**Лабораторная работа №4**

**Тема**: Вычислительные поля в запросах. Добавление, обновления, удаления информации из таблиц БД с помощью QBE-запросов. Параметрические запросы, перекрестные запросы.

**Цель**: Научиться создавать запросы с вычислительными полями. Изучить правила построения выражений. Научиться строить запросы на создание таблицы, на добавление информации из другой таблицы, на изъятие записей, на обновление информации в БД, перекрестные запросы и запросы с параметрами с помощью QBE-запросов.

**Содержание работы по варианту индивидуального задания:**

1. Создайте запрос к таблице вашей БД, который в последнем столбце динамической таблице содержал бы информацию из предыдущих двух полей, разделенную знаками препинания и вспомогательным текстом. Например: Альфа-Львов, тел .: 77-89-54.

2. Создайте запрос, с помощью которого можно было бы высчитывать данные таким образом, первую строчку прилагался ко второму и размножался на третий. 25% от полученного разместить в одной из колонок динамической таблице.

3. Создайте запрос с вычислительным полем Сообщение. Это поле будет содержать сообщение: «Заканчивается срок действия» если значение в поле Срок действия от 10 до 0 и сообщение «Срок действия окончен», если срок действия - 0.

4. Создайте выражение в запросе, который заменяет пустые записи в таблице на 0 или на «Пусто».

5. С помощью QBE-запроса добавьте записи с одной таблицы БД к записям другой таблицы БД.

6. Удалите из таблицы БД записи, соответствующие некоторым критериям с помощью QBE-запросов.

7. Заменить в таблице БД значения, отобранные на основе определенных критериев на новые, которые будут результатом выполнения выражения.

8. С помощью QBE-запроса создайте «архивную» копию записей таблицы БД.

9. Создайте запрос, который спрашивает значение критерия, по которым будет осуществляться выбор информации.

10. Создайте перекрестный запрос.

**Теоретические сведения:**

Вычислительные поля позволяют создавать и показывать выражения на основе имеющихся полей. Выражения является главным средством выполнения многих операций Microsoft Access. Выражение представляет комбинацию символов-идентификаторов, операторов и значений, которая дает определенный результат.

Вот несколько примеров, в которых удобно использовать выражения.

• Комбинирование полей Фамилия и Имя, разделенных запятыми, в выражение с целью показа их на экране, например, в таком виде: **Баркер, Диана.**

• Добавление дней, недель или месяцев до текущей даты с целью вывода на экран будущей даты.

• Перемножение значений полей, таких как количество и Цена, чтобы вывести на экран общую стоимость.

Access может образовывать вычислительные поля с других полей, буквальных значений и функций. Перед тем как заняться построением выражений, давайте ознакомимся с их синтаксисом, принятым в Access.

**Синтаксис вычислительных полей**

Для вычислительных полей требуется использовать определенный синтаксис. Например, для комбинации полей Фамилия, Имя и запятая, синтаксис будет таким: Выражение1: [Фамилия] & "," & [Имя].

Синтаксис определяет структуру команды, которую вы используете. В вычислительных полях определяются стандартные операторы и их корректное применение в соответствующие поля.

Первая часть приведенного вычислительного поля Выражение1, является именем нового поля. Access предложит вам его, если этого не сделаете вы сами. Именно на это имя вам нужно будет ссылаться в дальнейшем при использовании этого поля в форме, отчете или другом выражении.

**Заключение полей в скобки**

Используя в запросах поля из таблиц, вы должны заключать эти поля в квадратные скобки. Если в названии поля не будет проблем, Access поставит скобки за вас. В этом отношении [Фамилия] и [Имя] будут хорошими примерами.

**Использование операторов в вычислительных полях**

Есть целый ряд операторов, которые можно использовать в вычислительных полях. Некоторые из них предназначены для математических вычислений, как, например, приведены в таблице.

**Таблица: Математические операторы для вычислительных полей**

|  |  |
| --- | --- |
| **Операторы** | Описание |
| **+** | Сложение |
| **−** | Вычитание |
| **\*** | Умножение |
| **/** | Деление |
| **^** | n-я степень |

Примером применения математического оператора может служить создание нового поля для общей стоимости путем умножения полей Количество и Цена. Это может выглядеть так: общая стоимость:

[Количество] \* [Цена].

Вы уже видели еще один оператор - амперсанд (&). Он был использован для того, чтобы соединить две строки с буквальной константой в один новую строку:

Выражение1: [Фамилия] & "," & [Имя].

Другое популярное вычислительное поле соединяет поля Город Область и ZIP - код с запятой в одно поле. Синтаксис для него может быть таким:

**Адрес: [Город] & "," & [Область] & "" & [ZlP-код]**

 а результат, например, таким: Александрия, КР 0246207.

В этом примере с адресом мы добавили пробелы и запятые между названиями города, области и ZIP - кодом, поместив их в кавычки и, используя их как буквальные константы.

**Примеры вычислительных полей**

Следующий список содержит еще несколько вычислительных полей, часто могут быть полезными:

**Стоимость: [Цена] \* [Количество]**

Общая стоимость: [Стоимость] + ([Стоимость] \* [ПодатковийКоефициент])

З0дней: [ДатаДоговора] +30

КодГорода Left ([Телефон] 3)

Не все поля, используемые в вычислительном поле, должны быть использованы в бланке запроса. Если поле представленное в списке полей области конструктора запроса, вы уже можете использовать его в бланке запроса. Не имеет значения и то, в каком порядке находятся поля или вычислительные поля.

**Создание простого вычислительного поля**

Для того чтобы показать, как создавать вычислительные поля, нам нужно создать запрос, с которым мы будем работать.

Создайте новый запрос, выбрав поля Фамилия и Имя из списка полей таблицы тблДолжникы (рис.1).

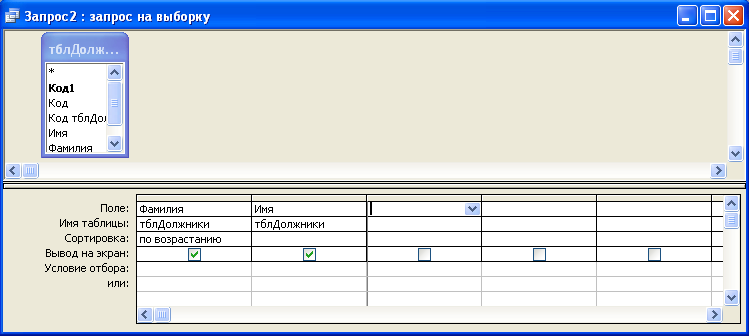


рисунок 1

Определите порядок сортировки по **возростанию** для поля **Фамилия**.

Теперь нажмите кнопку **Вид**, чтобы увидеть результат в виде таблицы (рис. 2).

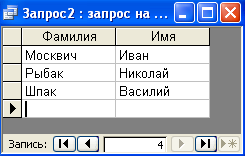
****

рисунок 2

Теперь мы можем продолжить работу с вычислительными полями.

Одним из самых простых и самых полезных примеров вычислительного поля слияния двух текстовых полей. Полезных том, что часто, например, вам нужно взять фамилии и имена и связать их каким-то образом на почтовых ярлыках, в отчетах или формах.

Перейдите в режим конструктора запроса. Затем в бланке запроса поместите курсор в пустой столбик рядом с полем **Фамилия**, справа от него.

Нажмите <Shift + F2>. Появится диалоговое окно **Область** **ввода**, как на рис.3.

Также для создания выражений в запросах используют **Построитель** **выражений**. Для этого, находясь в ячейке, где надо создать выражение, вызовем Построитель с помощью кнопки Построить панели инструментов.

Средство **Область** **ввода** является отличной возможностью для внедрения информации, когда вам хотелось бы видеть все поле.

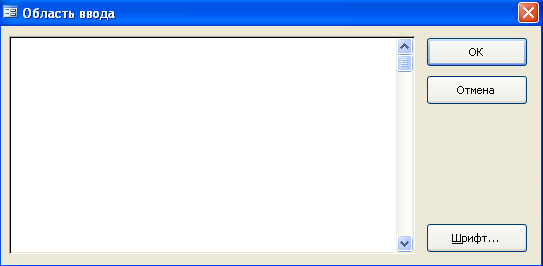
****

рисунок 3

Наберите **Фамилия** **&** "," **& Имя**, а затем нажмите ОК. Сразу же после этого Access преобразует введенное вами в **Выражение 1: [Фамилия] & "," & [Имя].**

Access создаст имя **Выражение1** (рис. 4).

В результате получим динамическую таблицу, где появилось новое поле **Выражение1**.

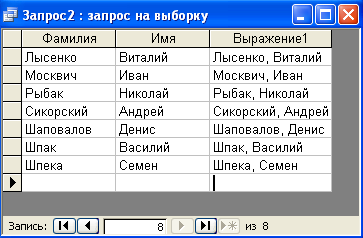
****

рисунок 4

Имя вычислительных полей можно менять. К примеру:

ПолноеИмя: [Фамилия] & "," & [Имя].

Определение свойства Подпись вычислительного поля

Как и в случае с полями таблицы, вы можете установить свойство **Подпись** для любого поля в запросе, чтобы на экран выводился текст, отличный от названия поля. Это очень удобно для вычислительных полей, когда вы можете выводить на экран слова с пробелами между ними и не иметь пробелов в названии поля.

Чтобы установить свойство **Подпись** вычислительного поля **ПолноеИмя**, что было вами создано, выполните следующее:

Поместите курсор в столбик **ПолноеИмя** запроса.

Выберите **Свойства** из меню **Вид**. Вы увидите окно свойств, подобное окну свойств текстового поля.

В строке свойства **Подпись** напечатайте **Полное** **имя** (рис.5).

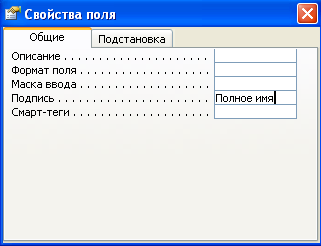
****

рисунок 5

Использование встроенных функций Access в вычислительных полях

Функции Access значительно расширяют ваши возможности по отображению данных. С помощью этих встроенных функций вы можете следующее:

• Выводить на экран не все, а только часть данных.

• Комбинировать данные с новой информацией, полученной с помощью функции Access.

• Комбинировать функции Access и выводить вполне новую информацию.

Среди примеров вычислительных полей, приводились выше, было поле, которое называется **КодГорода.** Для него было использовано выражение Left ([Телефон] 3). Оно попадает в первую категорию типов информации, предоставляемых функциями Access.

Функция Left () является встроенной функцией Access, подобно функции Date (). Функция Left () вместе с двумя своими аргументами, [Телефон] и 3, говорит Access, нужно взять три крайних слева символы поля Телефон и вернуть их в качестве результата.

Иногда функции требуют аргументов. В данном примере текущими аргументами являются поле [Телефон] и 3.

Еще одним примером использования функции Access является получение инициалов от имени и фамилии. Это тоже может быть выполнено с помощью функции Left (). Вот соответствующее вычислительное поле:

Инициалы Left ([Имя], 1) & Left ([Фамилия], 1)

**Обработка пустых значений в запросах**

Одна из проблем, возникающая при использовании вычислительного поля, порождается тем фактом, что поля могут содержать пустые значения. Пустое значение будет распространяться на все последующие результаты запроса.

Пустое значение (значение Null) присутствует в поле, когда в поле не вводилось никаких значений и не было никакого значения по умолчанию. Когда поле содержит пустое значение, это означает, что его значение неизвестно, и Access не знает, что в этом случае делать.

Чтобы разобраться с пустыми значениями, создадим временную таблицу, содержащую необходимые поля. В данном случае это поля Количество и Цена. Посмотрите на рис. 7, где представлен рабочий бланк запроса с полями Количество и Цена, а также с вычислительным полем ОбшаяСтоимость. Этот запрос никак обрабатывает пустые значения. На рис. 6 представлен результат в случае, когда пустые значения не обрабатываются.

Чтобы быть готовым к пустым значений, в Access есть функции, такие как IsNull () или Nz (). Синтаксис функции IsNull () такой: IsNull (выражение)

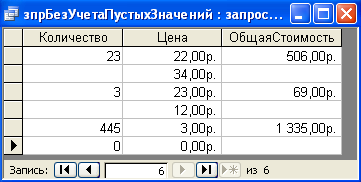


рисунок 6

Выражение может быть полем или вычислительным значением. Функция IsNull () вернет или Истина, или Ложь, в зависимости от того, является значение выражения пустым или нет. Предполагается, что это должно использоваться с функцией IIF (). Синтаксис функции IIF () такой: **IIF (критерий; значение1; значение2)**

При вызове функции ИIF (), если критерий дает Истина, то функция IIF () возвращает значение1, если критерий дает Ложь, то функция **IIF () возвращает значение2.**

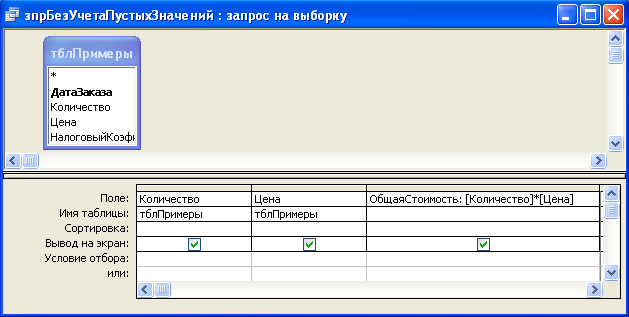
****

рисунок 7

Чтобы исправить запрос, показанный на рис.6, мы создадим вычислительное поле под названием ВерноеКво. Будет использована функция IIF () со значением IsNull (Количество) в качестве критерия (значение принимающей или Истина, или Ложь) Вот нужное выражение:

ВерноеКво: IIF (IsNull ([Количество]), 0, [Количество])

Если функция IsNull () (критерий) возвращает Истина, то вместо Количество используется 0 (значение1). Если IsNull () возвращает Ложь, то используется поле Количество (значение2). Вы можете увидеть это вычислительное поле на рис.8, где запрос подан в режиме конструктора запроса; результирующий письмо данным показан на рис.9.

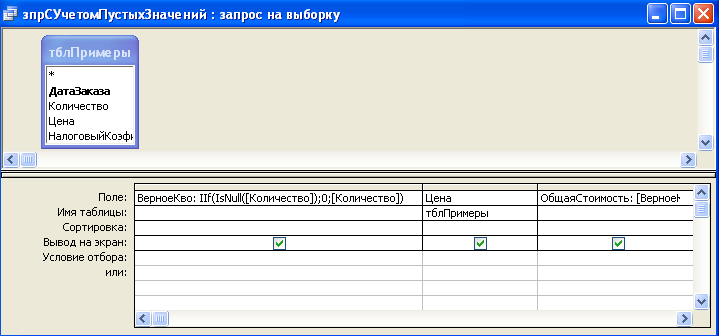
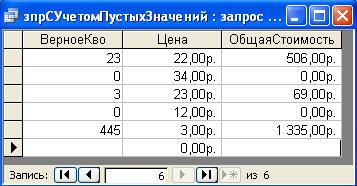
****

рисунок 8

Пустые значения теперь заменены нулями.

****рисунок 9

Обратите внимание, что вычислительное поле ОбщаяСтоимость теперь использует вычислительную поле ВерноеКво вместо поля Количество.

**Запросы других типов**

Мы уже рассмотрели запрос на выборку. В Access существуют и запросы других типов: запросы действия (в смену), параметрические запросы и перекрестные запросы.

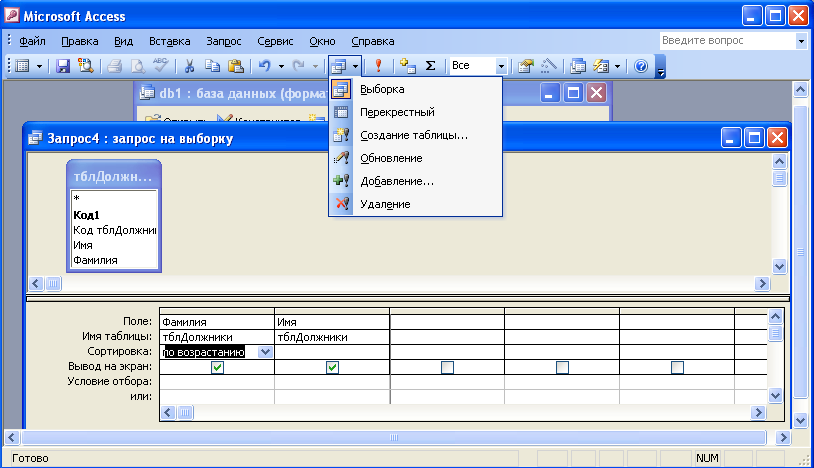


рисунок 10

**Запросы на изменение**

Запросы на изменение используют для изменения и переноса данных в таблицах, для обновления, добавления и удаления групп записей, а также для создания новой таблицы на основе одной или нескольких таблиц.

Различают четыре типа запросов на изменение: запрос на добавление, запрос на удаление, запрос на возобновление и запрос на создание таблицы.

**Запрос на добавление**

С помощью запроса на добавление записи из одной таблицы (все или отобранные запросу) можно поместить в конец другой таблицы. Обе таблицы могут находиться как в одной, так и в различных базах данных.

Если необходимо добавить записи в таблице из другой базы данных, сначала следует присоединить таблицу-источник в базу, содержащую целевую таблицу, с помощью команды Связать таблицы с подменю Получить внешние данные меню Файл. Для отбора записей, добавляют нужно составить запрос на выборку. Затем следует выполнить составленный запрос и оценить результат, переключившись в режим таблицы с помощью команды Datasheet View в меню Вид. После этого необходимо вернуться в режим конструктора и активизировать команду Запрос добавления меню Запрос.

В диалоговом окне при этом откроется Append пользователь должен задать в поле Имя таблицы имя таблицы, к которой он хочет присоединить данные из результирующего набора записей запроса. Закрытие диалогового окна с помощью кнопки ОК приводит к тому, что Access добавляет в бланк запроса строку Append To. В эту строку автоматически (или вручную) вставляются имена полей целевой таблицы, совпадающие с именами полей запроса. Далее следует выполнить запрос, нажав кнопку Run на панели инструментов. В специальном диалоговом окне Access укажет, сколько записей будет добавлено к целевой таблицы, и предложит подтвердить выполнение этой операции.

**Запрос на удаление**

Запросы этого типа служат для удаления из таблицы групп записей, соответствующих некоторому критерию отбора. Поскольку записи удалены с помощью запроса, нельзя восстановить, стоит тщательно анализировать критерии отбора.

Итак, прежде всего необходимо разработать запрос на выборку удаляемых записей, указав критерии в строке Критерии, выполнить этот запрос, чтобы проверить корректность заданных критериев, и вернуться в режим конструктора.

После этого следует активизировать в меню Запрос команду Удаление запроса. Access добавит в бланк запроса строку Удалить и введет в ее ячейки значение условия. Это означает, что пользователь может установить дополнительные критерии отбора. Далее следует выполнить запрос, нажав кнопку с изображением восклицательного знака на панели инструментов. В специальном диалоговом окне Access укажет, сколько записей удалится из таблицы, и предложит подтвердить удаление.

**Запрос на восстановление**

Используя запрос на возобновление, пользователь может изменить группу записей, отобранных на основе определенных критериев. При разработке запроса на восстановление прежде всего создается и проверяется запрос на выборку. После этого в режиме конструктора активизируется команда Запрос на обновление меню Запрос, в результате чего Access добавляет в бланк запроса строку Update To, предназначенный для указания новых значений полей таблицы. В качестве типичных могут выступать и вычислительные выражения. После нажатия кнопки Run Access укажет в специальном диалоговом окне, сколько записей изменится в таблице, и попытается подтвердить изменения.

**Запрос на создание таблицы**

На основе результирующего набора записей запроса можно построить новую таблицу с помощью запроса на создание таблицы. Такие запросы обычно применяют для архивации старых записей или для хранения резервных копий таблиц.

Сначала необходимо подготовить запрос на выборку и получить динамический набор данных Recordset, заодно проверив правильность запроса. Если результирующий набор записей соответствует вашим требованиям, вернитесь в режим конструктора и в меню Запрос выберите команду Запрос создания таблицы. Access откроет диалоговое окно Создание таблицы, в которое следует ввести имя новой таблицы. Далее нужно выполнить запрос, нажав кнопку со знаком на панели инструментов. В специальном диалоговом окне Access укажет, сколько записей добавляется в новую таблицу, и предложит подтвердить выполнение этой операции.

**параметрические запросы**

Запросы, представляющие собой варианты базового запроса и незначительно отличаются друг от друга, называются параметрическими.

Рассмотрим создание параметрического запроса для отбора из таблицы Адреса фамилии лиц, проживающих в указанном городе (поле Населенный пункт) и по определенному адресу (поле адрес).

Прежде всего разработаем запрос на выборку данных. Включим в него поля Фамилия, Населенный пункт и адрес. Поля строки Критерии заполните не конкретно критериям отбора, а обращением к пользователю для введения критерия. На основе этого критерия (с учетом других критериев, заданных в спецификации запроса) будет осуществляться отбор записей. Обращение необходимо взять в квадратные скобки, например: [Назовите город], [Укажите адрес].

Активизируйте команду Параметры меню Запрос. В поле Параметры диалогового окна Параметрические запросы введите текст, находящийся в поле Критерии спецификации запроса. В данном случае скобки не нужны. После определения типа данных в поле Тип данных и нажатия кнопки ОК запрос будет готов. Теперь можете приступить к вводу параметров.

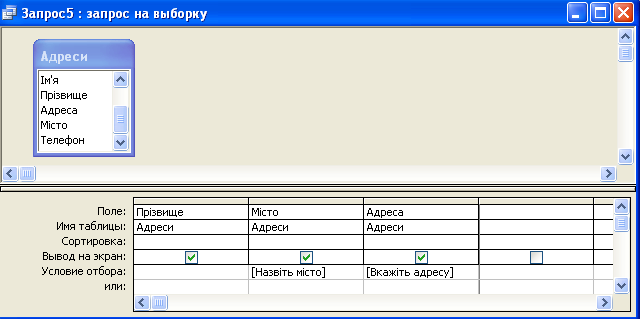
****

рисунок 11

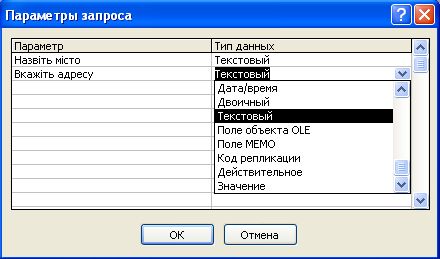


рисунок 12

Для проверки работоспособности запроса перейдите в режим таблицы, установив опцию Datasheet View в меню Вид. При этом Access обратится к вам с просьбой задать значение критерия в окне Введите значение параметра.

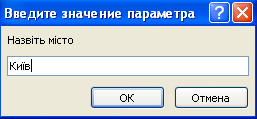


рисунок 13

Присвойте первому параметру значения Киев, а втором-Ленина 2, кв.15. После введения параметров будет получен результат выполнения параметрического запроса.

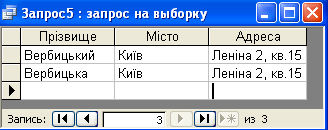
****

рисунок 14

**перекрестные запросы**

Перекрестные запросы позволяют компактно отображать отобранные данные и объединять одновременную информацию. Таблицы, в которых отражаются результаты выполнения перекрестного запроса, удобные для создания диаграмм и графиков с помощью мастера.

Определим объем выручки от продажи различных товаров при условии, что соответствующая информация сохраняется в базе данных отдельно для каждого месяца. При использовании перекрестного запроса наименование товара появляется в результирующем наборе данных только один раз, Access объединяет двенадцать записей по месяцам. Этим перекрестный запрос отличается от запроса на выборку, при выполнении которого в результирующий набор данных включается отдельно каждый из двенадцати месячных записей по данному товару.

Создадим перекрестный запрос Продавец - Покупатель на основе таблицы Торговля, используя для выбора фамилий покупателей таблицу адреса.

В меню Запрос активизируйте команду Перекрестный Запрос. Access отобразит в бланке запроса строки Total и Перекрестная таблица. Включите в бланк запроса поля Продавец и Объем операций с таблицы Торговля и поле Фамилия из таблицы адреса. Чтобы значение поля (например, Продавец) появилось в заголовках строк, нажмите на строке Перекрестная таблица и выберите в списке строку Заголовок срок. В строке Total для этого поля должна сохраниться стандартная установка Группировать по.

Чтобы значение поля Фамилия появились в заголовке колонки, нажмите на строке Перекрестная таблица и выберите строку Заголовок столбцов. В этом случае в строке Total должна остаться стандартная установка Группировать по.

Для отображения содержания поля в перекрестной таблице в качестве значения (объем операций) щелкните на строке Перекрестная таблица и выберите строку Значение. В строке Total для этого поля введите необходимую функцию- Sum.

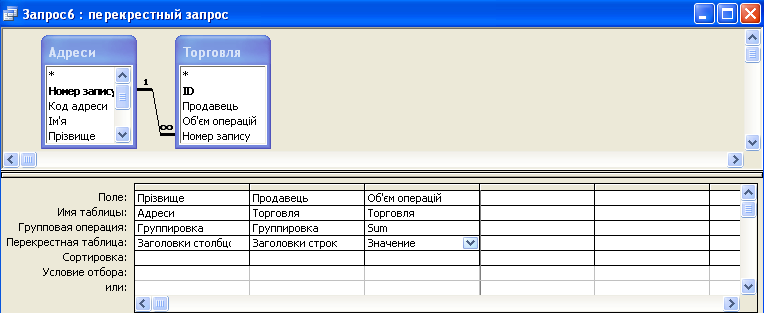
****

рисунок 15

При определении критериев отбора в строке Total для желаемого поля следует выбрать значение! Условие и ввести выражение в строку Критерии. Для этого поля строку Перекрестная таблица оставьте пустым. Критерии отбора мы вводить не будем.

Чтобы отобразить на экране результирующий набор записей созданного запроса, стоит перейти в режим таблицы.

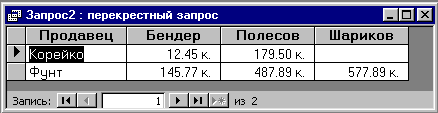
****

рисунок 16

**Создание перекрестного запроса с помощью мастера**

Рассмотрим процесс разработки перекрестного запроса с предыдущего примера с помощью специального мастера -Crosstab Query Wizard. Мастер позволяет создать перекрестный запрос на основе только одной таблицы или одного запроса.

В окне базы данных перейдите на вкладку Запрос и нажмите кнопку Новый. В диалоговом окне Новый запрос, которое при этом появится выберите значение Crosstab Query Wizard и нажмите кнопку ОК.

В первом диалоговом окне выберите таблицу или простой запрос на основе которого мастер создаст перекрестный запрос. Поскольку в нашем примере используется две таблицы, Торговля и адреса, сначала следует отразить с обеих таблиц данные, необходимые для перекрестного запроса. Эту работу мы уже проделали создав запрос Операции с клиентами с помощью мастера простого запроса. Выберите его в качестве базового и нажмите кнопку Далее.

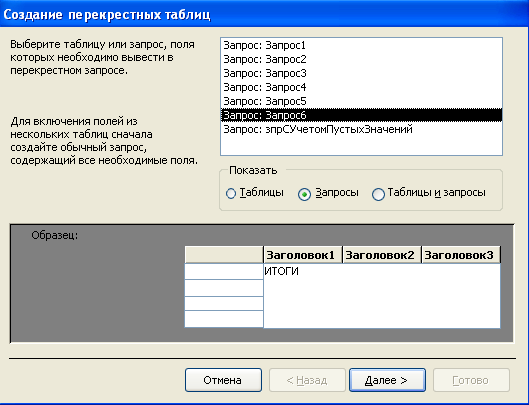
****

рисунок 17

Чтобы отметить, имена полей, которые должны использоваться в качестве заголовков строк перекрестной таблицы, из предложенных полей Доступные поля такого окна мастера выберите поле Продавец и поместите его в список Выбранные поля с помощью кнопки>.

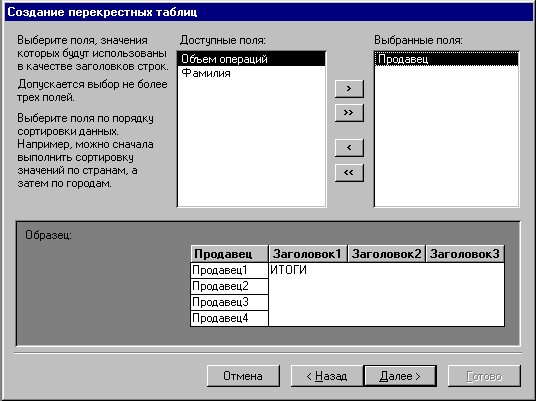
Перейдите в следующее окно мастера. В этом окне нужно выбрать поле, значение которого будут фигурировать в заголовках колонок таблицы с результатами выполнения перекрестного запроса. Выберите поле Фамилия и нажмите кнопку Далее.

рисунок 18

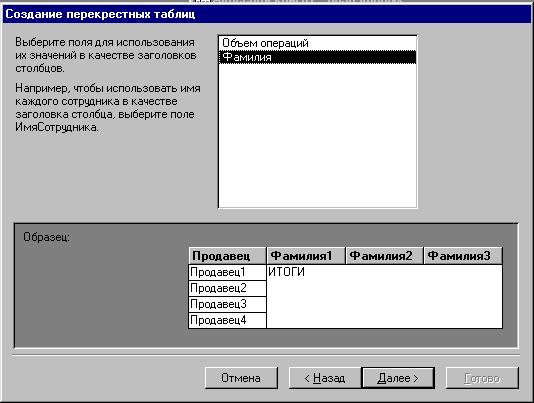
В следующем окне мастера укажите, значение которого поля должны находиться в ячейках результирующей таблицы и выберите операцию для обработки этих значень- Sum. Кроме этого в таблице с результатами выполнения перекрестного запроса нужно определить сумму значений в каждой строке. Для этого установите опцию Да, включите суммирование по срокам. В результате перекрестного запроса будет добавляться колонка со значениями объема продаж у каждого продавца

рисунок 19

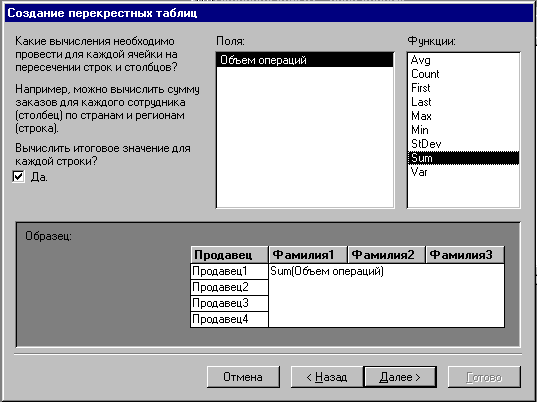


рисунок 20

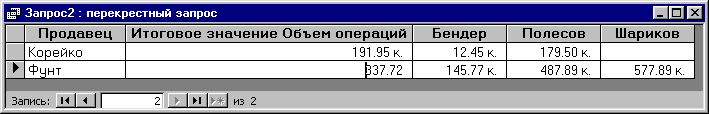


рисунок 21

Контрольные вопросы к лабораторной работе 4:

1. Какое поле называют вычислительным полем?

2. Что представляет собой выражение?

3. Перечислите правила построения выражений.

4. Перечислите операторы, которые используют в выражениях.

5. Какие функции называют встроенными функциями?

6. Каким образом можно обрабатывать пустые записи в запросах?

7. Для чего предназначен запрос на выборку?

8. Что меняется при определении свойства «Подпись» вычислительного поля?

9. Какие средства существуют в Access для построения выражений?

10. В каких случаях применяют функцию IsNull ()?

11. Как применить функцию IIF в вычислительных полях?

12. Запросы типам вы знаете?

13. Какие действия можно делать с помощью запросов на смену?

14. Чем отличаются запросы на изменение от запросов на выборку?

15. Что представляют собой параметрические запросы в чем их удобство?

16. Каким образом строят перекрестные запросы?

17. Можно ли добавлять информацию в таблице БД из других БД?

18. создает запрос на изменение динамический набор записей Recordset в итоге?

19. Как проверить действие запросов на смену?

20. Какие запросы используют для хранения резервных копий таблиц БД?

21. Какие возможности открывает запрос на возобновление?

22. Запросы какого типа служат для удаления из таблиц групп записей, соответствующих некоторому критерию отбора?