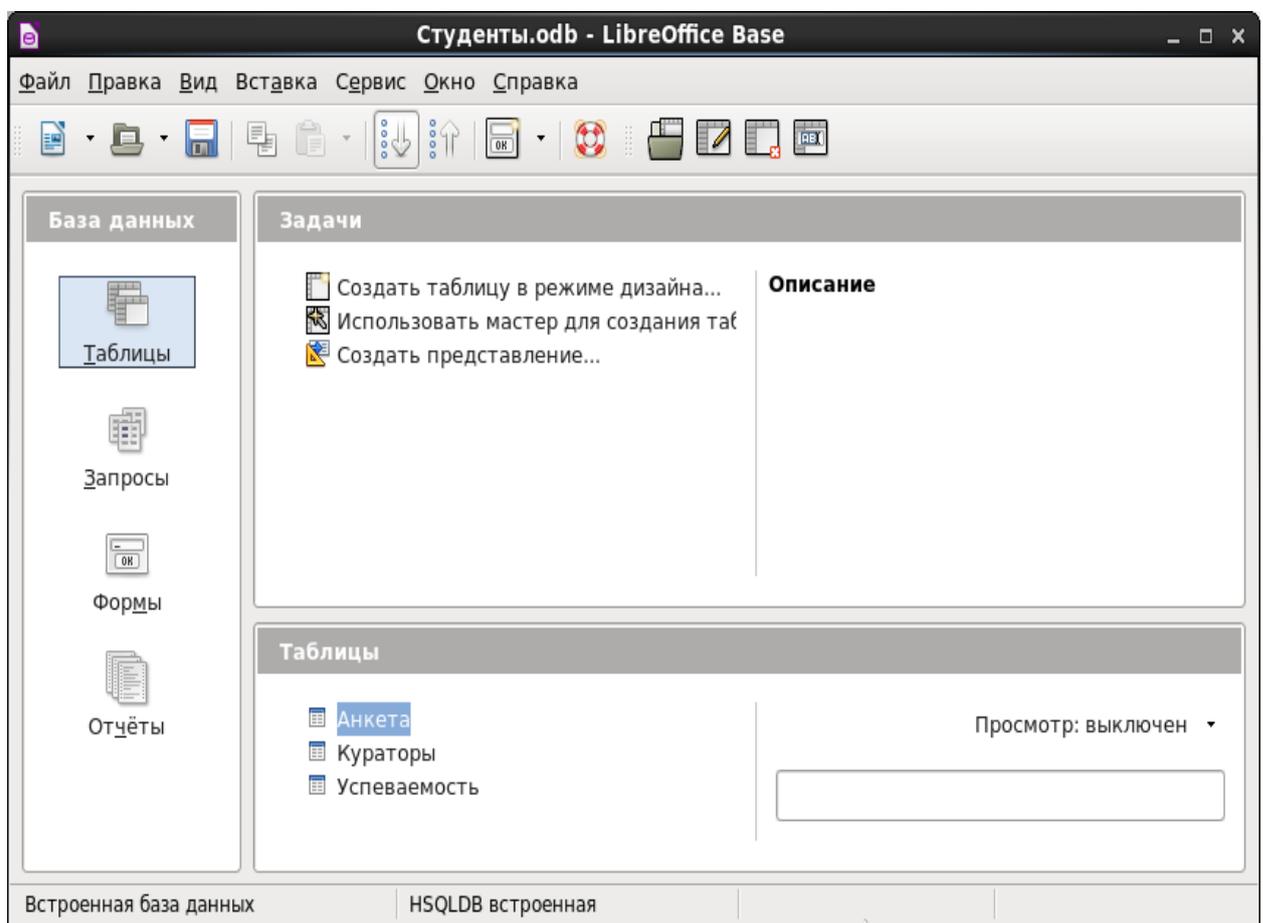


Рисунок 10.5 – Таблица **Кураторы** в списке созданных таблиц

2. В таблице **Кураторы** назначьте полю **Номер группы** Первичный ключ.
3. Откройте окно Конструктор связей.
4. На панели инструментов щелкните по кнопке  – Добавить таблицы и добавьте таблицу **Кураторы** в конструктор связей.
5. Свяжите таблицы **Кураторы** и **Успеваемость** посредством их обще-

го поля **Номер группы**. Установите для созданной связи параметры **обновить каскадно** и **удалить каскадно**, подтвердите изменения.

После установления связи схема данных должна выглядеть примерно так, как показано на [рисунке 10.6](#). Обратите внимание на линию, соединяющую таблицы **Кураторы** и **Успеваемость**. В начале линии (со стороны таблицы **Кураторы**) изображена цифра 1, в конце (со стороны таблицы **Успеваемость**) изображен символ списка (☐). Это значит, что записи этих таблиц находятся в соотношении один-ко-многим ([рисунк 10.6](#)).

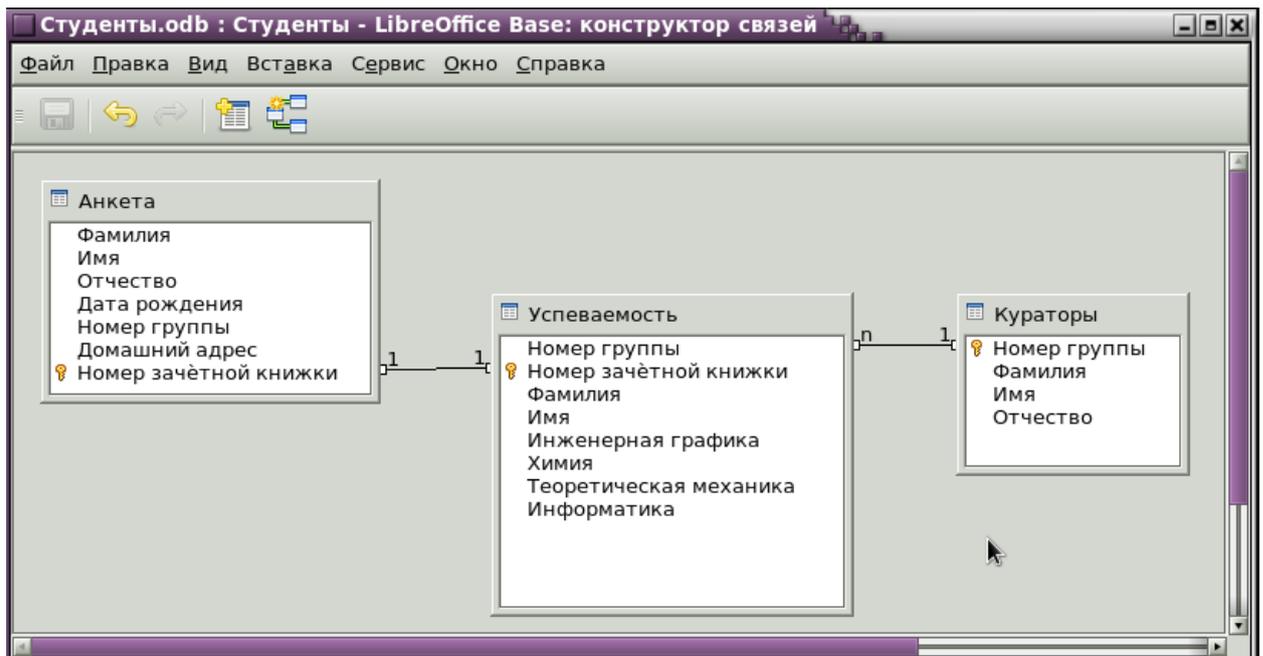


Рисунок 10.6 – Окно Конструктор связей

6. Закройте окно **Конструктор связей**, сохранив внесенные изменения.
7. Закройте окно таблицы **Кураторы**, затем окно базы данных **Студенты**.

5 КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какие типы межтабличных связей Вам известны?
2. Что такое внешний ключ?
3. Для чего нужны переключатели **обновить каскадно** и **удалить каскадно**?
4. Как вызвать на экран окно **Конструктор связей**? Окно **Добавить таблицы**?
5. Что означают символы 1 и ☐ в схеме данных?
6. Как установить связь между таблицами?
7. Как удалить межтабличную связь?

ЛАБОРАТОРНАЯ РАБОТА № 11

Тема: **ЗАПРОСЫ НА ВЫБОРКУ, УСЛОВИЯ В ЗАПРОСАХ**

1 ЦЕЛЬ РАБОТЫ

Познакомиться:

– с назначением, структурой и порядком создания запросов на выборку.

Научиться:

- создавать запросы на выборку;
- создавать запросы с условиями;
- создавать запросы с параметром.

2 ОБЪЁМ ЛАБОРАТОРНОЙ РАБОТЫ И ПОРЯДОК ЕЁ ВЫПОЛНЕНИЯ

Данная лабораторная работа выполняется в течение двух часов. Изучение теоретического материала идёт одновременно с выполнением практических заданий, для закрепления которых нужно выполнить упражнения для самостоятельной работы. Результаты выполнения лабораторной работы необходимо сохранить в указанной преподавателем папке.

3 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ. ТЕХНОЛОГИЯ ВЫПОЛНЕНИЯ РАБОТЫ

3.1 Назначение и роль запросов в СУБД LibreOffice Base

В общем случае база данных LibreOffice Base состоит из нескольких видов объектов, только один из которых таблицы. Следующий вид объектов – запросы. Запросы – это тоже таблицы, но **временные и результирующие**, созданные по основным. Временными таблицы запросов называются потому, что не запоминаются на диске, а результирующими – потому что создаются они как результат отбора и обработки данных из основных таблиц. Назначение запросов заключается в выполнении конкретных действий с информацией из основных таблиц: отбора по какому-либо параметру (ключу), проведения расчетов, упорядочивания и т.п. При большом объеме данных запрос позволяет быстро вывести нужные данные и провести их обработку.

С помощью запросов также реализуется важный принцип работы СУБД – *разграничение доступа*. Согласно этому принципу вся информация, хранящаяся в основных таблицах, не должна быть доступна всем. Каждый пользователь имеет право только на ту ее часть, которая нужна ему для конкретной работы. Это позволяет исключить повреждения базы данных и различные злоупотребления со стороны пользователей. В то же время благодаря разграничению доступа с одной базой данных может работать множество пользователей, не мешая друг другу.

3.2 Виды и принцип действия и запросов

Как средство обработки данных, запросы – наиболее универсальный и гибкий инструмент LibreOffice Base. С их помощью можно проводить самую разнообразную обработку информации: отбирать, сортировать, фильтровать, выполнять довольно сложные расчеты, подводить итоги и даже изменять основные таблицы. Соответственно этому различают **запросы на выборку, вычисляемые запросы, итоговые запросы, запросы на изменение.**

Принцип действия запросов состоит в том, что запрос выполняет свою работу без изменения основных таблиц (за исключением запросов на изменение). Он только извлекает данные из одной или нескольких таблиц, обрабатывает их и предоставляет пользователю результат во временной результирующей таблице. Подобный принцип действия похож на фотосъемку, поэтому иногда запрос называют «мгновенным снимком» с основных таблиц.

Еще одно характерное отличие запросов заключается в том, что результирующая таблица запроса не сохраняется на диске. Тем не менее, запрос запоминается в общей базе данных, но не как таблица с данными, а как некая управляющая структура, т.е. программа работы запроса. Каждый раз, когда пользователь открывает запрос, происходит выполнение этой программы, причем вследствие высокой скорости работы компьютера конечный результат – таблица с данными – появляется почти мгновенно. Такой принцип действия очень удобен на практике: запрос всегда отображает текущее содержание основных таблиц, и если в них произошли изменения, то они автоматически будут отражены и в запросе.

Рассмотрим наиболее распространенный вид запросов – запросы на выборку. В этом виде запросов данные извлекаются из одной или нескольких основных таблиц, а также и из других запросов. Кроме выборки, возможна и другая обработка информации: сортировка, фильтрация, вычисления и т.п.

3.3 Создание запросов на выборку

Для создания запросов существует специальный язык **SQL** (Structured Query Language), однако на практике знать его вовсе не обязательно. Удобство LibreOffice Base заключается в том, что любой запрос можно создавать «с нуля» в диалоговом режиме, т.е. без программирования. Компьютер в этом случае автоматически генерирует программный код SQL, который пользователь при необходимости может просмотреть и отредактировать. Это запросы QBE (англ. Query by Example, запрос по образцу) – способ создания запросов к базе данных с использованием образцов значений полей в виде текстовой строки. Реализации QBE преобразуют пользовательский ввод в формальный запрос к базе данных, что позволяет пользователю создавать сложные запросы без необходимости изучать более сложные языки запросов.

Для диалогового создания запросов LibreOffice Base предлагает несколько способов:

1. Создать запрос в режиме дизайна...
2. Использовать мастер для создания запроса...
3. Создать запрос в режиме SQL...

Самый быстрый из них – это **Использовать мастер для создания запроса...** Рассмотрим последовательность действий в этом режиме. Конечно, перед созданием запроса всегда должны быть созданы основные таблицы БД. Выполните следующие действия:

1. Откройте исходную базу данных **Студенты**.
2. В секции **База данных** щелкните по пиктограмме **Запросы**, в секции **Задачи** щелкните команду **Использовать мастер для создания запроса...** Откроется диалоговое окно **Мастер запросов** содержащее 8 шагов ([рисунок 11.1](#)).

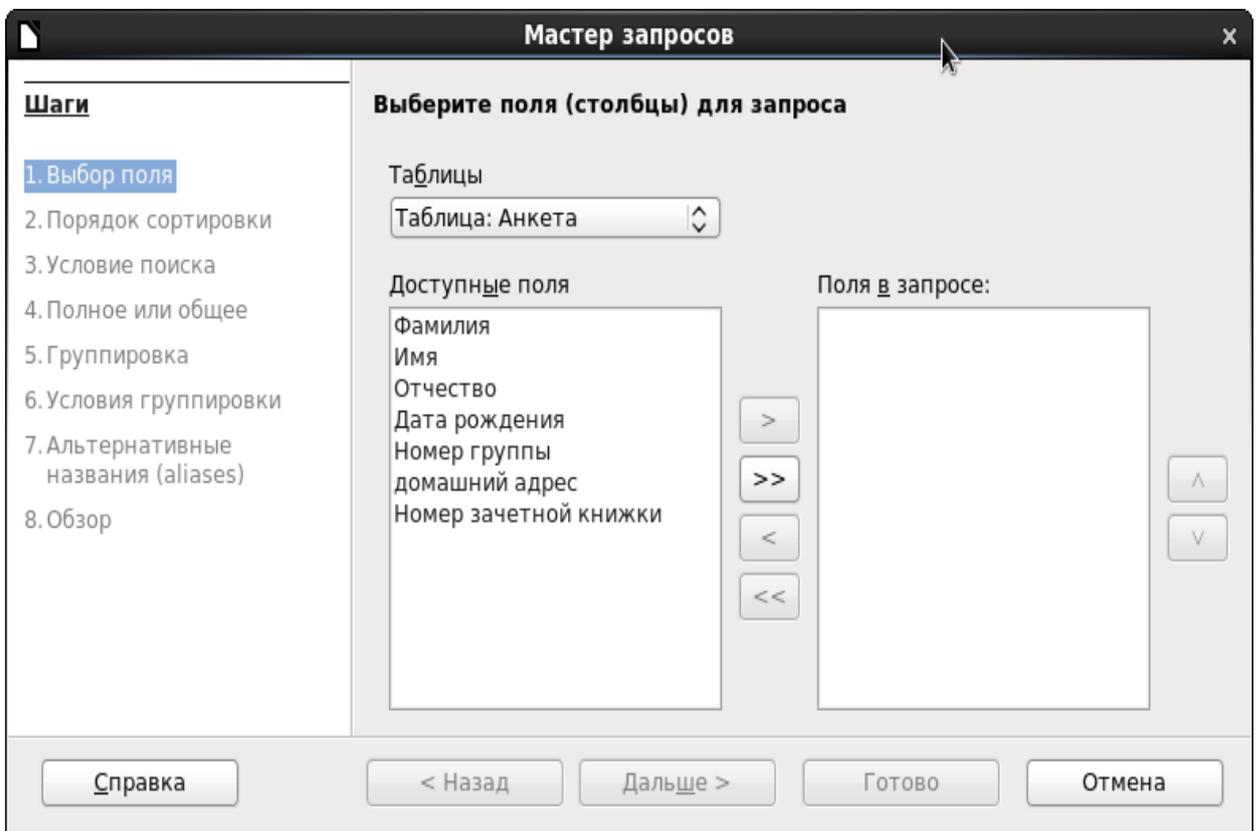


Рисунок 11.1 – Мастер запросов

3. На первом шаге указаны **Доступные поля** таблицы **Анкета**. Выберите из них щелчком по кнопке **>** или двойным щелчком по соответствующему полю следующие: **Номер группы**, **Номер зачетной книжки**, **Фамилия**, **Имя**, **Отчество**, **Дата рождения**. При этом каждое поле автоматически переносится в графу **Поля в запросе** ([рисунок 11.2](#)).

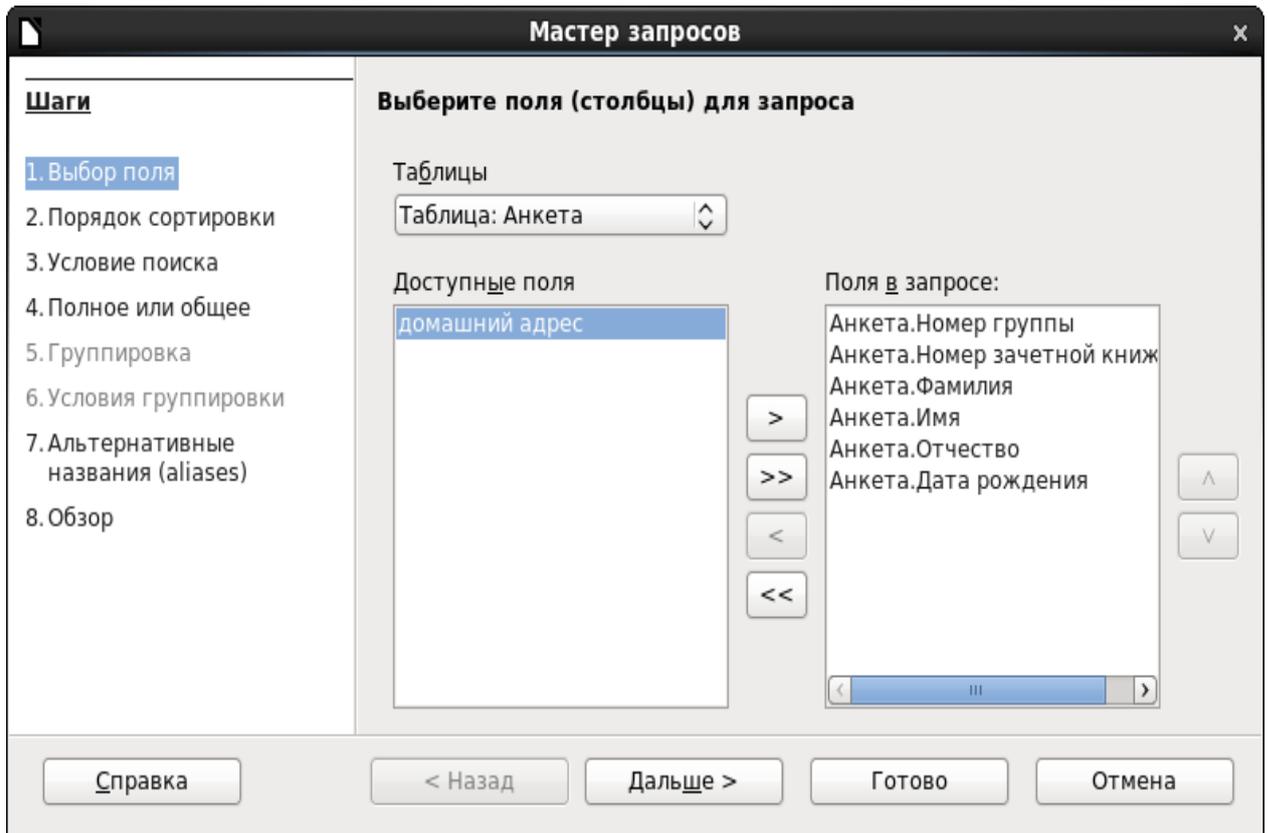


Рисунок 11.2 – Выбор полей таблицы в запрос

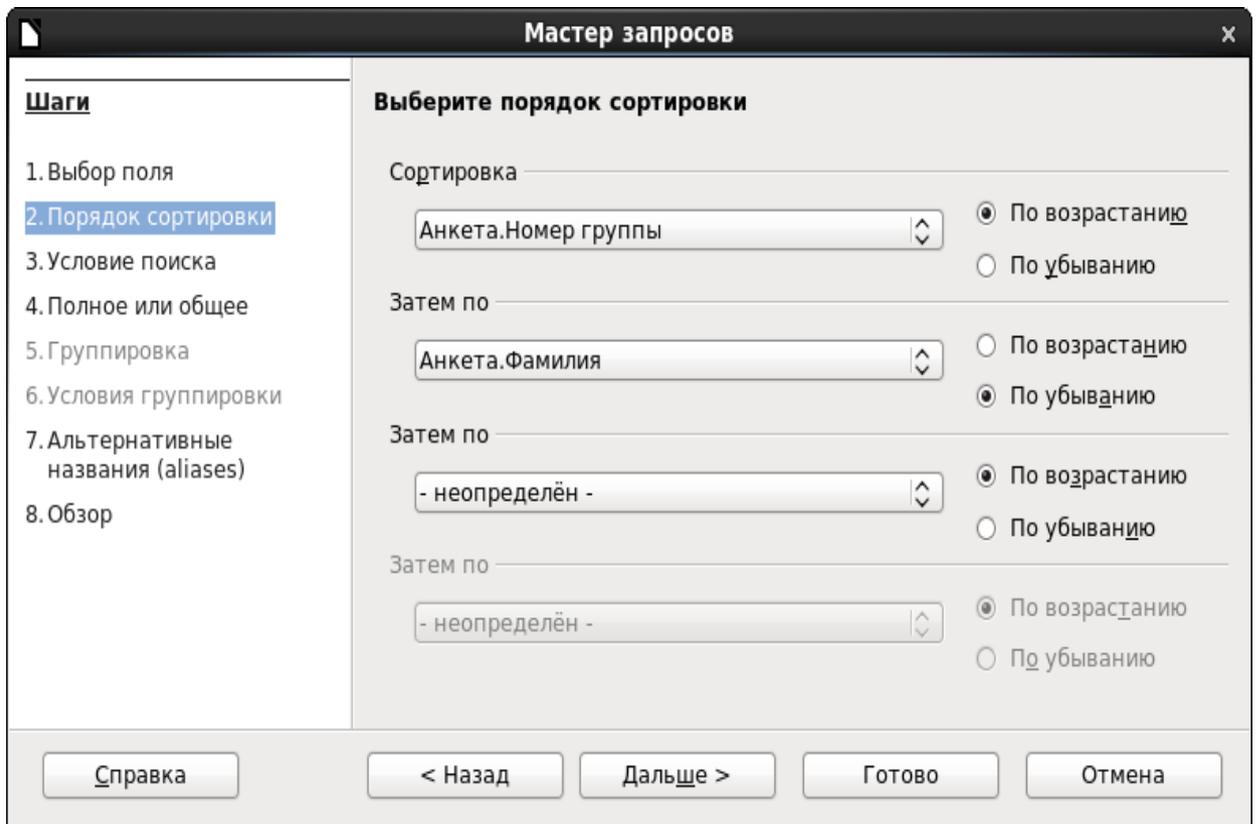


Рисунок 11.3 – Выбор порядка сортировки записей в запросе

4. После выбора полей следует щелкнуть кнопку **Дальше>**. Мастер перейдет ко второму шагу формирования запроса **Порядок сортировки**. В графе **Сортировка** из раскрывающегося списка выберите поле **Анкета.Номер группы**, **По возрастанию**; Затем по выберите **Анкета.Фамилия**, **По убыванию**. Щелкните по кнопке **Дальше>** ([рисунок 11.3](#)).

5. Третий шаг **Условие поиска**, оставьте переключатель **Соответствие всем из следующих**. Щелкните по кнопке **Дальше>**.

6. Четвёртый шаг мастера **Полное или общее**. Выберите тип запроса **Детальный запрос (показывать все записи запроса)**. Щелкните по кнопке **Дальше>**.

7. Так как мы выбрали детальный запрос пятый и шестой шаги мастера будут пропущены. На седьмом шаге можно задать полям альтернативные названия (aliases). Оставим предлагаемый мастером запросов вариант. Щелкните по кнопке **Дальше>**.

8. На восьмом шаге появляется возможность на ввод имени запроса и выбор дальнейших действий: **Показать запрос** или **Изменить запрос**. По умолчанию указано первое действие, с ним и следует согласиться. Введите имя: **ПростойЗапрос1**. Щелчком по кнопке **ГОТОВО** создание запроса завершается. На экране появляется его результирующая таблица.

9. Сохраните запрос , но не закрывайте его.

В уже готовом запросе есть возможность применить фильтр. Наиболее простой фильтр –  **Быстрый фильтр**, кнопка которого расположена на панели инструментов окон запросов и таблиц. Для этого выделяют какое-либо значение в ячейке запроса или таблицы, которое будет являться критерием отбора и нажимают кнопку **Быстрый фильтр**. После этого в результирующей таблице будут видны только данные, соответствующие выделенному значению. Для возвращения всех записей предназначена кнопка  – **Удалить фильтр**. Можно также установить **Стандартный фильтр** – , но для его использования надо знать как задаются отборочные критерии.

Гораздо более универсальным и гибким способом создания запросов является режим **Создать запрос в режиме дизайна...** Это конструктор запросов, его также называют **созданием по образцу** или с помощью **Бланка запроса**.

Теперь рассмотрим последовательность действий в режиме **Создать запрос в режиме дизайна...** Для этого:

1. В окне базы данных **Студенты** щелкните пиктограмму **Запросы**, в секции **Задачи** – команду **Создать запрос в режиме дизайна...**

2. В появившемся окне **Добавить таблицу или запрос** выберите таблицы **Анкета** и **Успеваемость** – по одной, щелкая каждый раз кнопку **Добавить**. Выбранные таблицы и связи между ними показываются маленькими окнами в верхней части **Бланка запроса**. После этого окно **Добавить таб-**

лицу или запрос можно закрыть. Система присваивает создаваемому запросу пока условное имя: **Запрос1**.

3. Главная часть **Бланка запроса** имеет следующий вид ([рисунок 11.4](#)).

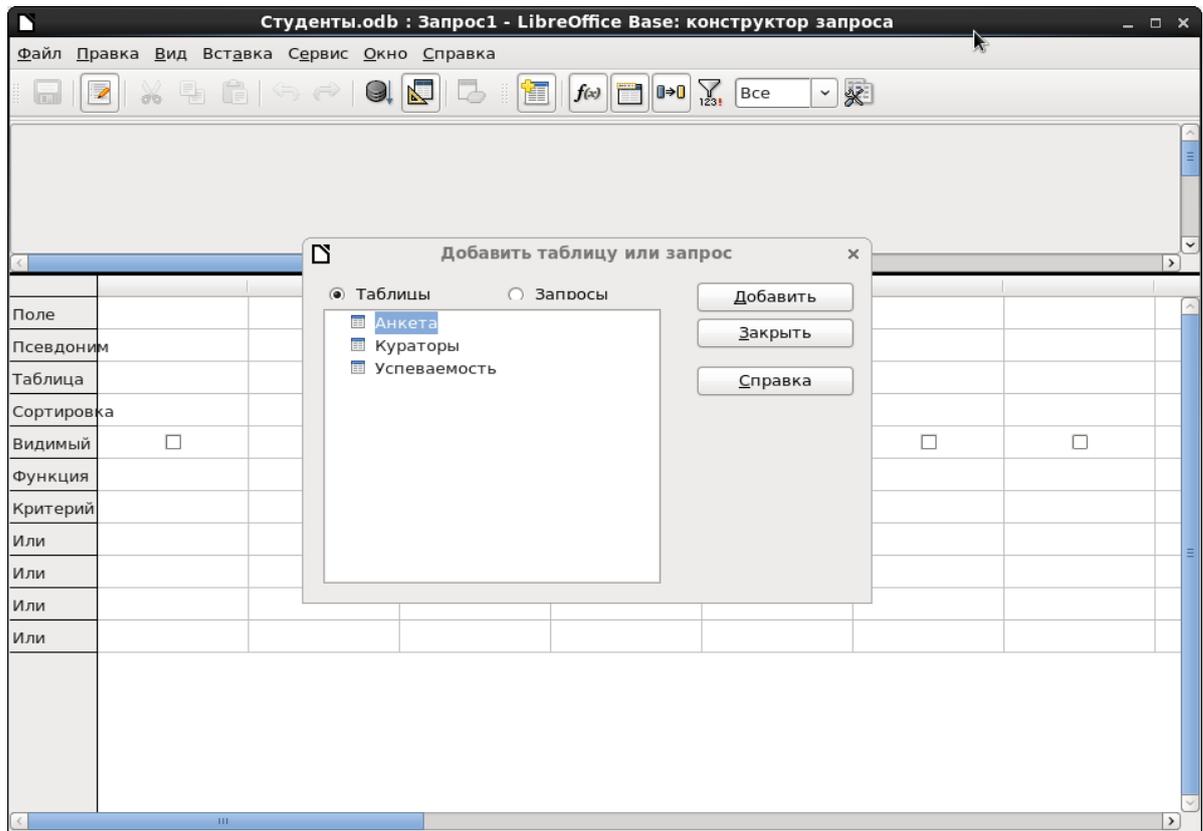


Рисунок 11.4 – Заполняемая часть «Бланка запроса»

Задача пользователя – заполнить эти строки в соответствии со структурой запроса. При этом строку **Поле** можно заполнять как с клавиатуры (вручную), так и перетаскиванием названий полей мышью из окон в верхней части **Бланка запроса**, что, конечно, более удобно. После заполнения строки **Поле** строка **Имя таблицы** заполняется автоматически, т.е. в ней появляется имя таблицы. Занесите таким образом в бланк поля **Фамилия**, **Имя**, **Отчество** из таблицы **Анкета** и поля **Инженерная графика**, **Химия**, **Теоретическая механика**, **Информатика** из таблицы **Успеваемость**.

4. В строке **Сортировка** можно выбрать опцию **По возрастанию**, **По убыванию** или **Отсутствует**. Назначение этих опций вам уже известно. Установите для поля **Фамилия** сортировку по возрастанию, что значит в алфавитном порядке.

5. После заполнения строки **Поле** в строке **Видимый** появляется знак . Он говорит о том, что данное поле будет видимым в результирующей таблице запроса. Щелчком по значку его можно убрать, и тогда поле станет невидимым, хотя его данные могут участвовать в расчетах и условиях.

6. Строка **Критерий** является одной из самых важных в запросах. Для поля **Информатика** в этой строке укажите **>3**, нажмите **Enter**. Тем самым, мы задаем условие выборки на тех студентов, которые получили по инфор-

матике оценку выше тройки ([рисунок 11.5](#)).

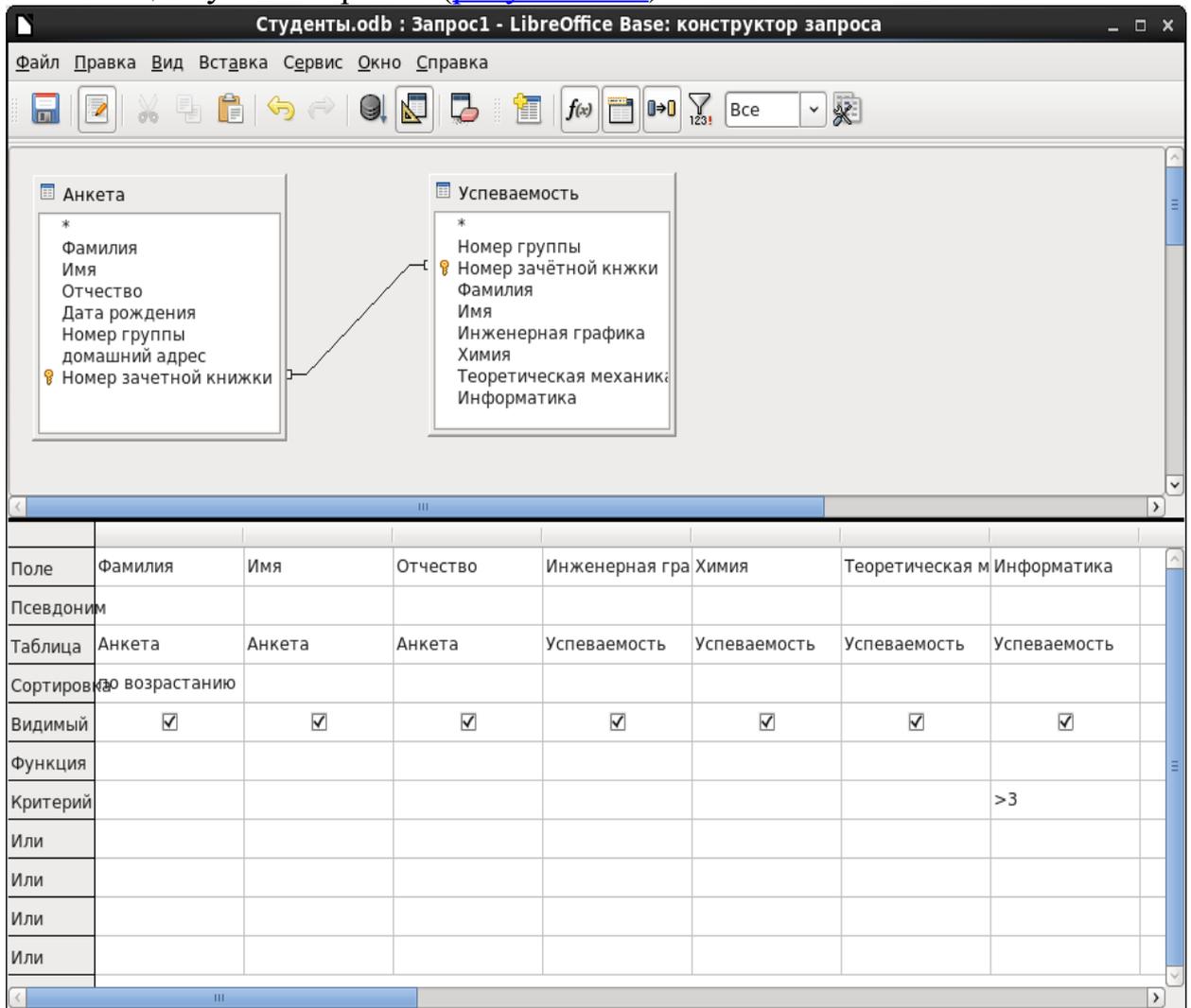


Рисунок 11.5 – Бланк запроса **Запрос по информатике**

7. Когда запрос будет окончательно готов, нужно проверить его работу, щелкнув по кнопке **Выполнить запрос** –  на панели инструментов. Результат работы запроса выводится также в верхней части окна конструктора запросов. Затем его следует сохранить. Система это делает автоматически при закрывании окна запроса щелчком по значку **X**. Появляется диалоговое окно, в котором предлагается сохранить изменения структуры или макета запроса **Запрос1**. Следует согласиться, ответив **Да**. Затем появляется окно для ввода имени запроса, введите имя **Запрос по информатике** и щелкните **Да**. После этого запрос будет сохранен в текущей базе данных **Студенты** под этим именем как один из ее объектов.

Примечания:

- не следует путать имена объектов базы данных с именем всей базы данных;
- после запуска запроса при обнаружении ошибок запрос можно исправить здесь же в конструкторе при выведенных данных, или переключив командой **Вид – Вид** в режим конструктор запросов, или снова открыв его через контекстное меню командой **Изменить**.

3.4 Для чего нужны условия

Мы познакомились с очень простым запросом на выборку. Однако на практике простая выборка применяется редко: почти всегда отбираемые данные должны соответствовать каким-то условиям. Один из способов для этого является постановка фильтров в готовом запросе. Однако фильтр **не запоминается** в структуре запроса, каждый раз его надо ставить заново. Иначе работают условия, которые записываются в двух последних строках **Бланка запроса**. Эти условия сохраняются в структуре запроса и выполняются каждый раз при его открытии.

3.5 Составление выражений условий

Запрос выполняет свою задачу по программе, записанной в **Бланке запроса**. При этом особое значение имеет строка **Критерий**, в которой записываются **выражения условий**, определяющие в конечном итоге отбор данных в результирующую таблицу. Правильные результаты работы запроса возможны только при безошибочной записи выражений условий.

Условия в запросах могут быть как сложными, так и очень простыми. Наиболее простым условием является **конкретное значение**, по которому производится отбор данных из основных таблиц. Применяются также **операции отношения** со знаками $>$, $<$, $=$, $<>$, **логические операторы NOT (НЕ), AND (И), OR (ИЛИ)**, а также **оператор параметра LIKE**. Действия, обусловленные знаками отношений и логическими операторами, соответствуют математическому смыслу этих операций: больше, меньше, равно, не равно и т.д. Некоторые примеры условий и действий, производимых ими, приводятся в [таблице 11.1](#).

Таблица 11.1 – Примеры условий в запросах

Поле	Условие	Действие
ПунктНазначения	'Ростов'	Отображает доставку товаров в пункт назначения Ростов
ПунктНазначения	'Киев' Or 'Минск'	Отображает доставку товаров в Киев и Минск
ДатаОтгрузки	#02.02.95# AND #03.02.95#	Выборка данных на товары, отгруженные 2-го и 3-го февраля 1995 года
Оценка	NOT 2 или <> 2	Вывод данных по условию оценки, не равной двойке
Оценка	>=4	Вывод данных по оценкам, большим или равным четверке (т.е. четверки и пятерки)
ФамилииСтудентов	Like 'A*'	Вывод данных на фамилии, начинающиеся с буквы А (символ * заменяет произвольный набор символов)

Примечания:

- обратите внимание на то, что текстовые данные указываются в **апострофах**, а даты в **символах #** (решетка);
- значок * означает «все»;
- одновременно со строкой **Критерий** работают строки **Или**. В них можно указывать условия, которые будут выполняться как логическое ИЛИ. Например, условие 'Киев' Or 'Минск' можно записать так: 'Киев' в строке **Критерий** и 'Минск' в строке **Или**
- оператор Like имеет следующий формат записи: **ЛИКЕ «образец»**.

Как работают эти форматы, хорошо видно из примеров [таблицы 11.1](#).

Создания запросов с условиями осуществляется в режиме **Конструктора**. Последовательность действий при этом ничем не отличается от обычных запросов на выборку. Важно только правильно записать выражения условий, исходя из требуемой логики работы запроса.

Для примера рассмотрим, как создать запрос на «двоечников» (студентов, получивших хотя бы одну двойку), а затем модифицируем его в запрос на «отличников»: студентов, получивших пятерки по всем предметам.

Выполните следующие действия:

1. В главном окне базы данных **Студенты** щелкните пиктограмму **Запросы**, затем команду **Создать запрос в режиме дизайна...**
2. В бланк запроса добавьте таблицы **Анкета** и **Успеваемость** и создайте новый запрос по [рисунку 11.6](#). Двойки вводите, начиная со строки **Критерий** и затем по строкам **Или**. Проверьте работу запроса, выполнив его запуск.

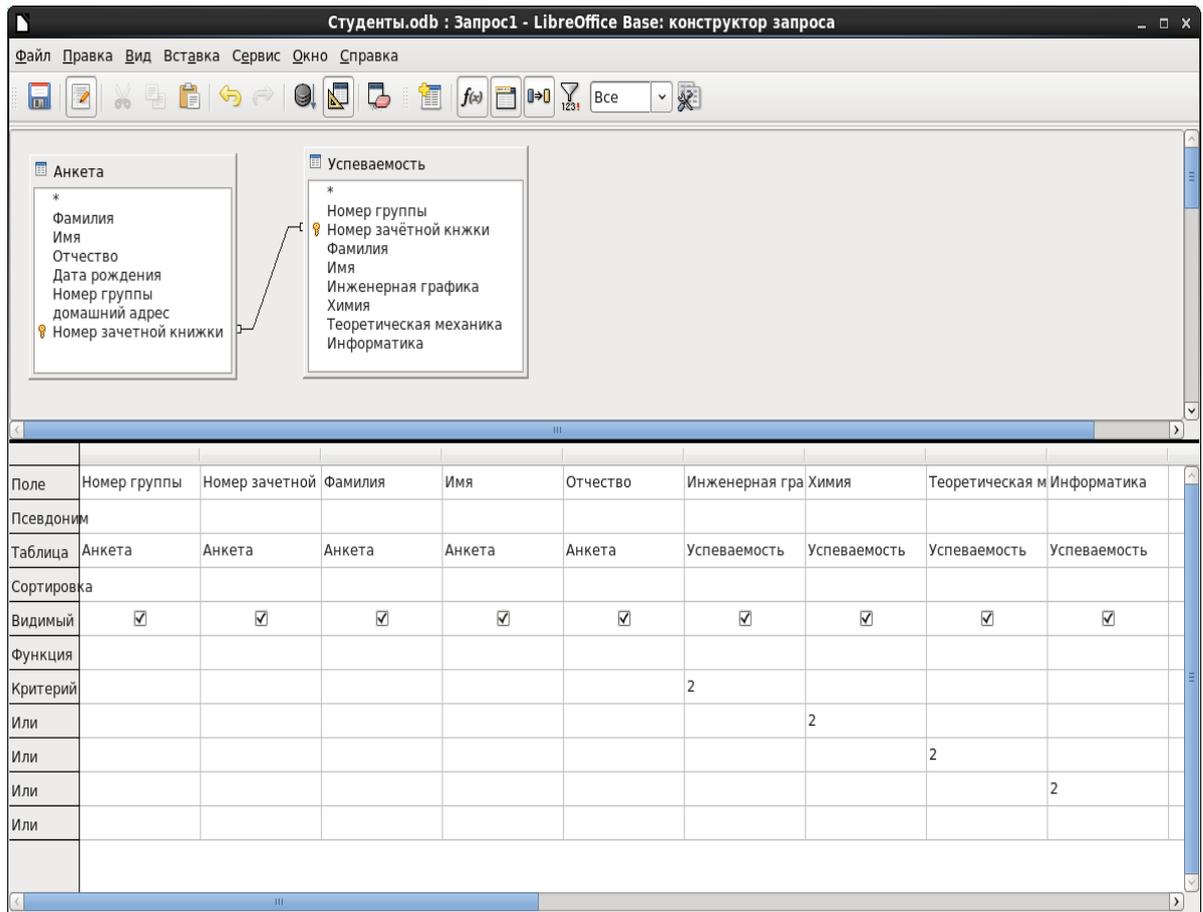


Рисунок 11.6 – Бланк запроса «Запрос на двоечников»

3. Здесь же в режиме **Конструктора** отредактируйте **Бланк запроса**, то есть удалите двойки и введите пятёрки, как показано на [рисунке 11.7](#).

4. В строке **Критерий** мы указали, что все оценки должны быть равны пятёрке. Правильным было бы и условие из одной цифры 5, а также >4 .

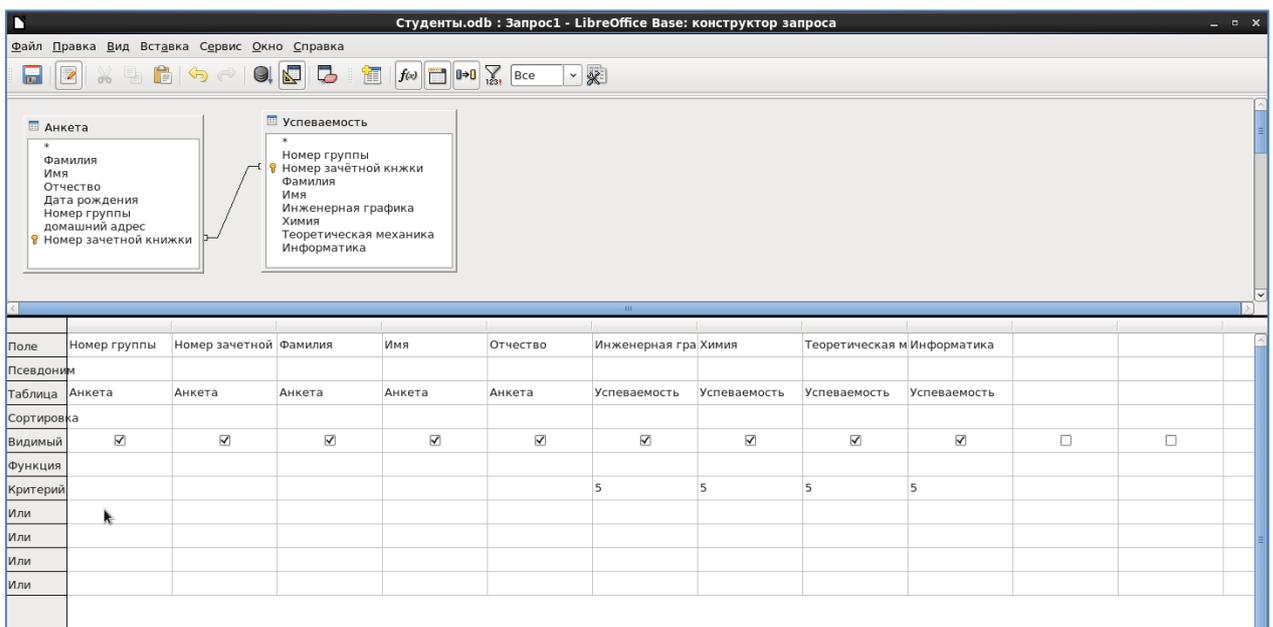


Рисунок 11.7 – Бланк запроса «Запрос на отличников»

5. Запустите запрос на выполнение и убедитесь, что запрос действительно выводит данные на отличников.

6. Для сохранения этого запроса щелкните по вкладке **Файл**, выберите команду **Сохранить как**, в появившемся окне введите новое имя запроса **Запрос на отличников** и нажмите кнопку **Да** ([рисунок 11.8](#)).

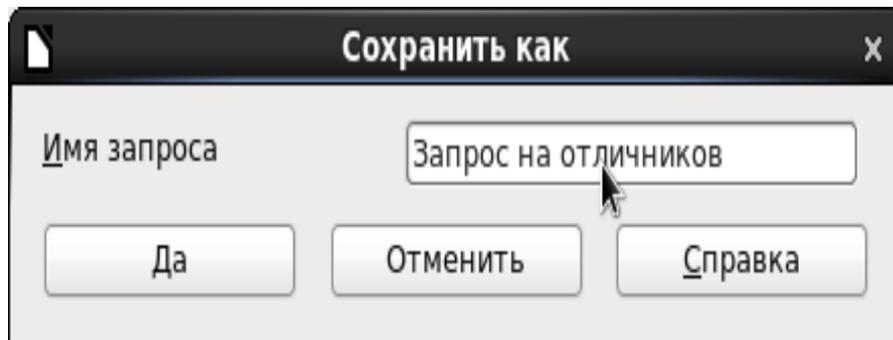


Рисунок 11.8 – Окно сохранения

Еще большими возможностями обладают запросы, которые позволяют задавать текстовые или иные значения отбора при выполнении самого запроса.

Чтобы разобраться в действии таких запросов, выполните следующие упражнения:

1. В базе данных **Студенты** проделайте начальные операции по созданию нового запроса в режиме **Создать запрос в режиме дизайна**.... Выберите таблицы **Анкета** и **Успеваемость** и поля **Фамилия**, **Имя**, **Отчество**, **Инженерная графика**, **Химия**, **Теоретическая механика** и **Информатика** из этих таблиц.

2. Поставим задачу так: запрос должен выводить оценки студентов, фамилии которых начинаются на букву «С». Из [рисунок 11.9](#) следует, что для этого потребуется оператор **LIKE** в последнем формате записи по таблице 13. Соответственно оформим **Бланк запроса**.

3. После оформления **Бланка запроса** ([рисунок 11.9](#)) запустите запрос на выполнение – . При этом появляется информация, указанная в задании, будут выведены оценки студентов, фамилии которых начинаются на букву «С».

4. Сохраните этот запрос как **Запрос по фамилиям на букву С**.

Студенты.odb : Запрос по фамилиям на букву С - LibreOffice Base: конструктор запроса

Файл Правка Вид Вставка Сервис Окно Справка

Анкета

- *
 - Фамилия
 - Имя
 - Отчество
 - Дата рождения
 - Номер группы
 - домашний адрес
 - Номер зачетной книжки

Успеваемость

- *
 - Номер группы
 - Номер зачётной книжки
 - Фамилия
 - Имя
 - Инженерная графика
 - Химия
 - Теоретическая механика
 - Информатика

Поле	Фамилия	Имя	Отчество	Инженерная гра	Химия	Теоретическая м	Информатика
Псевдоним							
Таблица	Анкета	Анкета	Анкета	Успеваемость	Успеваемость	Успеваемость	Успеваемость
Сортировка							
Видимый	<input checked="" type="checkbox"/>						
Функция							
Критерий	LIKE 'С*'						
Или							
Или							
Или							
Или							

Рисунок 11.9 – Оформление «Бланка запроса» с оператором LIKE

4 ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

1. Создайте **Простой запрос на оценки** так, чтобы в его таблице остались видимыми только фамилии и оценки по информатике, причем оценки по информатике были отсортированы в убывающем порядке. Сохраните запрос, откройте его в режиме **Таблицы** и убедитесь в правильном виде результирующей таблицы.

2. В режиме дизайна создайте новый запрос под названием **Двоечники по информатике и химии** для вывода фамилий и **ВСЕХ** оценок студентов, которые получили двойки одновременно по **Информатике** и по **Химии**. Проверьте работу запроса.

3. Создайте новый запрос с именем **Группа МП12** в котором выведите фамилии, имена отчества и оценки студентов обучающихся в этой группе и фамилию куратора для группы **МП-12**.

4. В таблице **Анкета** в режиме **Конструктор** создайте новое поле **Город**, куда занесите название населенных пункт постоянного проживания студентов, взяв его из поля **Домашний адрес**.

5. В режиме дизайна создайте запрос под именем **Запрос по городу**, выводящий фамилию, имя, отчество и полные адреса студентов, проживающих в одном из двух населенных пунктов (например, Сальске или Зернограде).

6. Создайте запрос под именем **Запрос по дате**, выводящий фамилию, имя, отчество и дату рождения студентов, родившихся в первой половине 2002 г.

5 КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какие виды запросов существуют?
2. В чем заключается основной принцип работы запросов?
3. Какие способы создания запросов вы знаете?
4. Основные режимы просмотра запросов?
5. Что может входить в условие отбора?
6. Каково назначение оператора Like?
7. Как получить информацию запросом по почтовому индексу 347740?

ЛАБОРАТОРНАЯ РАБОТА № 12

Тема: **ВЫЧИСЛЯЕМЫЕ ЗАПРОСЫ И СОЗДАНИЕ ФОРМ**

1 ЦЕЛЬ РАБОТЫ

Познакомиться:

- со структурой, видами и порядком создания форм.

Научиться:

- создавать вычисляемые поля в запросах;
- применять группировки и статистические функции;
- создавать формы.

2 ОБЪЁМ ЛАБОРАТОРНОЙ РАБОТЫ И ПОРЯДОК ЕЁ ВЫПОЛНЕНИЯ

Данная лабораторная работа выполняется в течение двух часов. Изучение теоретического материала идёт одновременно с выполнением практических заданий, для закрепления которых нужно выполнить упражнения для самостоятельной работы. Результаты выполнения лабораторной работы необходимо сохранить в указанной преподавателем папке.

3 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ. ТЕХНОЛОГИЯ ВЫПОЛНЕНИЯ РАБОТЫ

3.1 Создание вычисляемых полей

Назначение СУБД заключается не только в хранении информации, но и в обработке ее по нужным для пользователя законам. Для этого в LibreOffice Base предусмотрен набор математических и статистических функций, позволяющих производить любые вычисления. Следует иметь в виду, что **вычисления никогда не производятся в основных таблицах, для этого есть другие объекты LibreOffice Base, в частности, запросы.** Именно в запросах можно найти значение, рассчитанное по одному или нескольким полям, вычислить сумму, среднее значение, определить количество записей, получить новую дату, отстоящую от текущей на определенное число дней, и т.п. При этом действует главный принцип работы запросов: **результаты расчетов не запоминаются.** Каждый раз запрос заново производит вычисления, предоставляя пользователю результат во **временной результирующей таблице.**

Существует два способа вычислений в запросах:

- с помощью **вычисляемых полей;**
- с помощью **групповых операций и статистических функций.**

Рассмотрим первый способ. Как мы уже знаем, обычные запросы содержат поля, выбранные из основных таблиц. **Вычисляемые поля в запросах – особые, их нет в основной таблице.** Пользователь сам создает вычис-

ляемые поля путем записи **расчетных выражений** в Бланке запроса.

Выражением в LibreOffice Base называется комбинация имен полей, констант и функций, соединенных знаками математических операций +, -, *, /, ^ . При открытии запроса выражения (если они не содержат ошибок) сразу вычисляются и пользователь видит уже готовый результат в соответствующем поле.

Рассмотрим создание **вычисляемого поля** на примере.

1. Откройте учебную базу данных **Студенты**, щелкните по пиктограмме **Запросы**, затем по команде **Создать запрос в режиме дизайна...**, в появившемся окне **Добавить таблицы** выберите таблицу **Успеваемость** щелкните по кнопке **Добавить** и закройте окно **Добавление таблицы** (бланк запроса не закрывайте).

2. В бланк запроса вынесите следующие поля: **Фамилия**, **Инженерная графика**, **Химия**, **Теоретическая механика** и **Информатика**, установите курсор в свободном столбце **Бланка запроса** в строке **Поле** и наберите следующее

("Химия"+"Информатика"+"Инженерная графика"+"Теоретическая механика")/4
([рисунок 12.1](#)).

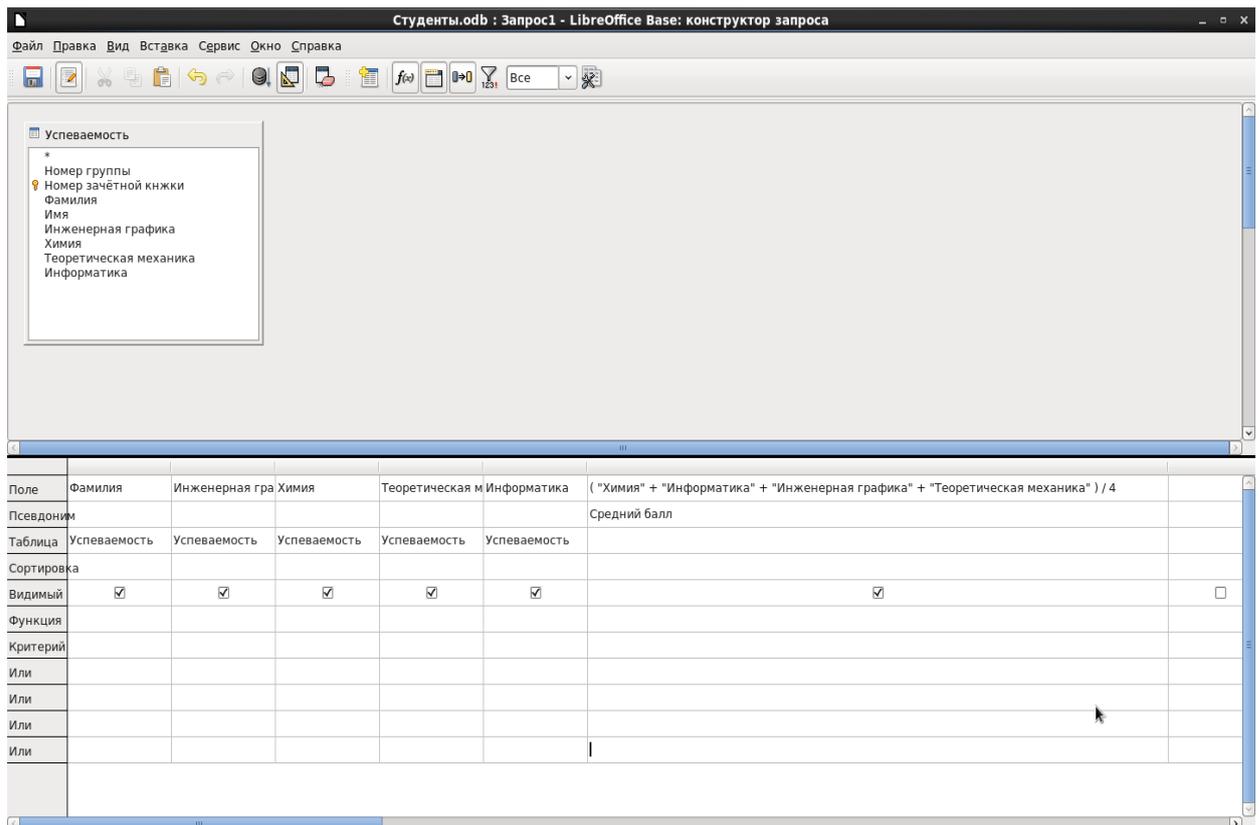


Рисунок 12.1 – Бланк запроса

При наборе выражений следует строго соблюдать правило: имена полей, входящие в выражение, указываются в кавычках.

Ниже в строке **Псевдоним** введите **Средний балл**.

3. Запустите запрос, щелкнув по кнопке **Выполнить запрос**. При выполнении запроса новый столбец будет называться **Средний балл**, в нем будет подсчитан средний балл успеваемости по каждому студенту.
4. Сохраните запрос с именем **Средний балл по студенту** и закройте его.

3.2 Групповые вычисления в запросах

Кроме «одиночных» вычислений в LibreOffice Base предусмотрена возможность одновременной обработки множества записей. Такие действия называются **групповыми операциями** и выполняются с помощью **статистических функций**.

Групповые операции отличаются особой логикой выполнения: **записи вначале группируются – разбиваются на группы по какому-либо признаку – а затем уже над группами выполняются конкретные действия статистическими функциями**. При этом поля группировки и расчетов никогда не совпадают. Ключом, по которому разбиваются записи, может быть, например, пол студентов (мужской или женский), а действием – расчет количества студентов того и другого пола.

Создание запроса с групповыми операциями выполняется в режиме дизайнера по общепринятой методике. Отличия заключаются в выборе полей: в **Бланк запроса** следует вносить **только поля группировки и расчетов**. В частности, для указанного выше примера таковыми являются: поле **Пол** для группировки и поле **Фамилия** для расчета количества студентов. Выполните следующие упражнения:

1. Откройте таблицу **Анкета** в режиме **Конструктора** и добавьте новое текстовое поле **Пол** длиной три знака. Сохраните изменения структуры запроса. Затем заполните поле **Пол** соответствующими данными (**муж** или **жен**).

2. Перейдите к созданию нового запроса в режиме **Создать запрос в режиме дизайнера...**. Добавьте таблицу **Анкета**, а из нее выберите поля **Пол** и **Фамилия**.

3. В строке запроса **Функция** поля **Фамилия** появится стрелка раскрывающегося списка ▼. Щелчком мыши по этой строке и далее по стрелке вниз раскрывается меню групповых операций. Выберите из списка для поля **Фамилия** функцию **Количество**, для поля **Пол** – функцию **Group** ([рисунок 12.2](#)). Для поля **Фамилия** введите **Псевдоним – Количество студентов**.

4. Обратите внимание на то, что для поля **Пол** выбрана функция **Group**, т.е. по этому полю записи будут разбиваться на группы. Для поля **Фамилия** выбрана функция **Количество**, подсчитывающая количество студентов. Когда вы оформите **Бланк запроса** таким образом, записи будут разгруппированы по полу и подсчитаны по фамилиям, то есть мы узнаем сколько мужчин и женщин содержится в основной таблице базы данных.

5. Запустите запрос на выполнение и проверьте результирующую таблицу запроса.

6. Сохраните запрос под именем **Запрос по количеству студентов**.

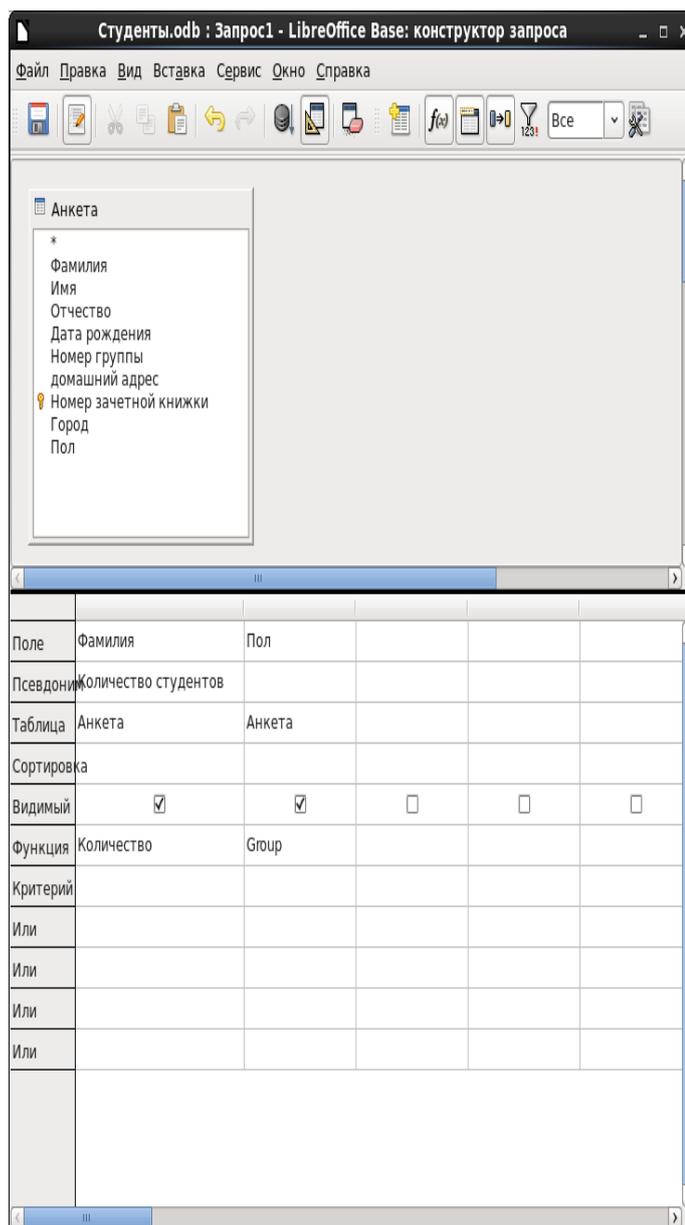


Рисунок 12.2– Выбор групповой операции

Принцип групповых вычислений также лежит в основе так называемых **Итоговых запросов**. Рассмотренный выше запрос является итоговым, однако на практике применяются гораздо более сложные итоговые запросы, в которых выполняется не одно, а несколько групповых действий. При этом для одного группируемого поля вычисляются несколько расчетных полей, где определяется сумма, среднее значение, среднеквадратичное отклонение и т.п.

Такие расчеты называются подведением **итогов**. При подведении итогов для каждого расчетного поля выбирается своя статистическая функция. Назначение основных статистических функций приводится ниже:

- Сумма;
- Среднее;