



СЕЧЕНОВСКИЙ
УНИВЕРСИТЕТ

И.А. Иконникова
Т.Е. Новосёлова
Н.Н. Пронькин
И.Ф. Семёнычева

ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ БАЗ ДАННЫХ В МЕДИЦИНЕ

Часть II

УЧЕБНОЕ ПОСОБИЕ



ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
ПЕРВЫЙ МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ МЕДИЦИНСКИЙ УНИВЕРСИТЕТ ИМЕНИ
И.М. СЕЧЕНОВА МИНИСТЕРСТВА ЗДРАВООХРАНЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
(СЕЧЕНОВСКИЙ УНИВЕРСИТЕТ)

ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ БАЗ ДАННЫХ В МЕДИЦИНЕ

ИКОННИКОВА Ирина Александровна
НОВОСЁЛОВА Тамара Евгеньевна
ПРОНЬКИН Николай Николаевич
СЕМЁНЫЧЕВА Ирина Флюровна

Учебное пособие

Москва
2021

УДК 004
ББК 16
И74

Рецензент:

Герасимов А.Н. – д.ф.-м.н., профессор, заведующий кафедры «Медицинская информатика и статистика» Первого Московского государственного медицинского университета имени И.М. Сеченова Министерства здравоохранения Российской Федерации (Сеченовский Университет)

Авторы

Иконникова Ирина Александровна, Новосёлова Тамара Евгеньевна,
Пронькин Николай Николаевич, Семёнычева Ирина Флоровна

Информационная технология баз данных в медицине. Часть II [Электронный ресурс]: учебное пособие – Эл. изд. - Электрон. текстовые дан. (1 файл pdf: 78 с.). - Иконникова И.А., Новосёлова Т.Е., Пронькин Н.Н., Семёнычева И.Ф. 2021. – Режим доступа: <http://scipro.ru/conf/database.pdf>. Сист. требования: Adobe Reader; экран 10".

ISBN 978-1-387-25950-2

Учебное пособие рассмотрено и одобрено на заседании кафедры «Медицинская информатика и статистика» 26 августа 2021 г., протокол № 1.

Учебное пособие по изучению дисциплины «Информационная технология баз данных в медицине» разработано в соответствии с требованиями государственного образовательного стандарта.

В пособии представлены методические материалы по курсу «Информационная технология баз данных в медицине». Изложены базовые сведения и освещен круг вопросов, связанных с построением и функционированием баз данных в медицине.

Учебное пособие дает обзор теоретических основ в области организации данных, а также формирует практические навыки проектирования, создания и эксплуатации баз данных.

Прикладные задачи из медицинской практики реализованы средствами СУБД Microsoft Access, которая входит в стандартный пакет Microsoft Office – наиболее популярный комплекс приложений среди аналогичных систем автоматизации офисной деятельности.

Материал излагается простым, доступным языком.

ISBN 978-1-387-25950-2



© Иконникова И.А., Новосёлова Т.Е., Пронькин Н.Н., Семёнычева И.Ф. 2021
© Первый Московский государственный медицинский университет имени И.М. Сеченова (Сеченовский Университет), 2021
© Оформление: издательство НОО Профессиональная наука, 2021

Содержание

ВВЕДЕНИЕ.....	5
ОСНОВЫ РАБОТЫ С БАЗАМИ ДАННЫХ В СУБД ACCESS	6
1. Стартовое окно приложения Access	6
2. Создание таблиц БД	8
3. Простейшие операции над данными.....	16
3.1. Поиск и замена данных.....	16
3.2. Использование фильтра	17
3.3. Запросы	21
4. Формы.....	35
5. Генерация отчётов	41
6. База данных из двух взаимосвязанных таблиц.....	47
6.1. Установление связи между таблицами базы данных ...	49
6.2. Операции над данными взаимосвязанных таблиц	52
СЛОВАРЬ ТЕРМИНОВ И ОПРЕДЕЛЕНИЙ (ГЛОССАРИЙ)	60
Библиографический список	76

ВВЕДЕНИЕ

Современные информационные технологии нашли применение в самых различных областях человеческой деятельности, в том числе и в медицине. В значительной мере это связано с характерной для постиндустриального общества тенденцией к укрупнению и “универсализацией” предприятий. Масштабные научно-лечебные комплексы демонстрируют эффективную работу на основе широкого применения передовых научных и технических достижений. В частности, это современные компьютерные системы, нацеленные на сбор, хранение, обработку и передачу значительных объёмов информации в процессе организационно-хозяйственной деятельности лечебного предприятия.

В этой связи принципиальное значение имеют два обстоятельства. Во-первых, традиционная «ручная» обработка медицинской документации обнаружила ряд серьёзных проблем. Действительно, на постоянно растущих объёмах текущих данных неотвратимо проявляется так называемый человеческий фактор – на однообразной работе люди часто ошибаются, что абсолютно недопустимо в медицине, где любая ошибка может иметь необратимые последствия. Кроме того, изготовление бумажных сводок и отчётов серьёзно (а иногда фатально) задерживает по времени информационные обмены между различными медицинскими структурами.

Отсюда становится очевидным, что широкое применение компьютеров и сопутствующих им информационных технологий обработки данных в медицине объективно необходимо. Технология баз данных напрямую соответствует задаче эффективной организации информационных обменов как в рамках отдельных крупных медицинских центров, так и в кооперации разрозненных небольших лечебных пунктов (например, ФАПов).

В связи с применением информационных технологий, основанных на использовании компьютеров и коммуникационных средств связи, широко используется понятие «информационная система». *Информационная система (ИС)* представляет собой совокупность средств и методов, предназначенных для создания и поддержки информационной модели какой-либо части реального мира (называемой *предметной областью*) с целью удовлетворения информационных потребностей пользователя. Компьютерные информационные системы позволяют хранить большие объёмы данных, осуществлять в них быстрый поиск, вносить изменения, выполнять всевозможные манипуляции с данными (выборка по заданному критерию, агрегирование, обновление и так далее).

Основой всякой информационной системы является база данных. *База данных (БД)* – организованная по определенным правилам совокупность взаимосвязанных данных, объединенных вместе по определенному признаку (относящихся, например, к заданной предметной области). Под данными здесь понимается информация, подготовленная для переработки и хранения в компьютере.

В основе любой информационной системы лежит среда хранения и доступа к данным, которую образуют базы данных (БД) и соответствующие им системы управления базами данных (СУБД).

ОСНОВЫ РАБОТЫ С БАЗАМИ ДАННЫХ В СУБД ACCESS

1. Стартовое окно приложения Access

В настоящее время рынок программных продуктов демонстрирует широкое предложение компьютерных разработок в области создания и эксплуатации баз данных. В учебных целях, однако, разумно обратить внимание на СУБД Access, которая входит в стандартный поставку Microsoft Office и пользуется в настоящий момент наибольшей популярностью среди систем автоматизации офисной деятельности.

СУБД Access (пер. с англ. – доступ) имеет ряд очевидных преимуществ, а именно, это привычный графический интерфейс, полная русификация и интеграция с другими приложениями Microsoft Office. Кроме того, СУБД Access располагает большим количеством *Мастеров* и *Построителей* для создания разнообразных приложений без непосредственного программирования.

Первичное знакомство с Access предполагает владение элементарными навыками работы с графическим интерфейсом и главными принципами функционирования системы Windows. Однако каждая новая версия Microsoft Office имеет ряд принципиальных особенностей. Так, для приложений Office 2007 компания Microsoft разработала графический интерфейс, в котором больше нет столь привычных ниспадающих меню и панелей инструментов. Строка меню в Access 2007 имеет вид ленты (Ribbon). Здесь вместо панелей инструментов, заполненных кнопками, теперь предлагаются на выбор образцы форматирования, рисунки, пиктограммы, раскрывающиеся списки и так далее.

Ниже на рис. 1 приведено рабочее окно Microsoft Access с обозначением терминологических особенностей и геометрии инструментальных средств.

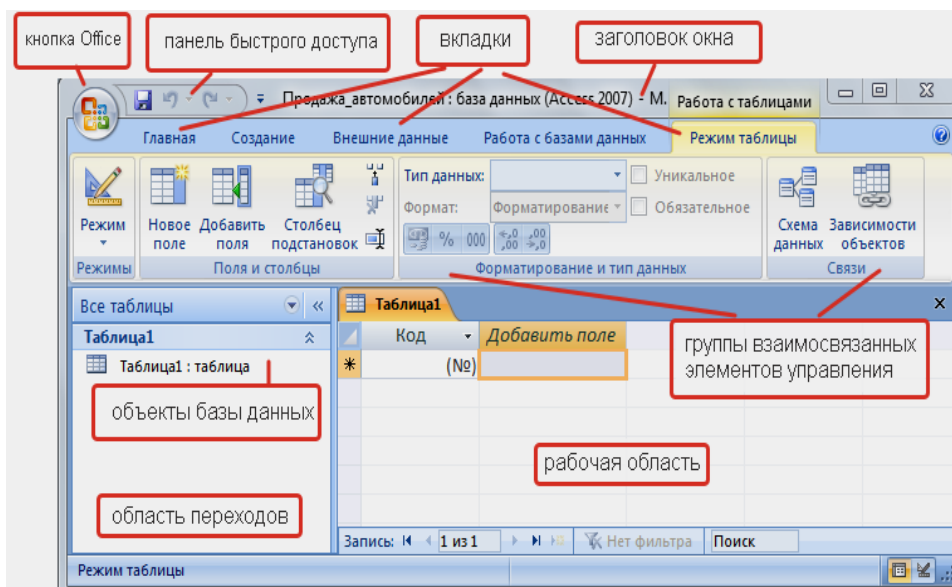


Рис. 1. Элементы интерфейса MS Access 2007

Файл базы данных (БД) в Access содержит все объекты базы данных (таблицы, запросы, связные отношения...), такая интеграция упрощает решение проблем, как с хранением объектов, так и с копированием, удалением и другими операциями над семейством таблиц и БД в целом.

Перечислим основные объекты БД.

Таблицы используют для определения и для хранения совокупности данных об отдельных объектах БД, которые объединены между собой общей темой – *предметной областью* приложения БД. Например, таблицы **Пациенты, Врачи, Отделения, Процедуры, Аптека** ... представляют в совокупности БД **СТАЦИОНАР**.

Каждая из перечисленных таблиц состоит из множества записей по одной и той же тематике, структура записей одинакова и закреплена шапкой таблицы - совокупностью заголовков столбцов (*полей*).

Запросы в основном предназначены для извлечения данных (в том числе и из разных таблиц) на основании заданных критериев отбора – это так называемые *запросы на выборку*. Критерий отбора (условие для включения данных в выборку) может иметь сложную структуру с применением арифметических (*, /, +, -, ^) и логических (and, or, not) операций, а также операций отношения (<, >, <=, >=, =, <>).

Другой тип запроса - *запрос на изменение* - предназначен для добавления, удаления и обновления данных в уже существующих таблицах. Кроме того, запросы на изменение можно использовать для создания новых таблиц.

Перекрестные запросы предназначены для группирования данных и представления их в систематизированном виде.

Формы представляют собой электронные бланки, используемые чаще всего для ввода данных в таблицу. Форма упрощает и чётко регламентирует трудоемкий процесс информационного наполнения базы, поэтому именно экранная форма дает возможность поручить ввод данных в БД персоналу с невысокой квалификацией.

Другое назначение форм – представление информации из БД в удобном и привычном для пользователя виде (бланки, сводки, ведомости и др.). Опытный пользователь может использовать форму для графического отображения данных из БД, а также для встраивания новых функциональных элементов, дополняющих стандартный интерфейс СУБД.

Отчеты служат для отображения итоговых данных из таблиц и запросов в удобной для просмотра и печати форме (например, итоговая экзаменационная ведомость, справка о выполнении перечня услуг для каждого пациента и тому подобное). Зачастую отчеты распечатывают или включают в документы других приложений. Можно разработать отчет самостоятельно с помощью *конструктора*, либо воспользоваться готовыми вариантами (*автоотчеты*), либо сконструировать отчёт под управлением *Мастера*.

Макросы и модули служат для расширения функциональных возможностей СУБД, например, для автоматизации часто выполняемых задач (открытие формы,

печать отчета и тому подобное). Макросы создаются в СУБД Access путем выбора макрокоманд из предложенного списка, тогда как модули пишутся на языке программирования Visual Basic для приложений (VBA - Visual Basic for Applications). Пусть разными инструментами, но оба компонента нацелены на программирование дополнительных процедур обработки данных, которые невозможно реализовать с помощью стандартных средств СУБД Access.

2. Создание таблиц БД

Специфику организации базы данных в Access будем обсуждать в рамках традиционной методики, то есть с привлечением конкретного примера.

Рассмотрим упрощённую в учебных целях информационную модель некоторого лечебного учреждения (назовём его **СТАЦИОНАР**), которое будет представлено двумя основными объектами: **Пациенты** и **Госпитализация**.

Сначала запускаем MS Access. Далее на странице **Приступая к работе с Microsoft Access 2007** в разделе **Новая пустая база данных** щелкаем на пиктограмме **Новая база данных**. В правой части окна указываем имя создаваемого файла (**СТАЦИОНАР**, расширение .accdb система проставит автоматически), затем определяем папку, в которой будет размещен этот файл, наконец щелкаем кнопкой мыши на значке «Создать» (рис 2).

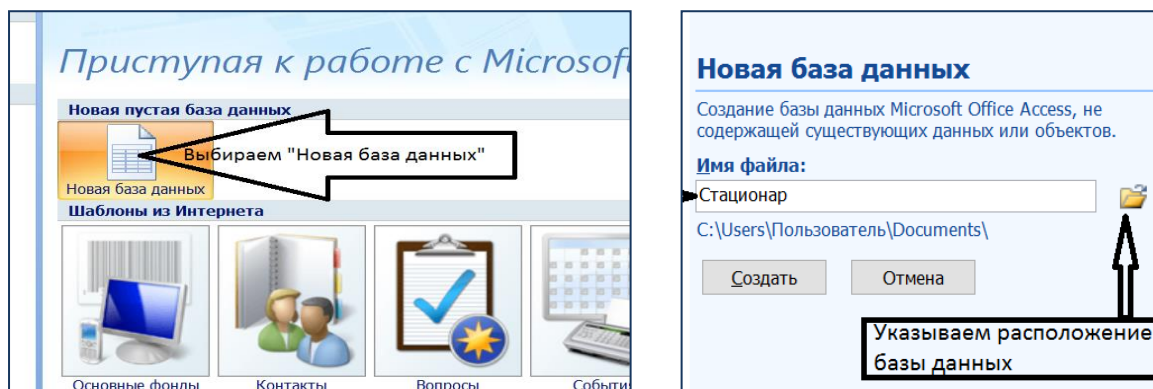


Рис. 2. Создание и именование нового файла БД

Основным элементом любой БД является таблица с записями, которые характеризуют конкретный объект, моделируемый в компьютере посредством информационной таблицы. Другими словами, часть реального мира (как объект приложения технологии БД) в процессе изучения представляет его компьютерная модель – совокупность таблица записей. Бывает, что очень простой объект (например, записную книжку) можно смоделировать в БД всего одной таблицей. Напротив, сложные *предметные области* (например, крупное предприятие) потребуют организации БД из десятков, а может и сотен таблиц.

Конструирование таблицы (*отношения* БД) предполагает, что будут определены: а) имена всем её столбцам (*полям*), б) указан тип данных в них и в) задан размер всех полей, плюс, возможно, какие-то дополнительные указания. Совокупность требований а) – в) представляет так называемую *схему отношения*.

Так, в нашем примере предусмотрим следующие имена полей в таблице **Пациенты**: № полиса, ФИО пациента, пол, дата рождения, место жительства, вид страхования.

Для каждого поля таблицы необходимо выбрать один из допустимых *типов* данных, основные из них:

текстовый – текст из букв – цифр, протяженность которого устанавливается параметром *размер поля*, максимальная длина равна 255 символам;

поле MEMO – текстовые данные пояснительного характера объемом не более 65535 символов, которые хранятся в специально организованном фрагменте памяти;

числовой – числовые данные, диапазон возможных значений определяется используемым числовым форматом (целые, вещественные ...) через параметр *размер поля*,

дата/время – даты и время, начиная с 100 г. по 9999 г. включительно;

денежный – представленные в денежных единицах числа с точностью до 15 знаков в целой и до 4 знаков в дробной части;

счетчик – натуральное число в поле данного типа представляет порядковый номер, то есть увеличивается на единицу при добавлении каждой новой записи *автоматически*, поэтому значения этих полей не могут редактироваться;

логический – для данных логического типа с двумя вариантами возможных значений: «истина» (кодируется единицей) или «ложь» (ноль);

поле объекта OLE – объект (например, электронная таблица или рисунок), созданный с помощью другого приложения;

гиперссылка – это адрес узла Web-страницы в Интернете;

мастер подстановок – программа для автоматического заполнения поля путём выбора значений из другой имеющейся таблицы или из списка предложенных значений.

Наконец, после указания типа данных для каждого поля следует указать его размер, определённый по выбранному формату данных либо значением по умолчанию.

Так, для текстовых и числовых данных соотношение типа и предельно допустимых размеров поля (то есть длины поля записи) приведены ниже в таблице 1.

Таблица 1

<i>Тип</i>	<i>Размер</i>
Байт (1 байт)	Целые числа от 0 до 255
Целое (2 байта)	Целые числа от -32768 до +32767
Длинное целое (4 байта)	Целые числа от -2 147 483 648 до +2 147 483 647
С плавающей точкой (4 байта)	С точностью до 6 десятичных знаков от $-3.4 \cdot 10^{38}$ до $+3.4 \cdot 10^{38}$
С плавающей точкой (8 байт)	С точностью до 10 десятичных знаков от $-1.797 \cdot 10^{308}$ до $+1.797 \cdot 10^{308}$
Текстовый	от 1 до 255 символов (по умолчанию – 50)

Формат поля задает визуальную форму представления данных при выводе на экран или печать. Для числовых, денежных и дата/время данных в Access предусмотрены типовые наборы стандартных форматов представления. Например, для данных типа дата/время существует следующий набор форматов:

- *полный формат даты* (15.06.2021 17:32:41);
- *длинный формат даты* (15 июня 2021 г.);
- *средний формат даты* (15-июн-2021);
- *краткий формат даты* (15.06.2021);
- *длинный формат времени* (17:32:41);
- *средний формат времени* (5:32);
- *краткий формат времени* (17:32).

В некоторых случаях длину поля в таблице определяет *маска ввода* – это шаблон для ввода данных с фиксированной структурой, то есть имеющих одинаковый формат. Маска ввода автоматически отображает неизменяемые символы поля (табл. 2), в результате пользователю остаётся заполнить только пустые позиции.

Таблица 2

<i>Обязательные символы</i>	<i>Необязательные символы</i>
0 – цифра	9 – цифра
L – буква	# – цифра, знак +, знак – , пробел
A – буква или цифра	a – буква или цифра
@ – любой символ или пробел	C – любой символ или пробел

Маска ввода, например, для обычного номера телефона может иметь следующий вид: (8-000)000-00-00.

С учётом приведённых выше инструментов конструирования таблиц схема отношения **Пациенты** может быть спроектирована следующим образом:

Таблица 3

ПАЦИЕНТЫ

<i>Имя поля</i>	<i>Тип данных</i>	<i>Размер поля</i>
№ полиса	числовой	целое
Фамилия	текстовый	20
Пол	текстовый	3
Дата рождения	Дата/время	Краткий формат даты
Место жительства	текстовый	20
Вид страхования	текстовый	20

СУБД Access допускает различные пути создания таблицы:

- с помощью конструктора таблиц;
- с помощью *Мастера таблиц* (типовые таблицы, для которых разработаны стандартные шаблоны);
- путем ввода данных (обычно применяется для таблиц с простой структурой, например, при создании справочников).

Остановимся на первом варианте, то есть структуру первой таблицы **Пациенты** оформим в режиме **Конструктора**. Для этого нужно щелкнуть мышью на пиктограмме **Режим** в левом верхнем углу окна и выбрать **режим Конструктор** (рис. 3).

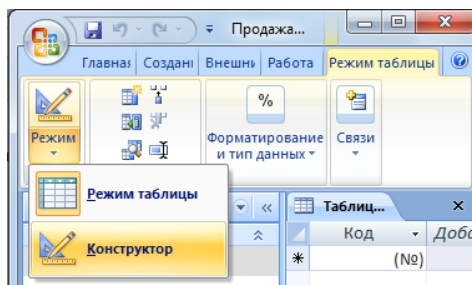


Рис. 3. Выбор режима **Конструктор**

Окно **Конструктора** (рис. 4) разделено на две части. В верхней части нужно указать имена полей и тип данных в них. Область «Описание» предназначена для поясняющего текста – необязательный для заполнения фрагмент.

В нижней части **Конструктора** в разделе «Свойства поля» можно выбрать дополнительные настройки числового и других форматов полей.

Так, согласно проекту (табл. 3), первое поле *№ полиса* предполагает хранение в нём данных числового типа, это будут целые числа (раздел «Свойства поля»), то есть размер поля (табл. 1) ограничен шестизначными целыми числами.

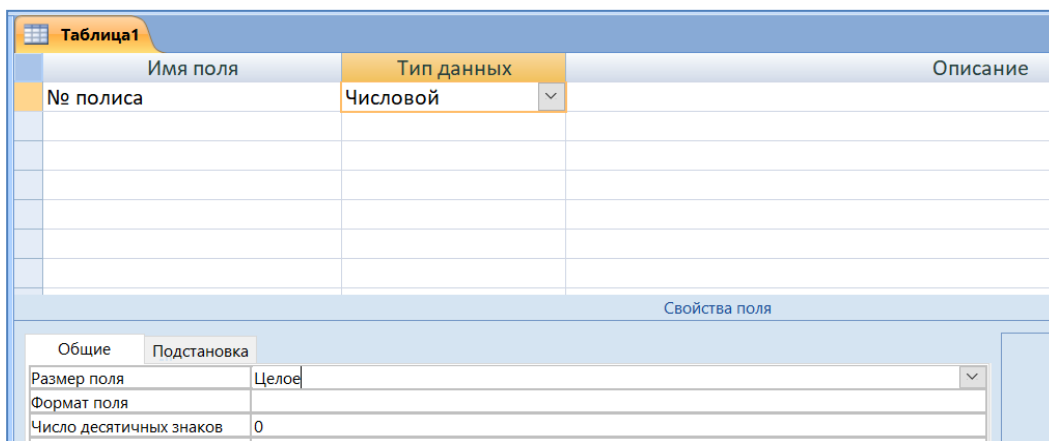


Рис. 4. Окно Конструктора для описания схемы отношения

Подобным образом оформляем структуру второго поля с именем *Фамилия*, тип данных – текстовый, длина поля пусть будет равна 20 позициям.

Обратим внимание на третье поле *Пол*, в нём, как это следует из названия, будет повторяться одно из двух возможных значений: М или Ж. Чтобы вручную не набирать каждый раз то М, то Ж, организуют ниспадающий список, который содержит все варианты возможных значений для данного поля. Пользователю остаётся только выбрать из этого списка подходящей для данной записи вариант.

Соответствующая процедура называется организацией *поля подстановки из списка возможных значений*. Для этого в нижней части окна конструктора – «Свойства поля» - выбирают вкладку «Подстановка». Далее (рис. 5) определяют следующие параметры процедуры:

- Тип элемента управления – Поле со списком;
- Тип источника строк – Список значений;
- Источник строк – перечисление (через точку с запятой) вариантов возможных значений для данного поля, в нашем примере - М и Ж. Кавычки, как признак текстовой константы, будут проставлены автоматически.

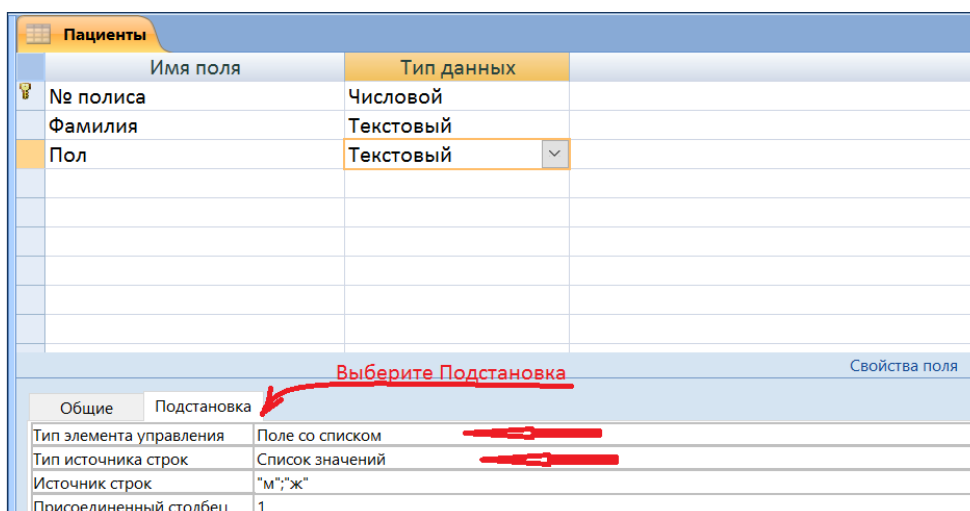


Рис. 5. Организация поля подстановки из списка возможных значений

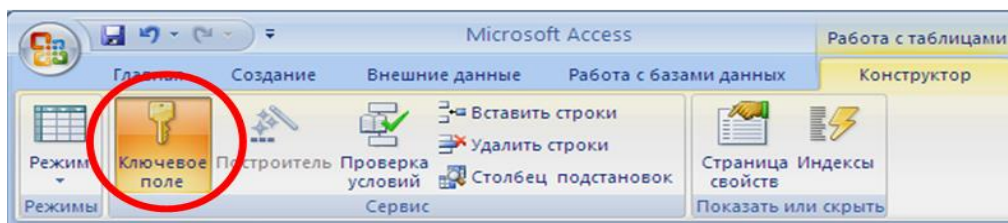
Дальнейшее описание структуры таблицы **Пациенты** не вызывает никаких трудностей – достаточно сверять свои действия с проектом (табл. 3). Заметим только, что поле *Вид страхования* разумно оформить подстановкой по аналогии с полем *Пол*. В качестве возможных значений укажем следующие варианты: ОМС (обязательное медицинское страхование), ДМС (добровольное медицинское страхование) и ММС (международное медицинское страхование).

Пациенты	
Имя поля	Тип данных
№ полиса	Числовой
Фамилия	Текстовый
Пол	Текстовый
Дата рождения	Дата/время
Место жительства	Текстовый
Вид страхования	Текстовый

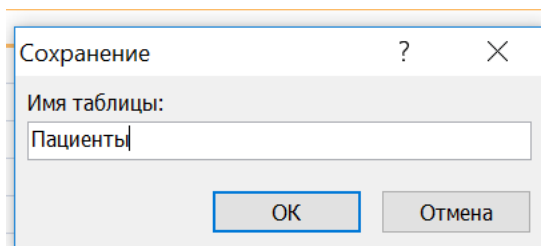
Общие	Подстановка
Тип элемента управления	Список
Тип источника строк	Список значений
Источник строк	"ОМС";"ДМС";"ММС"
Присоединенный столбец	1
Число столбцов	1

Конструирование таблицы завершает обязательное действие - указание ключевого поля.

В нашем примере, очевидно, на роль ключевого претендует поле *№ полиса*, поэтому установим на нём указатель записей и щелкнем по кнопке с изображением ключа на панели инструментов (либо щелкнем на этом поле правой кнопкой мыши и в контекстном меню назначаем данное поле ключевым).



Далее следует закрыть конструктор и в диалоговом окне «**Сохранение**» присвоить таблице (пусть пустой, но уже с «нарисованной» структурой) имя **Пациенты**.



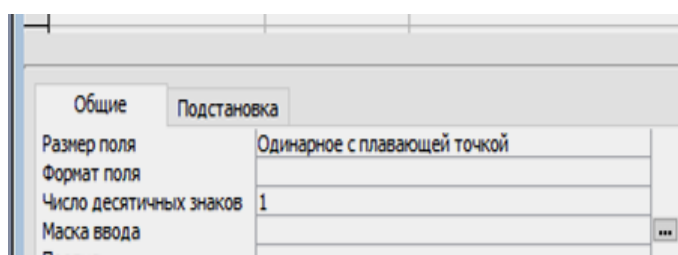
Важно отметить, что готовый проект таблицы можно модифицировать: добавлять или удалять поля, изменять их параметры. При этом все исправления разумно внести до начала заполнения таблицы данными, поскольку попытка изменить параметры полей уже заполненной таблицы может повлечь за собой потерю или искажение данных.

Для редактирования структуры таблицы нужно открыть её в режиме **Конструктора**. Для этого среди объектов базы данных в области переходов выделим имя таблицы и из контекстного меню (или с помощью кнопки «Вид») выберем **Конструктор**.

Для удаления поля из проекта таблицы следует выделить ненужную строку, щелкнув мышью в области маркировки поля, и в контекстном меню выбрать команду **Удалить строки** (если таблица была заполнена данными, то при удалении поля уничтожаются все находящиеся в нем данные).

Добавим для примера два новых поля «Вес» (в килограммах) и «Рост» (в метрах). Для этого откроем таблицу **Пациенты** в режиме Конструктора и в конце списка полей добавим два новых: поле *Вес* с числовым типом (целое), а для поля *Рост* – одинарное с плавающей точкой и одним десятичным знаком:

Пациенты : таблица	
Имя поля	Тип данных
№ полиса	Числовой
Фамилия	Текстовый
Пол	Текстовый
Дата рождения	Дата/время
Место жительства	Текстовый
Вид страхования	Текстовый
Вес	Числовой
Рост	Числовой



Сохраним отредактированный вариант таблицы с прежним именем.

Теперь следует наполнить таблицу **Пациенты** данными. Для этого в окне БД делаем двойной щелчок мышью по имени таблицы. В результате получим пустую таблицу, которая соответствует по структуре заказанному проекту. Вносим данные, передвигаясь в каждой записи по полям с помощью клавиш управления курсором либо клавишей <Tab>. Перемещение между записями выполняется посредством кнопок панели навигации (рис. 6), которая находится в левом нижнем углу окна таблицы либо командой **Перейти** (вкладка **Главная**).

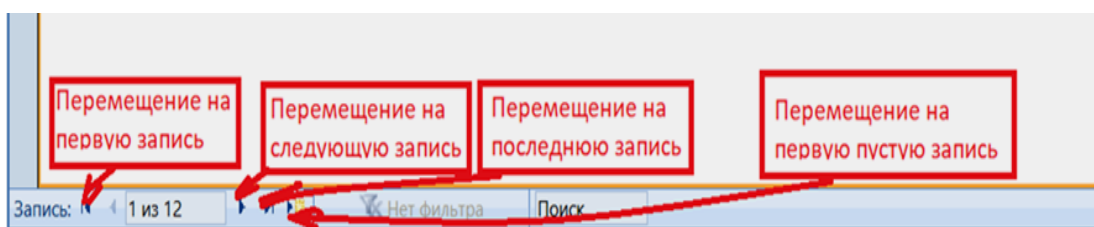


Рис. 6. Кнопки навигации по записям таблицы

Заполним таблицу данными (рис. 7), соблюдая некоторые условия, которые помогут нам в дальнейшем изучении СУБД Access:

1. Числа в ключевом поле (*№ полиса*) не должны повторяться. Пусть это будут натуральные числа, начиная с единицы.
2. Фамилии хотя бы двух пациентов начинаются с одной и той же буквы.
3. Города, указанные в качестве места жительства, должны иногда повторяться.

Ниже приведён пример наполнения таблицы конкретными данными.

№ полиса	Фамилия	Пол	Дата рождения	Место жительства	Вид страхования	Вес	Рост
1	Иванов	м	21.11.1952	Москва	ОМС	80	1,8
2	Петрова	ж	12.01.1991	Сочи	ОМС	70	1,6
3	Сидоров	м	04.10.1988	Москва	ОМС	95	1,8
4	Березина	ж	08.03.1955	Видное	ОМС	65	1,7
5	Дубов	м	15.05.1945	Берлин	ММС	85	1,8
6	Кошкин	м	05.06.1935	Москва	ДМС	75	1,6
7	Серов	м	08.11.2000	Калуга	ДМС	90	1,9
8	Волкова	ж	04.10.1953	Сочи	ОМС	60	1,6
9	Собакин	м	22.04.1998	Москва	ММС	100	1,8
10	Лисицын	м	16.11.1955	Коломна	ОМС	110	1,7
11	Соловьёва	ж	19.06.1957	Казань	ДМС	60	1,5
12	Антонова	ж	30.11.1953	Москва	ОМС	80	1,5

Рис. 7. Наполненная данными таблица-отношение **Пациенты**

Удаление устаревших либо ошибочных данных выполняется в два приема: сначала выделяют фрагмент ненужных записей в столбце маркировки (крайний левый

в таблице), затем применяют команду **Удаление** (вкладка **Главная**), либо из контекстного меню, либо с клавиатуры (клавишей «Del»).

Новые записи добавляют в произвольной последовательности, стартуя с первой свободной строки таблицы.


Операции копирования и перемещения маркированной записи выполняются обычным путем через буфер обмена.

Таблицу, заполненную данными и отредактированную, сохраняем. Будем использовать её в примерах при изучении основных операций по извлечению информации из базы данных.

3. Простейшие операции над данными

Цель создания БД – обеспечение информационных потребностей пользователя. Действительно, информационная поддержка производственного процесса предполагает систематический поиск данных из БД по заданному условию их отбора. Ниже будут рассмотрены типичные инструменты поиска данных, которые обеспечивает любая СУБД реляционного типа, в том числе и Access.

3.1. Поиск и замена данных

Операция поиска записей по указанному в поле значению выполняется кнопкой  (вкладка **Главная**). В диалоговом окне определяется имя поля и нужное значение в нем. Дополнительно можно указать область поиска: выше (ниже) указателя записей либо во всей таблице. Например, нужно найти данные на пациентов из Москвы (рис. 8). Обозначим условия поиска в диалоговом окне и активируем его выполнение.

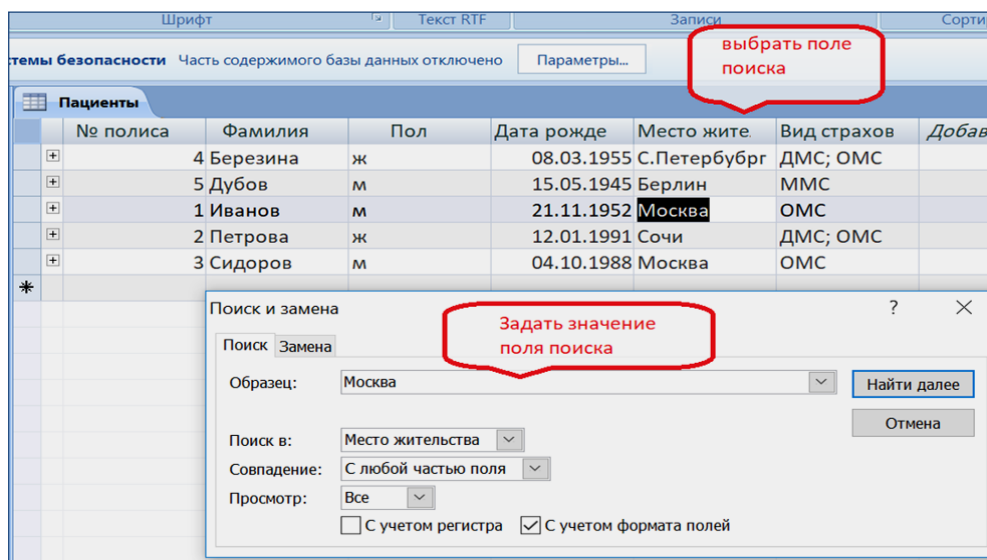


Рис.8. Диалоговое окно для задания параметров поиска записей из БД

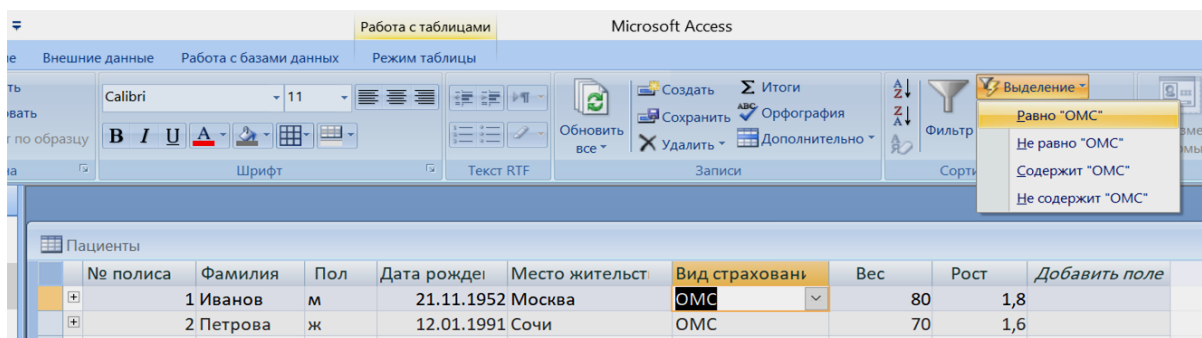
Результат: указатель записей установится на строке с данными на пациента с фамилией Иванов, дальнейший поиск (указание «найти далее») маркирует запись для пациента с фамилией Сидоров.

Вторая закладка этого же окна (*Замена*) выполняет поиск с последующей заменой “старого” значения на “новое”. С помощью этой операции можно, например, указать новое место жительства пациента.

3.2. Использование фильтра

Установка фильтра позволяет выбрать для последующей обработки часть записей из БД, которые удовлетворяют заданному условию - *критерию отбора*. Фильтры можно прилагать к таблице, запросу или форме, но фильтруются всегда данные только одной таблицы. Результат фильтрации – часть данных, отвечающих критерию отбора. Фильтры по сути - это одноразовые запросы без имени.

Простейший вариант – фильтр по выделенному. Критерий отбора в этом случае - одно конкретное значение данных, которое обозначают выделением. Например, пусть для последующей обработки нужны записи для пациентов с полисом обязательного медицинского страхования. Для этого в поле Вид страхования выделяем нужное значение – ОМС, затем на вкладке *Главная* (группа *Сортировка и фильтр*) выбираем команду *Выделение* с указанием способа фильтрации из ниспадающего списка.



Получаем таблицу записей, отвечающих условию фильтрации – список пациентов с полисом ОМС:

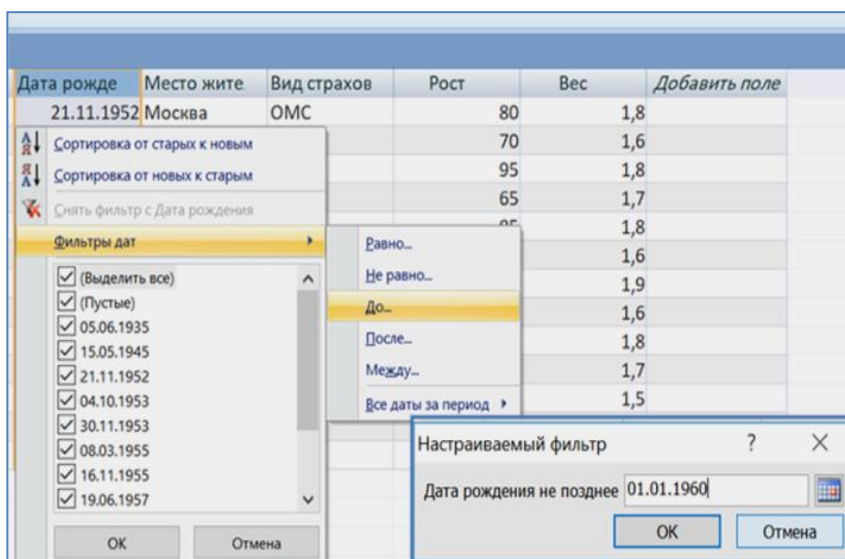
	№ полиса	Фамилия	Пол	Дата рождения	Место жительства	Вид страхования
▶	1	Иванов	м	21.11.1952	Москва	ОМС
	2	Петрова	ж	12.01.1991	Сочи	ОМС
	3	Сидоров	м	04.10.1988	Москва	ОМС
	4	Березина	ж	08.03.1955	Видное	ОМС
	8	Волкова	ж	04.10.1953	Сочи	ОМС
	10	Лисицын	м	16.11.1955	Коломна	ОМС
	12	Антонова	ж	30.11.1953	Москва	ОМС
*	0					

С полученной таким образом информационной таблицей (записи только для пациентов с ОМС) можно работать дальше сколь угодно долго. Чтобы обозначить факт работы «под фильтром», СУБД отображает кнопку «с фильтром» на навигационной панели внизу таблицы или формы:



Вернуть таблицу в исходное состояние, то есть снять фильтр, можно, отжав кнопку «Применить фильтр» (группа **Сортировка и фильтр** на вкладке **Главная**).

Некоторые часто применяемые фильтры доступны в виде команд контекстного меню – нужно только щелкнуть правой кнопкой мыши поле, к которому нужно применить фильтр. Система сама предложит набор типовых (для данного поля) условий отбора. Это так называемый настраиваемый фильтр, пример которого приведён ниже: ниспадающий список ограничений на значения поля *Дата рождения* позволяет отобразить список пациентов старше заданного возраста.



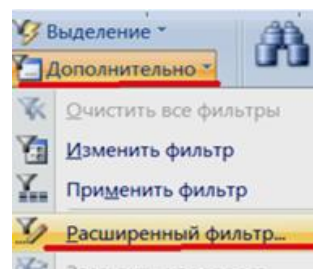
По типу поля фильтрации система сама организует набор подходящих операций для формулировки критерия, задача пользователя – уточнить его при необходимости. Далее исполняем команду на фильтрацию (кнопка «Применить фильтр», группа **Сортировка и фильтр** на вкладке **Главная**).

№ полиса	Фамилия	Пол	Дата рождения	Место жительства	Вид страхования	Вес	Рост
1	Иванов	м	21.11.1952	Москва	ОМС	80	1,8
4	Березина	ж	08.03.1955	Видное	ОМС	65	1,7
5	Дубов	м	15.05.1945	Берлин	ММС	85	1,8
6	Кошкин	м	05.06.1935	Москва	ДМС	75	1,6
8	Волкова	ж	04.10.1953	Сочи	ОМС	60	1,6
10	Лисицын	м	16.11.1955	Коломна	ОМС	110	1,7
11	Соловьёва	ж	19.06.1957	Казань	ДМС	60	1,5
12	Антонова	ж	30.11.1953	Москва	ОМС	80	1,5

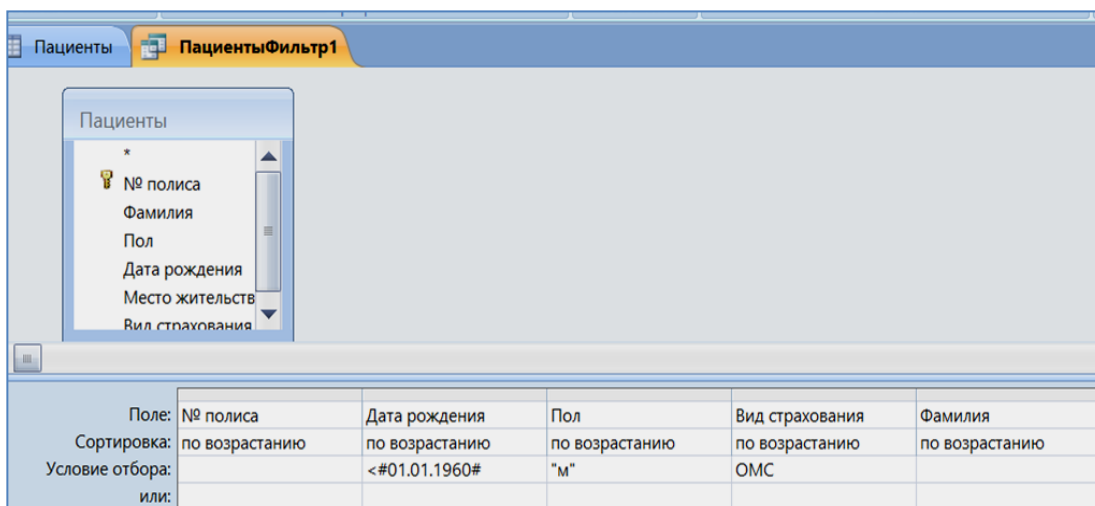
В результате получили список из 8 пациентов с датой рождения до 1 января 1960 года, что и требовалось сделать в поставленной задаче.

Более сложные условия фильтрации реализуют с помощью расширенного фильтра, который позволяет задать условие отбора сразу *по нескольким полям* таблицы.

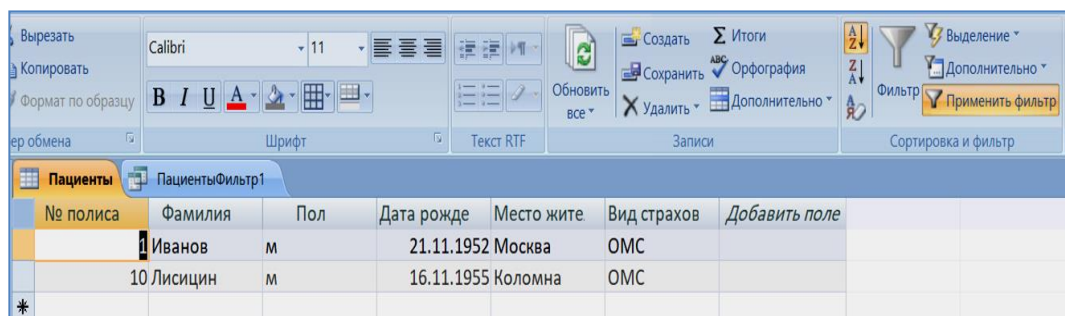
Расширенный фильтр формируют посредством заполнения бланка, в нём перечислены название всех полей таблицы, для каждого из них можно задать условие отбора в специально отведённой для этого строке «Условие отбора».



Пусть нужно, например, организовать список пациентов-мужчин с полисом ОМС и датой рождения до начала 1960 года:



После команды на установку фильтра (*Применить фильтр* группа *Сортировка и фильтр* на вкладке *Главная*) получим ожидаемый результат:



Более сложные условия фильтрации получаются с использованием логических операций (and, or, not).

По окончании работы с фильтрами следует вернуть таблицу с записями в исходный вид, то есть отменить фильтры посредством команды **Дополнительно - Очистить все фильтры**.

В завершение этого раздела обсудим две наиболее часто применяемые функции фильтра – **Сортировка по возрастанию** и **Сортировка по убыванию**, они выведены на панель инструментов как отдельные кнопки. Сортировка упорядочивает данные по содержимому выделенных полей. Чаще всего упорядочению подлежат данные числового типа (по возрастанию или по убыванию), даты (хронологический или обратный - хронологический порядок), текстовые данные (по алфавиту либо против алфавита). Польза подобной операции очевидна, например, при желании выбрать наиболее дешевый (или наиболее свежий) товар.

Для сортировки записей по одному из полей таблицы нужно выделить соответствующий столбец, щелкнув по его заголовку, затем нажать подходящую кнопку «Сортировка по возрастанию» или «Сортировка по убыванию» (группа **Сортировка и фильтр** на вкладке **Главная**).

Если нужно сортировать данные нескольких столбцов, то их также сначала выделяют, затем применяют команду на сортировку. В результате данные первого из выделенных столбцов упорядочиваются стандартно, данные во втором столбце будут отсортированы в рамках каждой из групп, сложившихся в первом столбце, и так далее. Пример сложной сортировки приведен ниже. Были выделены два поля: «Вид страхования» и «Фамилия». Упорядочение по первому из них выделило три группы (ДМС, ММС, ОМС), в пределах каждой из этих групп фамилии пациентов были выстроены в алфавитном порядке:

	Вид страхования	Фамилия
	ДМС	Кошкин
	ДМС	Серов
	ДМС	Соловьёва
	ММС	Дубов
	ММС	Собакин
	ОМС	Антонова
	ОМС	Березина
	ОМС	Волкова
	ОМС	Иванов
	ОМС	Лисицын
	ОМС	Петрова
	ОМС	Сидоров

Важно отметить, что при большом числе записей в файле данных установка фильтра (равно как сортировка записей) серьезно сокращает время доступа к нужной информации, следовательно, обеспечивает эффективную работу СУБД. Следует помнить, что СУБД Access выдает запрос на сохранение результатов сортировки при закрытии таблицы. Конструкции фильтров, напротив, не сохраняются, поэтому каждый раз их нужно создавать заново (либо сохранить нужный фильтр в виде

запроса, пользуясь соответствующей командой: **Фильтр – Дополнительно – Сохранить как запрос**).

3.3. Запросы

Запрос является объектом БД и представляет собой сохраненный под определенным именем набор команд для многократного применения к записям из базы данных.

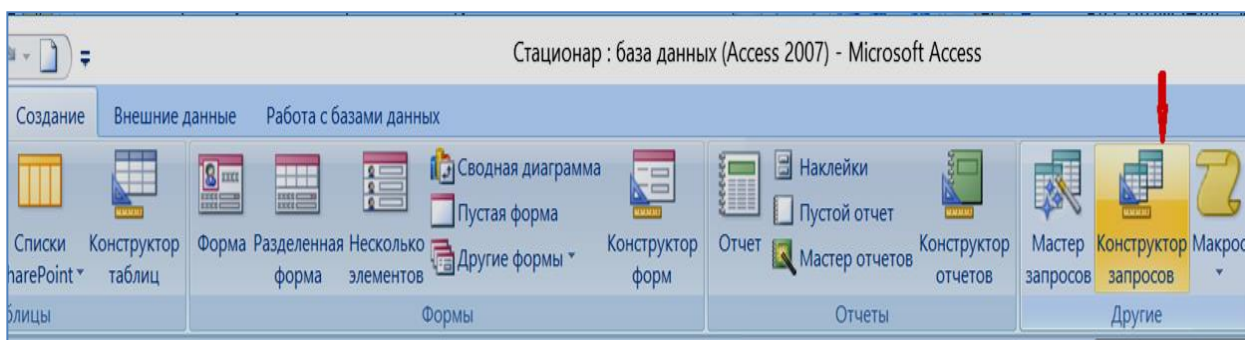
Основные функции запроса:

- выбирать нужные записи и/или поля из разных таблиц и запросов (запросы на выборку);
- выполнять расчеты на основе имеющихся в таблице данных (запросы на вычисление);
- добавлять новые, изменять или удалять устаревшие данные (запросы на изменение).

Запросы можно сохранять и использовать повторно, например, в качестве информационной основы для нового запроса.

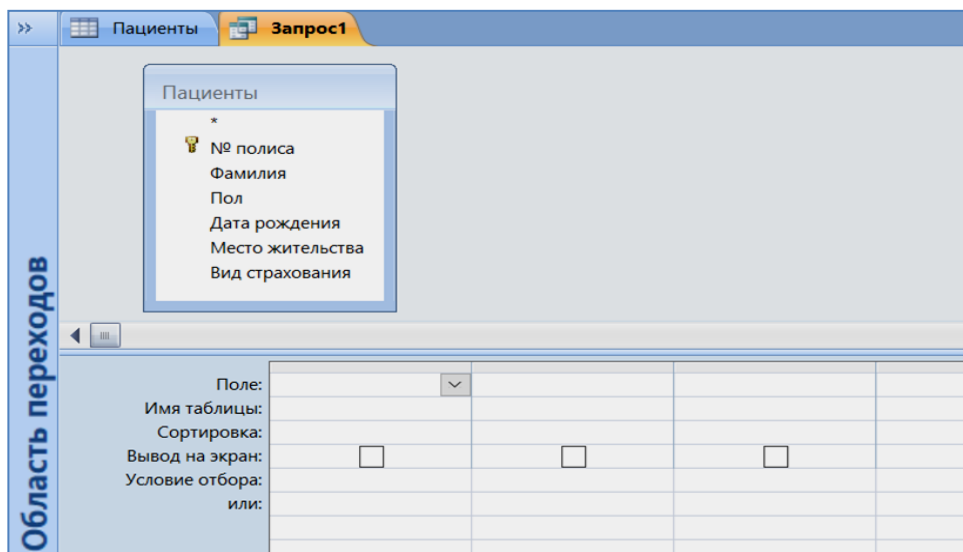
Запросы на выборку создают динамические, временные таблицы с записями данных из БД, которые удовлетворяют заданным условиям отбора (*критерию отбора*). Здесь «динамические» – значит меняющие свое содержание при изменении данных в БД. «Временные» – значит, что такие таблицы существуют только в течение сеанса, если их не записать в память соответствующей командой. Рассмотрим простейший способ формирования запросов – по образцу (так называемые QBE-запросы). Пользователь должен указать параметры запроса в окне конструирования, формируя тем самым образцы для поиска информации. QBE относится к непроцедурным языкам, так как он описывает только желаемый результат без указания пути его достижения.

Для построения запроса вызываем Конструктор запросов (вкладка **Создание**):



В результате появится пустое окно конструктора запроса и диалоговое окно «Добавление таблицы» поверх него. В последнем нужно указать имена всех таблиц,

которые служат источниками данных для запроса. В нашем примере БД «СТАЦИОНАР» имеется пока только одна таблица **Пациенты**, поэтому помечаем ее и даём команду **Добавить – Заккрыть**.

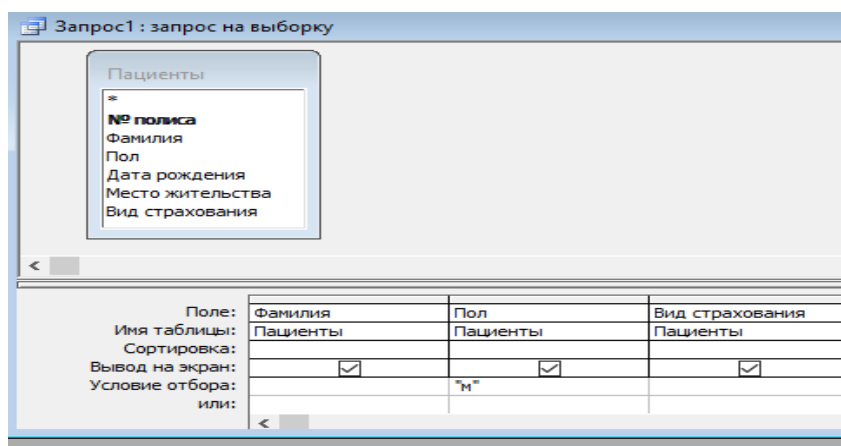


В верхней части окна находится схема только что добавленной в запрос таблицы-источника. Нижняя часть окна отведена под разработку запроса, то есть для указания:

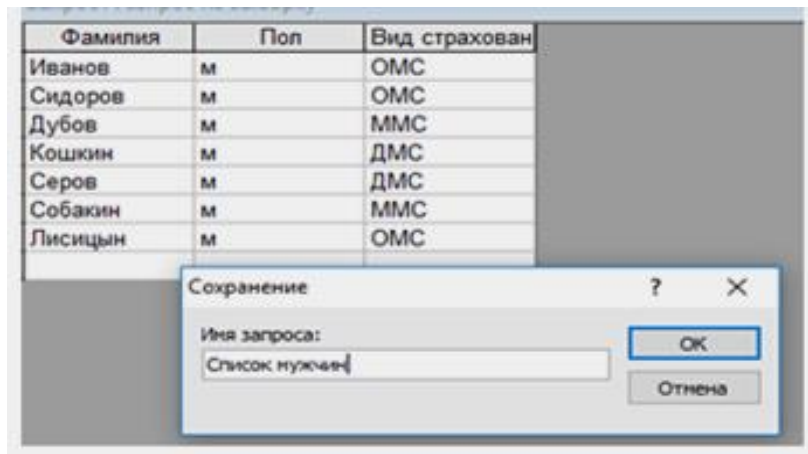
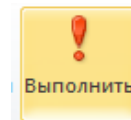
- полей – участников запроса;
- критериев отбора в этих полях;
- типа сортировки в любом из выбранных полей запроса.

Сформируем, например, список пациентов-мужчин с указанием их фамилии и вида страхования. Для этого:

- выберем в последовательных столбцах опции «Поле» *Фамилия*, *Пол* и *Вид страхования* из ниспадающего списка имён всех полей таблицы;
- в условии отбора для поля *Пол* вводим критерий «М», кавычки как признак текстового значения СУБД поставит автоматически;
- поемим флагами вывод на экран всех заказанных полей таблицы.



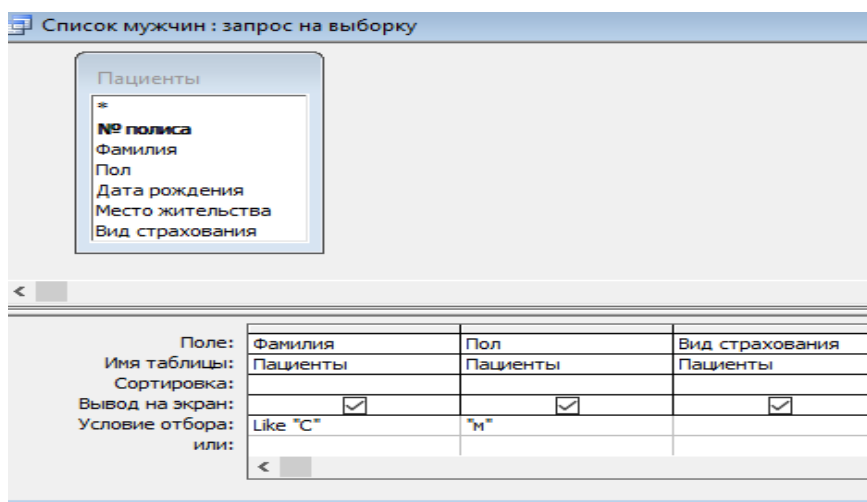
Запрос сформирован, команда на исполнение **Выполнить** дублируется кнопкой с восклицательным знаком на панели инструментов (группа **Результаты**).



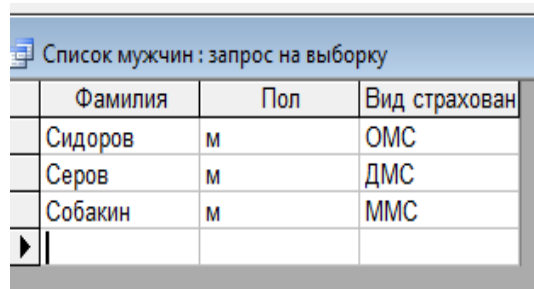
Результат сохраним с именем «Список мужчин». Отметим, что тут же в окне базы данных появился новый объект – запрос «Список мужчин».

Сохранённый в базе запрос можно редактировать, открыв его с этой целью в режиме Конструктора. «Старая» конструкция запроса просто дополняется новыми критериями. Например, в списке пациентов-мужчин выберем только тех, фамилия которых начинается с буквы «С».

Для этого имеющийся в БД запрос «Список мужчин» откроем в режиме Конструктора. Далее в строке «Условие отбора» для поля *Фамилия* нужно указать комбинацию С* (что означает: первая буква С, а после неё может стоять любой набор символов). После нажатия клавиши «Enter» в обновлённом условии отбора автоматически появится оператор подобия Like («схожие между собой по первой букве С»).



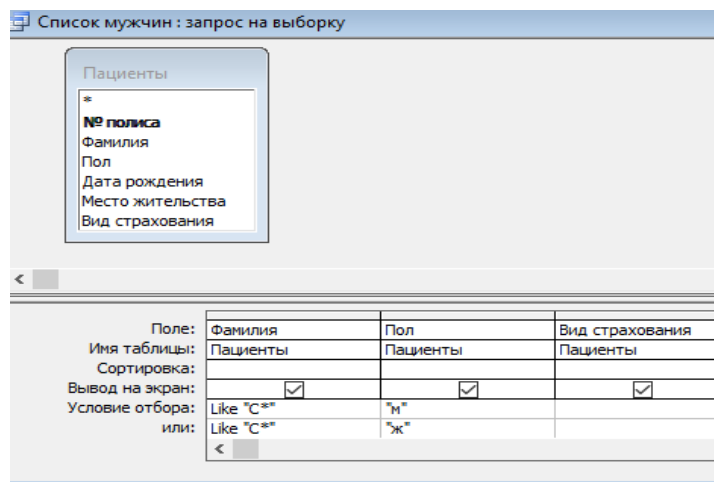
Результат – более короткий список из трёх пациентов-мужчин, а именно только тех, фамилии которых начинаются на букву «С».



Фамилия	Пол	Вид страхован
Сидоров	м	ОМС
Серов	м	ДМС
Собакин	м	ММС

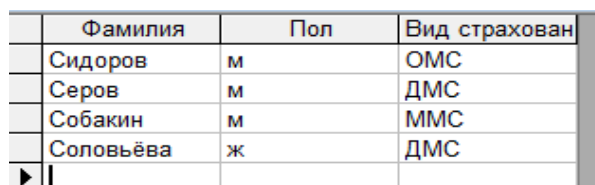
Приведённые примеры демонстрируют примечательный факт: если ограничения на значения полей размещены в одной и той же строке «Условие отбора», то предполагается их одновременное выполнение, то есть условия отбора соединяются союзом «И».

Бывает, однако, что логика запроса требует соединения условий отбора через союз «ИЛИ». Для формулировки таких условий в окне запросов предусмотрена специальная строка, которая так и озаглавлена - «ИЛИ». Пример. Найдём помимо мужчин также женщин-пациентов, фамилии которых начинаются с буквы «С».



Поле:	Фамилия	Пол	Вид страхования
Имя таблицы:	Пациенты	Пациенты	Пациенты
Сортировка:			
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Условие отбора:	или: Like "С*"	или: Like "С*"	

Приведённое здесь условие отбора трактуется так: либо мужчина **И** фамилия на букву “С” **ИЛИ** женщина **И** фамилия на букву “С”. Напомним, что приоритет логической операции **И** выше, чем логической операции **ИЛИ**, поэтому сформированный критерий отбора соответствует поставленной задаче. Результат запроса:



Фамилия	Пол	Вид страхован
Сидоров	м	ОМС
Серов	м	ДМС
Собакин	м	ММС
Соловьёва	ж	ДМС

Таким образом, условия отбора данных из БД могут быть довольно сложными, в частности, они могут включать логические операции (конъюнкция и дизъюнкцию в приведённом выше примере), выражения и некоторые функции Access (табл. 4). Ниже приведён ряд полезных примеров, которые демонстрируют типовые варианты, применяемые для записи условий отбора.

Таблица 4

Условие отбора	Трактовка условия отбора
ОМС	имеют значение «ОМС»
Not OMC	имеют значение, отличное от «ОМС»
< M	начинаются с букв А-Л
>= M	начинаются с букв М-Я
P*	начинаются с буквы Р
100	имеют числовое значение 100
<=900	имеют числовое значение 900 и меньше
Date()	имеют значение текущей даты
>=#01.01.94#	имеют значение даты позднее 01.01.94
Between #01.01.93# AND #23.01.93#	имеют значение между 01.01.93 и 23.01.93
.02.	имеют значение месяца Февраль
Null	содержат в поле пустое значение
Is Not Null	имеют значение, отличное от пустого

Параметрические запросы на выборку

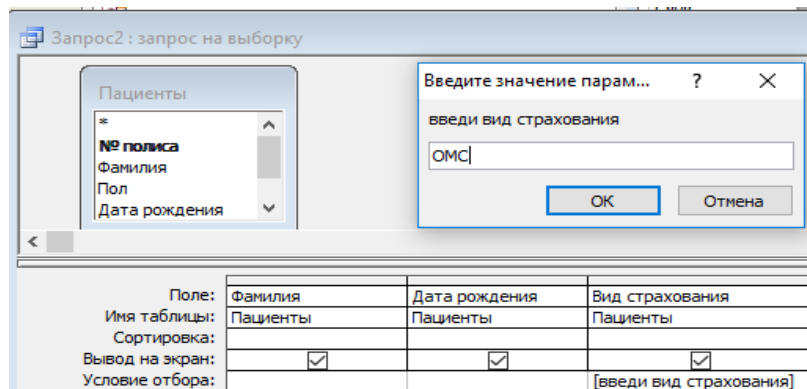
Для того чтобы сделать запрос более универсальным, можно включить в запрос переменную величину – параметр. Значение параметра заранее неизвестно, а определяется пользователем непосредственно перед выполнением запроса. Такой запрос называется параметрическим. Например, заведующего аптекой *регулярно* интересуется список препаратов с просроченной датой хранения. С этой целью создаётся (и сохраняется в базе) запрос, в котором перед исполнением нужно задать каждый раз новую дату проверки.

При создании параметрического запроса в строку «Условие отбора» вводится фраза **в квадратных скобках**, которая напоминает пользователю, что сейчас нужно указать значение параметра отбора. В нашем примере – дату проверки.

Таким образом, параметрический запрос – такой запрос, в котором одно или несколько значений в критерии отбора вводятся пользователем непосредственно при выполнении запроса.

Пусть в таблице «Пациенты» нас регулярно интересуют обладатели данного вида страхования. Вид страхования – параметр. Тогда в бланке запроса для поля *Вид*

страхования в строке Условие отбора в квадратных скобках запишем текст-напоминание: [введи вид страхования].



Выполнение такого запроса предваряет диалоговое окно "Введи значение параметра", в котором программа ожидает указания конкретного значения, которое будет подставлено на место выражения в квадратных скобках. И только после этого запрос будет выполнен.

The screenshot shows the result of the query in a table with the following data:

	Фамилия	Дата рождения	Вид страхования
	Иванов	21.11.1952	ОМС
	Петрова	12.01.1991	ОМС
	Сидоров	04.10.1988	ОМС
	Березина	08.03.1955	ОМС
	Волкова	04.10.1953	ОМС
	Лисицын	16.11.1955	ОМС
	Антонова	30.11.1953	ОМС

Сохранить в БД такой запрос лучше под именем «Запрос с параметром», напоминая тем самым, что каждый его повторный запуск позволяет изменять значение критерия отбора записей.

Запросы с групповыми операциями

Запросы этого вида работают как бы в два приёма. Сначала формируются группы – множества записей с одинаковыми значениями в указанном поле группировки. Затем в другом заданном поле для этих групп вычисляются итоговые значения с помощью одной из встроенных статистических функций:

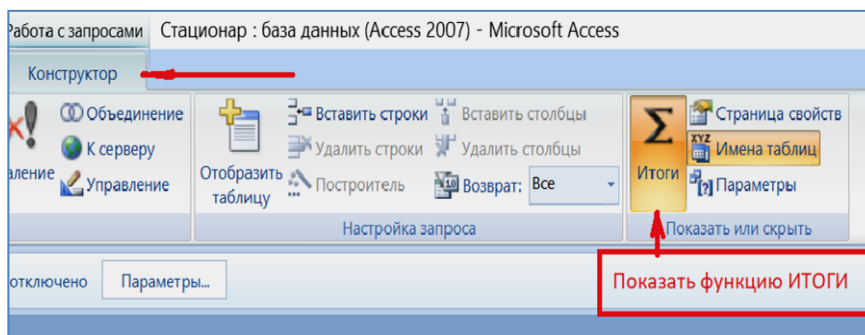
- Sum — сумма значений некоторого поля для каждой группы;
- Avg — среднее значений некоторого поля для каждой группы;
- Max, Min — максимальное, минимальное значение в группах для некоторого поля;
- Count — число значений в каждой группе для указанного поля (без учета пустых значений);

- StDev — среднееквадратичное отклонение значений в указанном поле для каждой группы;
- Var — дисперсия значений в указанном поле для каждой выделенной группы;
- First или Last — значение поля для первой или последней записи в каждой группе.

Статистические функции применяются для следующих типов полей: Числовое, Дата/время, Денежный.

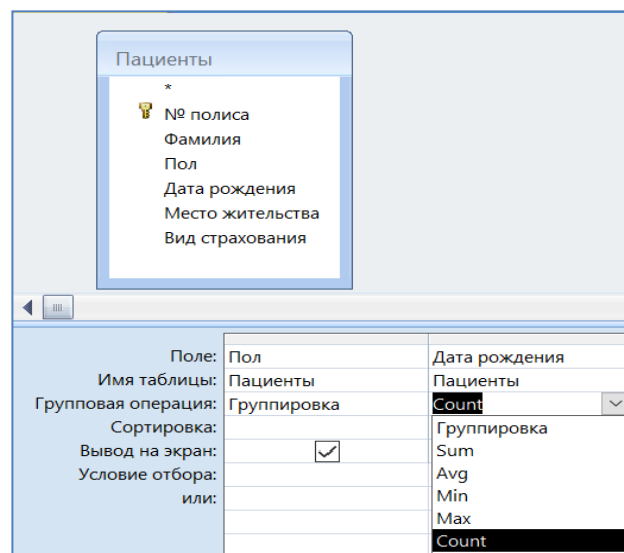
Процедура формирования запроса:

- вызываем Конструктор запросов;
- добавляем в запрос анализируемую таблицу;
- нажимаем кнопку со значком Σ на панели инструментов, указывая тем самым, что данный запрос будет использовать групповые операции.

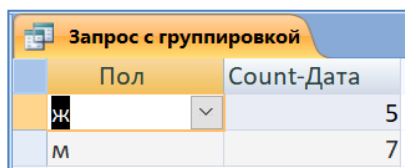


В результате в бланке запроса появится дополнительная строка «Групповая операция», в которой следует указать тип желаемого итога.

Например, определим количество женщин и количество мужчин среди пациентов стационара. Отберём в запрос два поля: *Пол* (поле группировки) и, например, *Дата рождения* (итоговое поле). Количество объектов в группе вычислим с применением статистической функции Count:



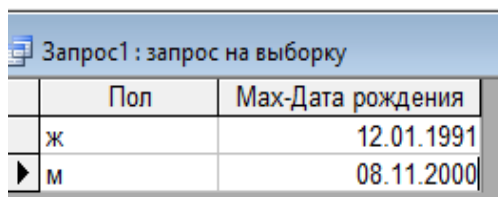
После выполнения запроса для каждого значения поля *Пол* будет вычислено количество значений в поле *Дата рождения*, то есть число женщин и мужчин:



Пол	Count-Дата
Ж	5
М	7

Результат: среди пациентов обнаружено 5 женщин и 7 мужчин.

В этом же запросе изменим статистическую функцию: вместо Count укажем Max. Очевидно, максимальная дата рождения соответствует признаку «самый молодой». Результат:



Пол	Max-Дата рождения
Ж	12.01.1991
М	08.11.2000

Таким образом, по дате рождения были зафиксированы самая молодая женщина-пациент и самый молодой пациент-мужчина.

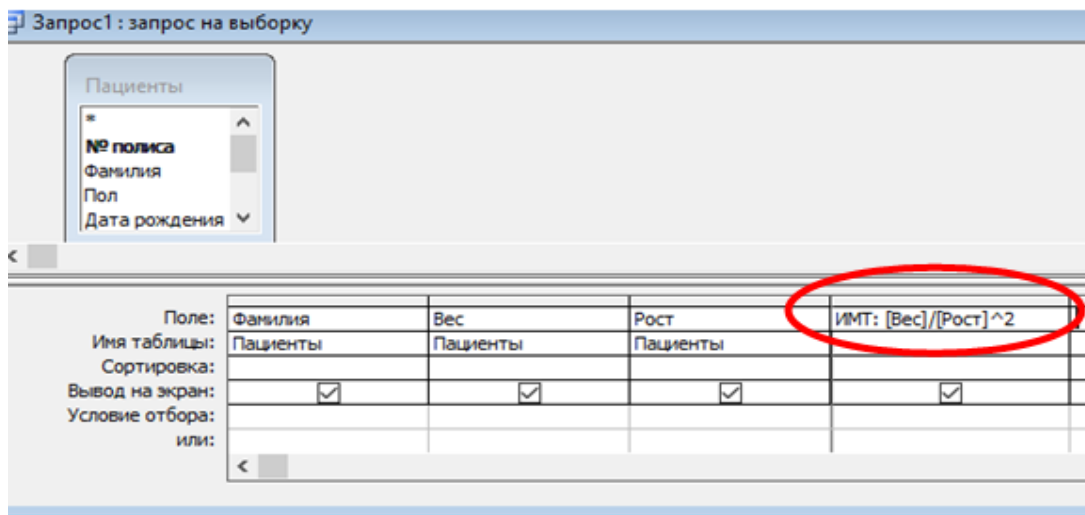
Запросы на выборку с вычисляемыми полями

Пусть поставлена следующая задача: по имеющимся в БД данным вычислить новый показатель. Например, по известным для пациентов величинам роста и веса нужно определить индекс массы тела (ИМТ).

Формально это значит, что в запросе следует построить дополнительное поле (назовём его ИМТ), значения которого будут вычисляться по формуле:

$$\text{ИМТ} = \text{Вес} / \text{Рост}^2.$$

При этом системе нужно указать, какие поля уже есть в таблице (именно из них выбирают исходные данные для расчётов), а какие поля будут новыми (содержать результаты расчётов). С этой целью принято правило, согласно которому имена «старых» полей заключают в квадратные скобки, а имя «нового» поля в формуле отделяют двоеточием:



Исполняем запрос, сохраняем его в БД с именем «ИМТ».

Фамилия	Вес	Рост	ИМТ
Иванов	80	1,8	24,691359333
Петрова	70	1,6	27,343749185
Сидоров	95	1,8	29,320989208
Березина	65	1,7	22,491348219
Дубов	85	1,8	26,234569291
Кошкин	75	1,6	29,296874127
Серов	90	1,9	24,930748548
Волкова	60	1,6	23,437499302
Собакин	100	1,8	30,864199166
Лисицын	110	1,7	38,062281602
Соловьёва	60	1,5	26,666666667
Антонова	80	1,5	35,555555556

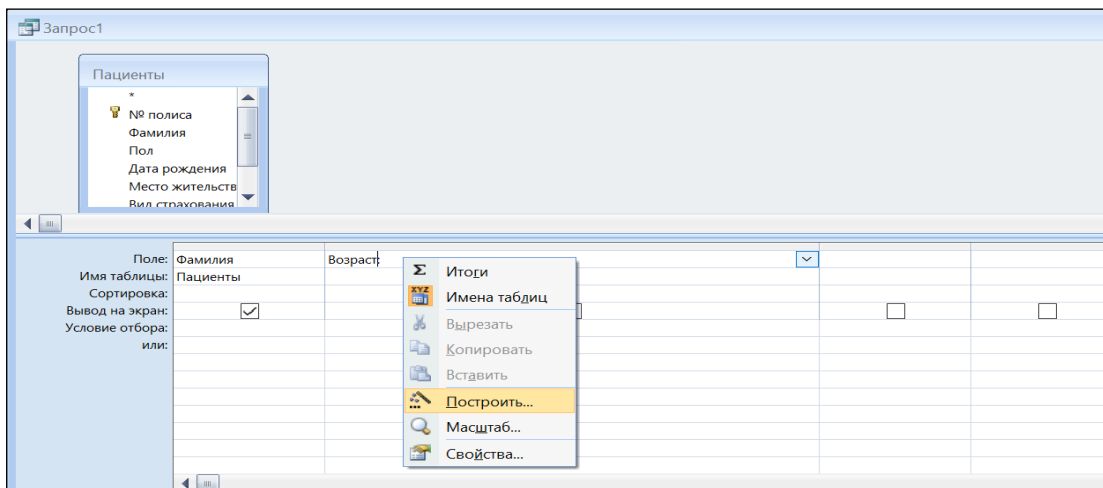
Следует помнить, что новый показатель (ИМТ) получен посредством *запроса на выборку*, поэтому его результат никак не отображается в данных исходной таблицы-источника **Пациенты**. Другими словами, полученные в результате запроса на выборку таблицы данных существуют в дальнейшем независимо от таблиц-источников.

На будущее заметим, что изменение данных в исходных таблицах-источниках тоже возможно, но достигается это другими инструментами СУБД. Ниже этот вопрос будет рассмотрен подробнее.

Построитель выражений

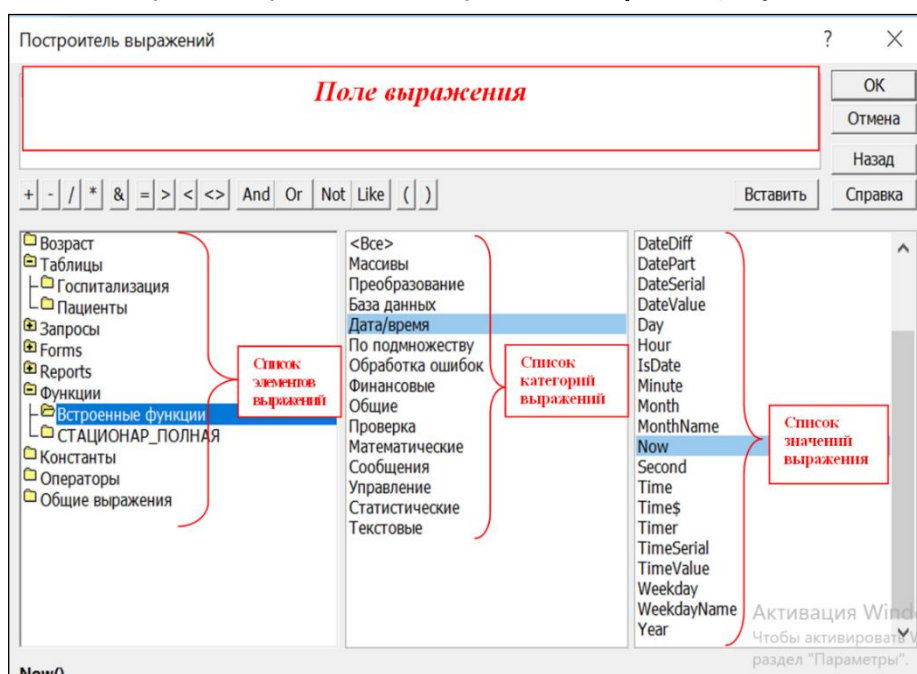
Для записи сложных расчётных формул удобно применять **Построитель выражений**, который вызывают с помощью контекстного меню в вычисляемом поле. Для примера, рассчитаем возраст каждого пациента на текущую дату. С этой целью создадим новый запрос по таблице **Пациенты**, введём в него поле **Фамилия**, а следующее за ним свободное поле озаглавим как **Возраст**.

С помощью контекстного меню вызовем **Построитель выражений**:



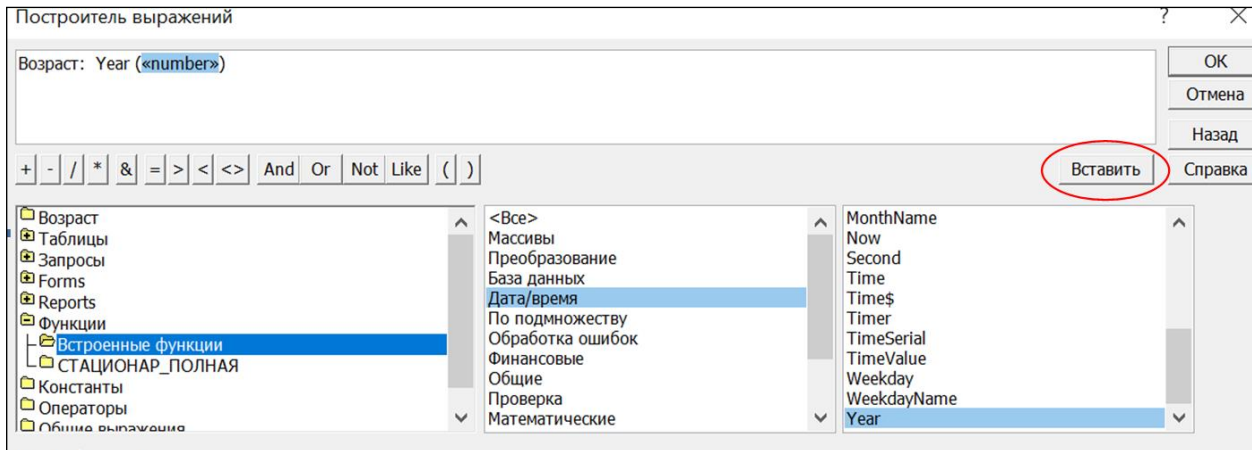
В нижней части Построителя выражений находятся три списка, из которых можно выбрать элементы конструируемого выражения.

- **Список элементов выражений** содержит объекты выражения верхнего уровня, в том числе объекты базы данных, функции, константы, операторы и т.п.
- **Список категорий выражений** содержит элементы или категории элементов, которые соответствуют выбору в **списке элементов выражений**. Так, если в **списке элементов выражений** выбраны **Встроенные функции**, то в **списке Категории выражений** будут категории функций (*дата/время, финансовые...*).
- **Список значения выражений** содержит значения (если они существуют) для двух выбранных ранее элементов и категорий. Например, если в левом списке выбран пункт **Встроенные функции**, а в среднем — категории функций - *дата/время*, то в **Списке значений выражений** будут указаны все встроенные функции из выбранной ранее категории *дата/время* (Day, Now, Month...).

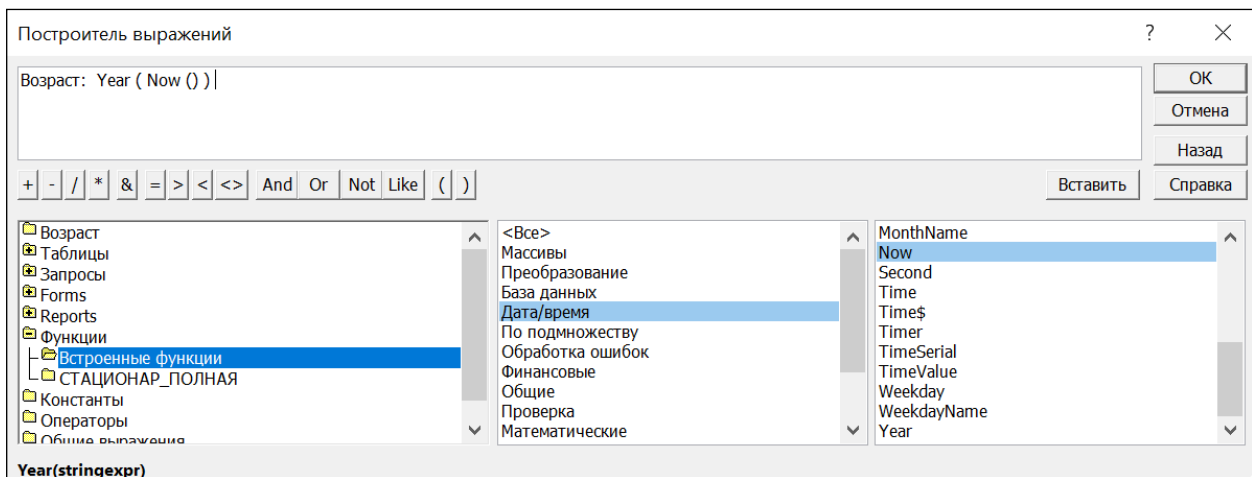


Для простоты будем полагать, что текущий возраст пациента – это разница (в годах) между текущим годом и годом из даты рождения пациента (что не совсем точно).

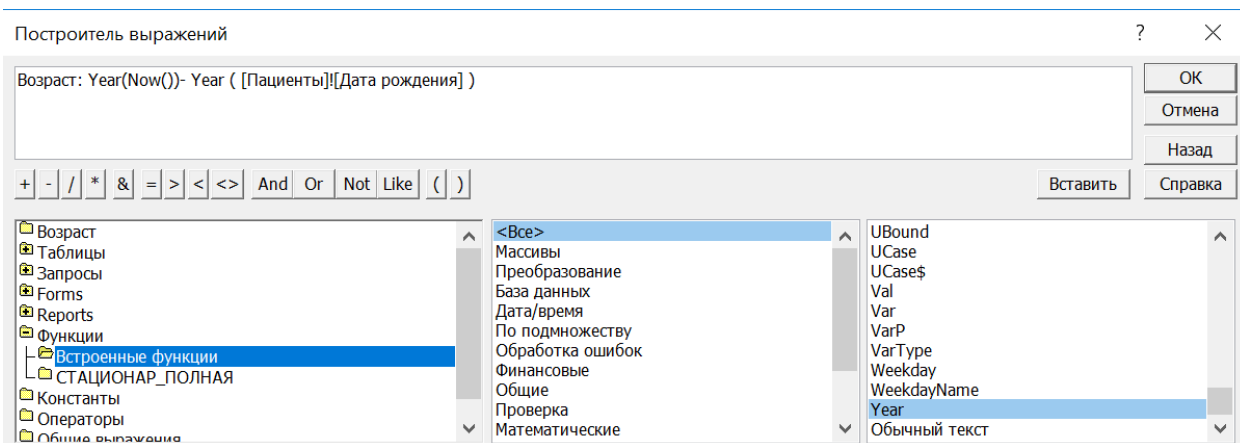
Из **Списка элементов выражений** выбираем **Встроенные функции**, далее из **Списка категорий выражений** выбираем функцию **Дата/время** и из **Списка значений выражений** выбираем функцию **Year**, нажимаем кнопку **Вставить**:



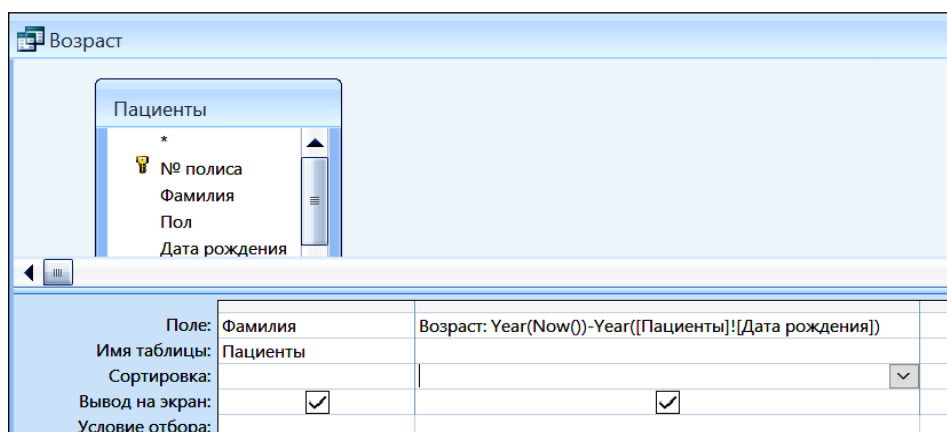
В качестве аргумента функции «number» вставляем текущую дату **Now**, ее также выбираем из **Списка значений выражений**:



Далее ставим знак минус, выбираем функцию **Year** и здесь в качестве аргумента этой функции выбираем **Дату рождения** из таблицы **Пациенты**:



Выражение построено, нажимаем ОК.



Выполняем запрос. Его результат:

Фамилия	Возраст
Иванов	66
Петрова	27
Сидоров	30
Березина	63
Дубов	73
Кошкин	83
Серов	18
Волкова	65
Собакин	20
Лисицын	63
Соловьёва	61
Антонова	65

Запись: 1 из 12 Нет фильтра

Запросы на выполнение действий

Наряду с привычными запросами на выборку существуют также *запросы на выполнение действий*, которые применяют как для создания новых, так и для модификации “старых”, то есть исходных таблиц данных.

Запросы на выполнение действий позволяют изменять таблицы базы данных: обновлять содержимое таблиц (*запрос на обновление*), дополнять новыми записями (*запрос на добавление*), удалять устаревшие записи (*запрос на удаление*), а также создавать дополнительные таблицы данных (*запрос на создание таблицы*).

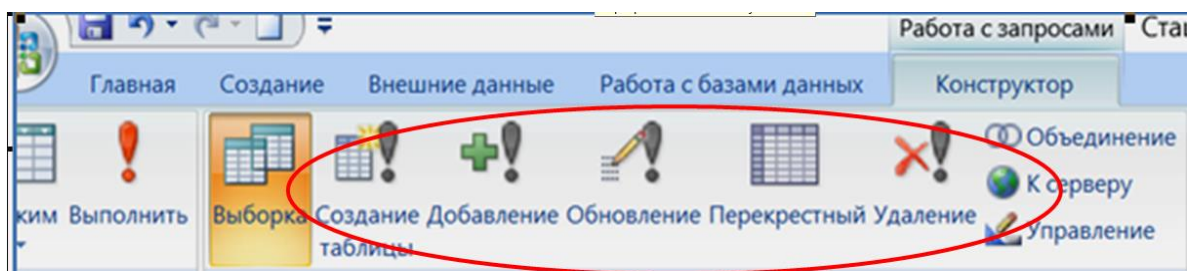


Рис. 9. Типы запросов на выполнение действий вкладки **Конструктор**

Каждое из этих действий выполняют в два приема: сначала создают запрос на выделение записей, которые нужно изменить, а затем указывают тип воздействия на отобранные данные.

Запрос на обновление данных **необратим**, это значит, что после его исполнения восстановить старые данные при ошибочной операции обновления уже невозможно. Поэтому использовать эту процедуру нужно с большой осторожностью. Тем не менее, в некоторых случаях запрос на обновление оказывается весьма полезным.

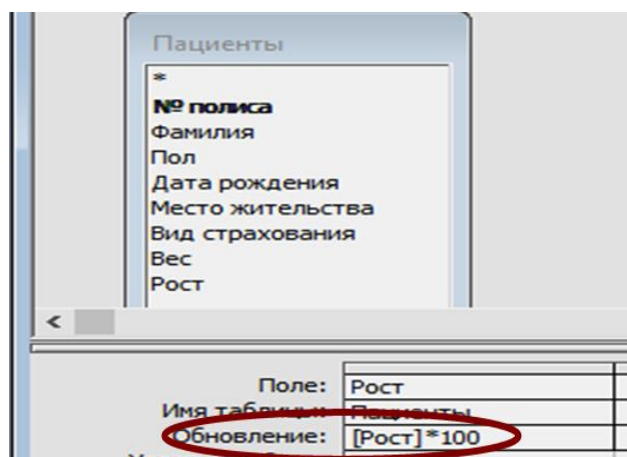
Дело в том, что запрос этого типа модифицирует определённые выделением данные при каждом повторном исполнении. Где это может быть полезно? Ответ: сохранив запрос на обновление с некоторым именем, вы можете использовать его, например, для *систематического* обновления величины заработной платы при ежемесячном увеличении её на процент инфляции. С помощью запроса на обновление можно также проводить уценку на заданный процент определённого набора товаров (причём не обязательно одних и тех же товаров).

Таким образом, приведённые выше примеры демонстрируют специфику запросов на обновление – они модифицируют данные в исходных, то есть объектных таблицах БД.

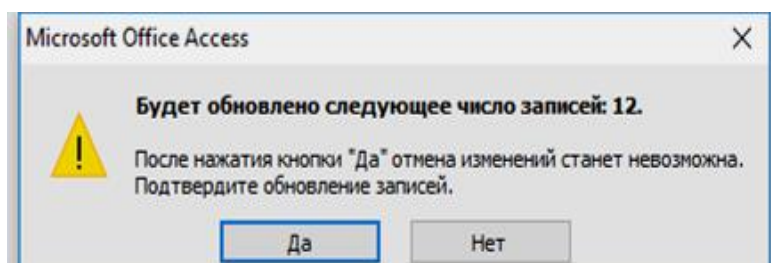
Пусть, например, в таблице **Пациенты** потребовалось указать рост в сантиметрах, тогда как в исходном состоянии рост был указан в метрах. Понятно, что имеющиеся в поле *Рост* значения нужно просто умножить на 100.

В конструкторе запросов добавляем таблицу **Пациенты**. Затем даем команду **Обновление** (на вкладке Конструктора запросов кнопка «Обновление», рис. 9).

В окне запроса появляется дополнительная строка для записи формулы, которая определяет новые значения в поле Рост:



После команды на исполнение запроса СУБД выдаёт предупреждение о количестве записей, которые будут изменены необратимо:



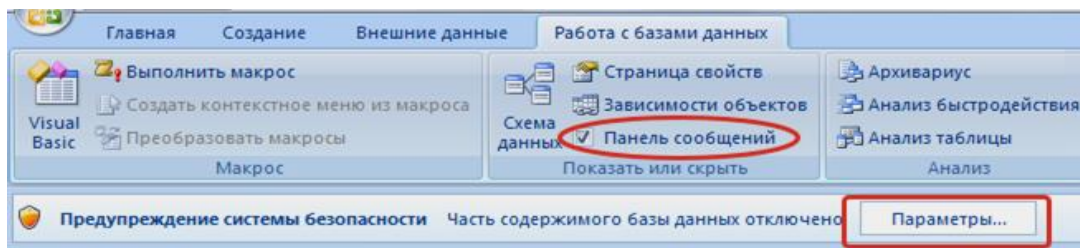
Таким образом, команда на внесение изменений в таблицу будет выполнена только после дополнительного одобрения планируемых действий. В результате можно убедиться, открыв для просмотра таблицу **Пациенты**.

Замечания.

- 1) Иногда по прошествии некоторого времени (чаще в результате ошибки) может потребоваться обратная замена обновлённых данных. Для этого, очевидно, следует выполнить ещё один запрос на обновление с подходящей формулой обратного пересчёта ($[Рост]/100$ в нашем примере).
- 2) В некоторых версиях Access предусмотрена блокировка всех запросов на изменение данных. Сообщение «Данное действие или событие заблокировано в режиме отключения» появляется в строке состояния при каждой попытке выполнить запрос на обновление.

Чтобы обойти этот запрет, нужно выполнить следующие действия:

- на панели сообщений нажмите кнопку «Параметры» (если нет панели сообщений, откройте вкладку **Работа с базами данных** и щелкните команду **Панель сообщений**);



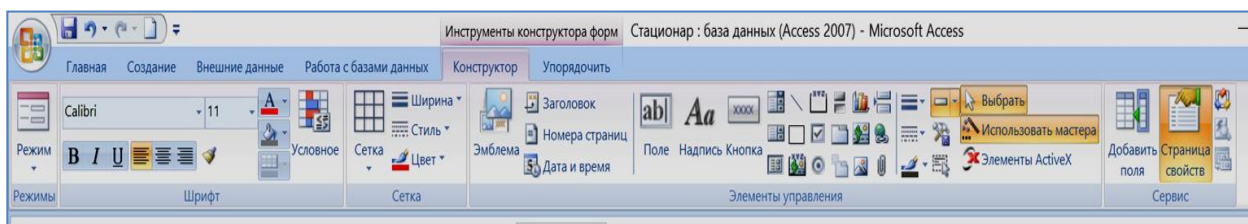
– в окне «Параметры безопасности Microsoft Office» установите переключатель в положение «Включить это содержимое». После этого снова выполните запрос на обновление.

4. Формы

Форма является объектом базы данных, с помощью которого удаётся автоматизировать ряд рутинных операций при работе с информацией из БД. А именно, формы в виде электронных бланков чаще всего используют для ввода данных в таблицу. Форма-бланк упрощает трудоемкий процесс наполнения базы, что дает возможность поручить ввод информации персоналу с невысокой квалификацией. Другое назначение форм – представление информации в удобном и привычном для пользователя электронном или печатном виде. Это, например, бланки, справки, ведомости и так далее.

Таким образом, формы являются одним из важных средств организации пользовательского интерфейса в приложениях Access.

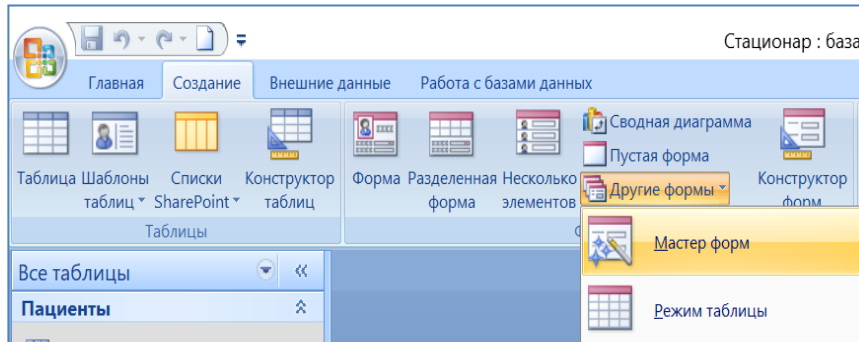
Любая форма строится на основе таблицы или запроса, которые служат источником нужных полей. Их расположение в форме произвольно (в отличие от таблицы), поэтому устанавливается из соображений наглядности или по желанию заказчика. Введение других (необязательных) элементов в форму связано с ее использованием в качестве отдельного печатного документа. Дополнительные элементы, такие как рисунки, графики, надписи и тому подобное сосредоточены на специальной инструментальной панели, которая сопровождает Конструктор форм.



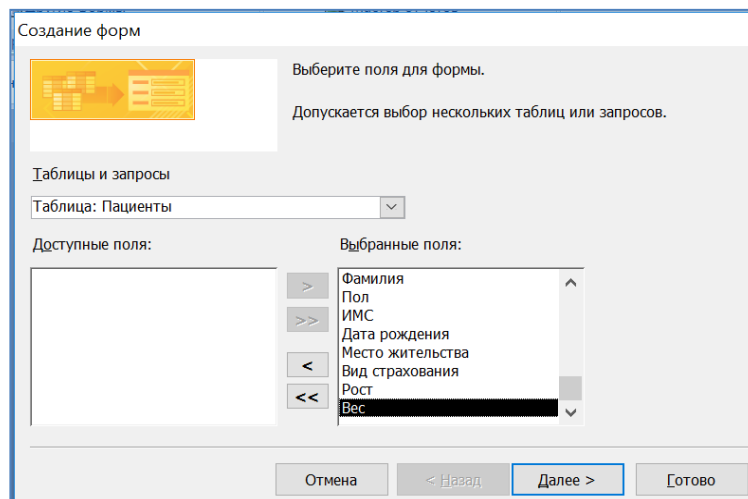
В зависимости от поставленной задачи Access предлагает разные по сложности варианты создания экранной формы. Так, с помощью Мастера можно разработать

простую и надёжную форму в один столбец, которая облегчит неквалифицированному пользователю ввод новых данных в БД. Соответствующая процедура предполагает следующие шаги.

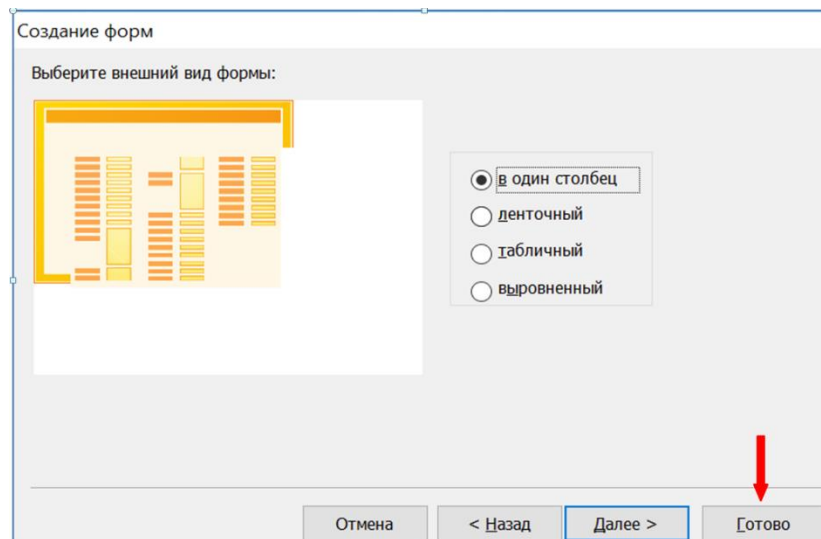
1. Выделить таблицу *Пациенты*, далее во вкладке *Создание* выбрать *Другие формы - Мастер форм*.



2. С помощью кнопки **>>** следует перенести все поля из списка **Доступные поля** в список **Выбранные поля**, щелкнуть по кнопке **Далее**.



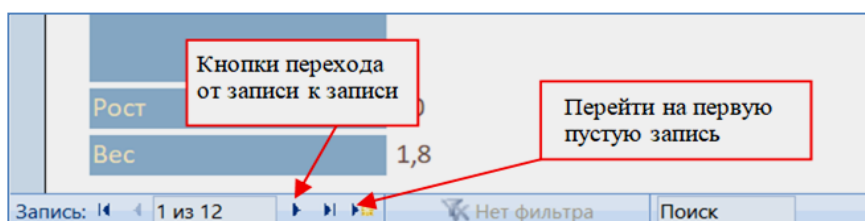
3. Выбрать тип формы в один столбец.



4. Выбрать стиль формы, пусть **Официальный**.
5. Задать имя формы. По умолчанию предлагается имя таблицы (или запроса), на которой основана форма. Лучше, конечно, предложить свой вариант, например, «Редактирование таблицы Пациенты». Тогда по имени формы можно определить цель её создания. **Готово**.

№ полиса	1
Фамилия	Иванов
Пол	м
Дата рождения	21.11.1952
Место жительства	Москва
Вид страхования	ОМС ДМС ММС
Вес	80
Рост	180

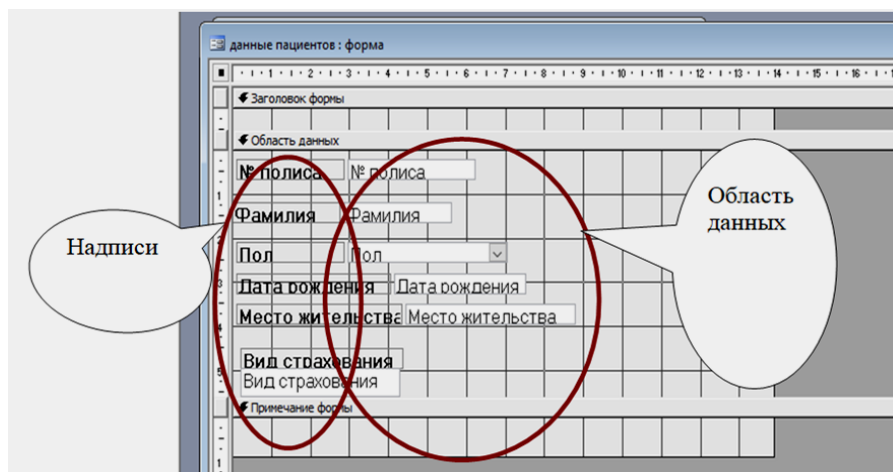
В результате получим бланк для редактирования каждой строки таблицы **Пациенты**, снабженный внизу удобными кнопками навигации по записям.



Основное назначение этой формы – введение новых и редактирование устаревших записей в имеющейся таблице. Теперь поддержание данных в актуальном состоянии можно доверить неквалифицированному персоналу: форма напрямую указывает для отобранных полей их имена, размер и варианты возможных значений (если здесь ранее было создано поле со списком). Такую форму следует сохранить в БД с подходящим именем для последующего регулярного использования.

Access сохраняет в базе данных только конструкцию формы, но не отображённые в ней данные. Изменение данных происходит в таблице, а не в форме, которая использует эту таблицу лишь в качестве информационной основы.

При желании созданную форму можно легко доработать в режиме Конструктора:



Название отображённых в Конструкторе областей напрямую указывают их назначение. Понятно, что в области заголовка следует поместить и красиво оформить название формы. Область данных состоит из парных прямоугольников, один из них определяет заголовок поля, другой – его содержимое. В примечании формы размещают дополнительную информацию, например, текущую дату, адрес, логотип и так далее. Помимо этих трёх основных структурных составляющих в конструкторе формы можно отобразить области верхнего и нижнего колонтитулов (выполнить щелчок в заголовке формы и из контекстного меню выбрать «колонтитулы страницы»).

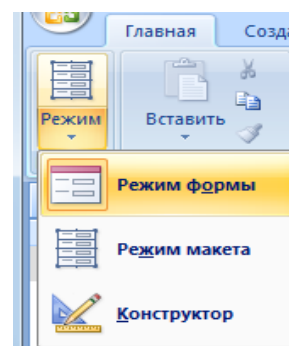
Содержимое заголовка и области примечания не изменяется при перемещении по записям, в отличие от области данных.

Представления экранной формы различаются в зависимости от вида выполняемых работ:

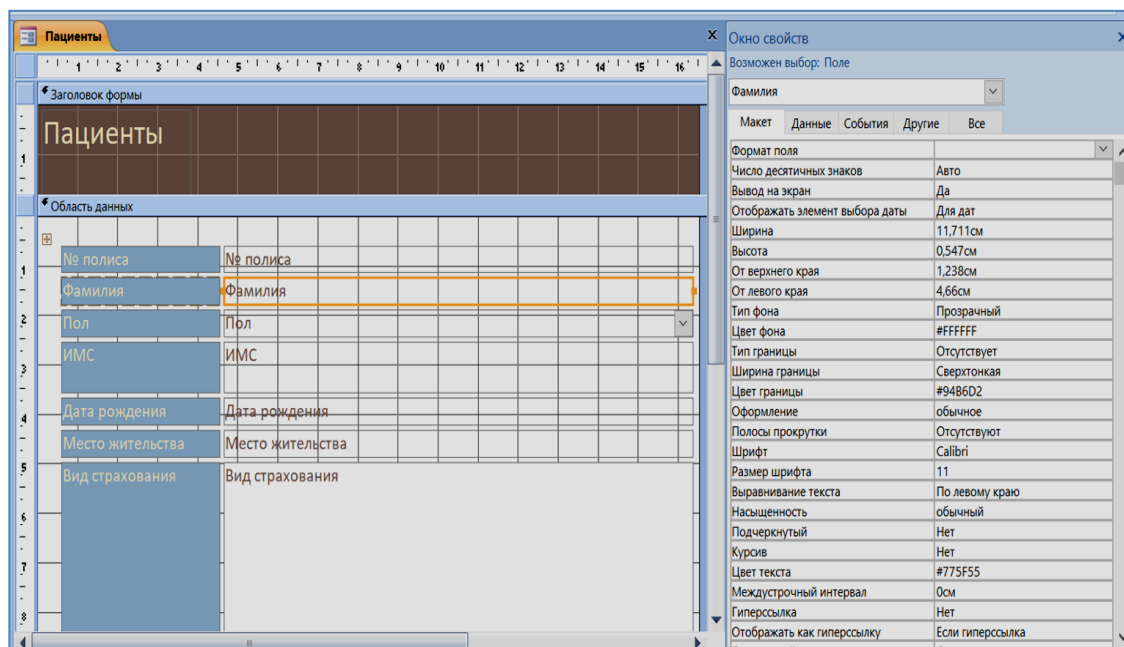
– *режим конструктора* – для создания собственной экранной формы с заданными свойствами, которые реализуют с помощью встроенных в форму элементов управления;

– *режим формы* – это рабочее состояние формы для выполнения операций добавления, удаления, обновления и отображения данных;

– *режим макета* – для редактирования и форматирования формы.



Так, в режиме макета можно подобрать шрифт, цвет, размер символов в надписях и в области данных, которые обозначены на рисунке выше. Можно также изменять размеры областей надписей и данных и их расположение, но эти изменения будут применяться ко всем элементам, входящим в макет.



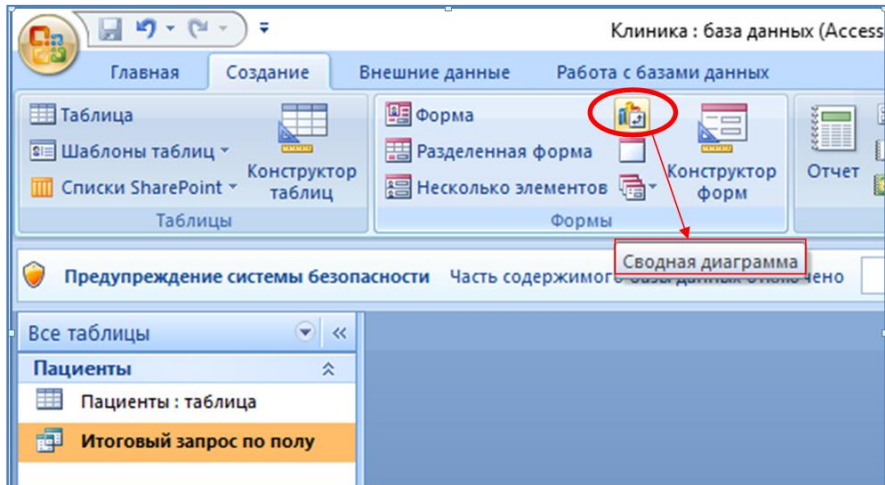
Самый простой путь форматирования формы – применение автоформата (**Работа с макетами форм – Формат - Автоформат**). В этой группе размещена галерея готовых форматов, которые можно использовать для преобразования внешнего вида формы: откройте галерею щелчком по стрелке внизу справа, а затем щелкните по подходящему формату. И он применится к форме.

Создание формы с диаграммой позволяет представить числовые данные, содержащиеся в форме, в наглядном графическом виде.

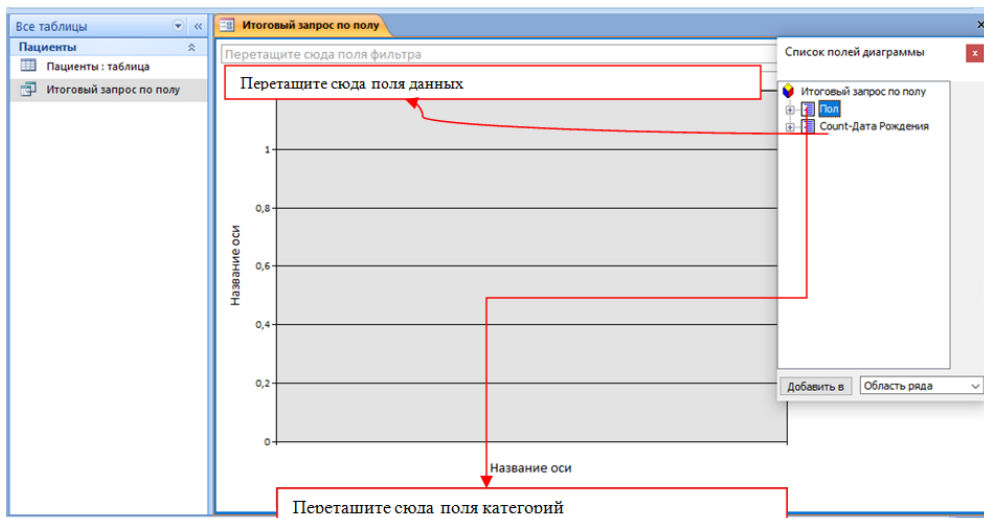
Технология построения диаграммы не отличается принципиально от той, которая используется в EXCEL: перед созданием диаграммы нужно указать содержание каких полей таблицы или формы будут использованы. По умолчанию значения столбца образуют на диаграмме *ряд данных*, а значение строки – *категории*. Можно указать несколько рядов и категорий.

Для простоты рекомендуем сначала заготовить подходящий по условию задачи запрос, затем на его основе оформить диаграмму.

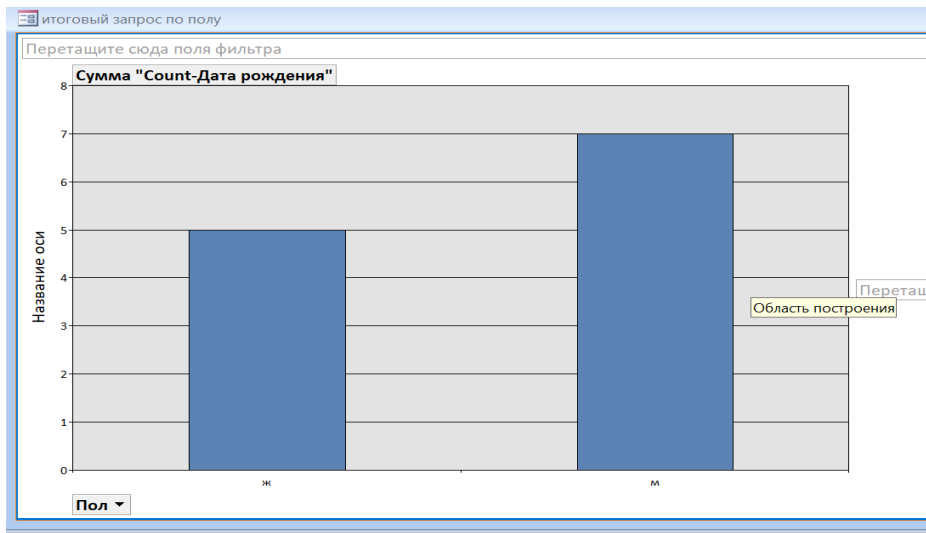
Построим, например, диаграмму, которая визуально сопоставляет количество пациентов мужского и женского пола. Соответствующий запрос уже есть в базе данных – это «Итоговый запрос по полу», поэтому выделим его в списке объектов БД, затем на вкладке **Создание** используем кнопку «Сводная диаграмма».



В полученное рабочее окно нужно перетащить поле *Пол* в область оси категорий, а поле *Count-Дата Рождения* в область для рядов данных:



В результате получаем гистограмму, которая наглядно отображает количество пациентов мужского и женского пола:



Имя форме с диаграммой задается, как обычно, при закрытии объекта, по умолчанию это будет имя запроса, на основании которого была построена диаграмма.

Результат произведённых действий удобно просматривать в режиме Формы, а редактировать в режиме Конструктора.

Замечание. После ознакомления с экранной формой можно отметить для себя, что вводить данные в БД можно двумя способами: непосредственно в таблицу и посредством формы. Выбор определяется частотой обновления информации. А именно, если данные изменяются редко, то для ввода данных лучше использовать режим таблицы. Напротив, при частом изменении вводимых данных лучше поместить их в форму, которая обеспечивает концентрацию внимания на содержание записей.

5. Генерация отчётов

Отчет создают в тех случаях, когда информацию из БД нужно систематизировать и подготовить для передачи заинтересованным лицам в электронном (рассылка по сети) или бумажном виде. В отличие от формы отчёт не только извлекает данные из БД, но структурирует и анализирует их, отображая исходно “сырые” данные в виде итоговой сводки. Формально это значит, что отчет имеет следующие характерные элементы:

- 1) заголовок отчёта, который отражает цель его создания;
- 2) шапка отчета из последовательности имен полей БД, включенных в отчет;
- 3) содержательная часть, которая представляет частично или полностью данные из БД, организованные в соответствии с шапкой отчета;
- 4) как правило, итоговые строки типа “в целом за отчетный период...”, “общее количество больных...” и тому подобное;
- 5) нумерацию страниц отчета, дату его создания, почтовые атрибуты организации.

Подведение итогов предполагает группировку данных из БД по выбранному признаку (имени поля). Например, пусть больничный стационар должен отчитаться по количеству госпитализированных пациентов (или по числу свободных койко-мест, или расходу лечебных препаратов...) за каждый год своего существования. Для этого нужно сначала разбить список всех пациентов на группы, каждая из которых перечисляет больных с одним и тем же годом госпитализации (*сгруппировать пациентов по полю «год поступления»*). Потом остаётся только посчитать число представителей в каждой группе (*определить суммарный подитог по полю «год поступления»*). Тогда в конце отчёта естественно вывести общее число пациентов за все годы (*общий суммарный итог по полю «год поступления»*).

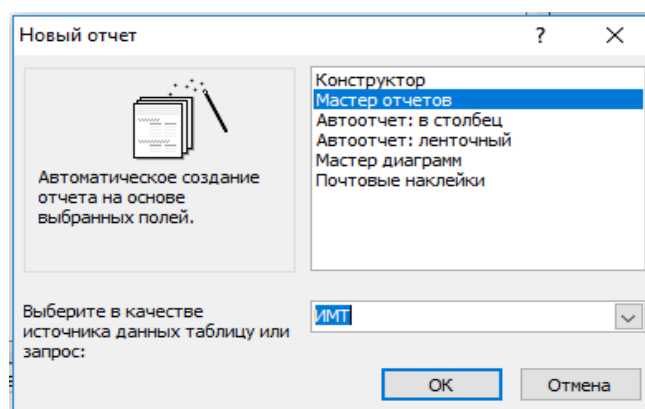
Таким образом, отчеты позволяют представить и распечатать данные в соответствии с принятой в документообороте структурой, а именно, «подведение текущих подитогов и общих итогов» по заданному признаку.

Отчеты очень похожи на формы. Так, режимы Конструктора форм и отчетов почти одинаковы. Однако есть принципиальные отличия. Отчеты предназначены исключительно для вывода данных на печать, поэтому для них нельзя установить режим таблицы или формы, а можно выбрать только режимы Конструктора и Предварительного просмотра.

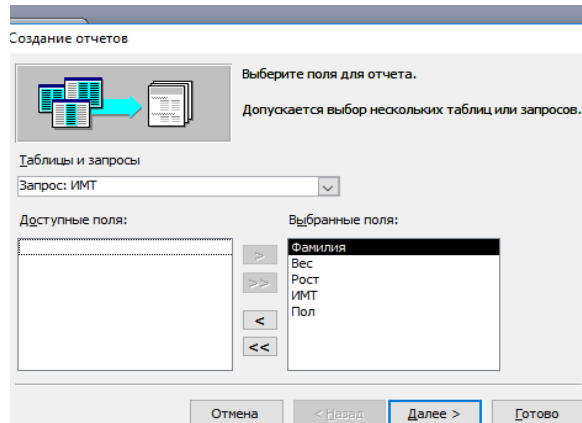
Сравним для примера степень ожирения (через ИМТ) для мужчин и женщин. Подготовим для этого данные: в имеющийся в базе запрос «ИМТ» введём (с помощью Конструктора) дополнительное поле *Пол*, сохраним запрос с тем же именем «ИМТ».

Фамилия	Вес	Рост	ИМТ	Пол
Иванов	80	1,8	24,691359333	м
Петрова	70	1,6	27,343749185	ж
Сидоров	95	1,8	29,320989208	м
Березина	65	1,7	22,491348219	ж
Дубов	85	1,8	26,234569291	м
Кошкин	75	1,6	29,296874127	м
Серов	90	1,9	24,930748548	м
Волкова	60	1,6	23,437499302	ж
Собакин	100	1,8	30,864199166	м
Лисицын	110	1,7	38,062281602	м
Соловьёва	60	1,5	26,666666667	ж
Антонова	80	1,5	35,555555556	ж

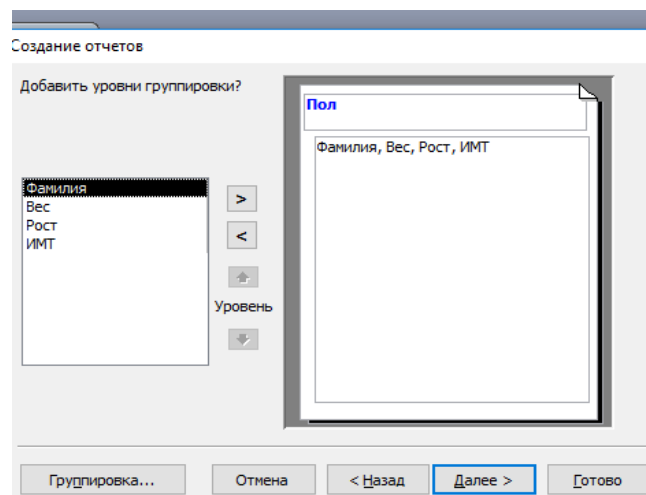
Начнём с простого варианта, то есть оформим отчёт с помощью Мастера. Для этого в списке объектов БД выделим запрос «ИМТ» в качестве источника данных, затем вкладка *Создание - Мастер отчетов*.



На первом шаге мастера следует перенести нужные поля из списка **Доступные поля** в список **Выбранные поля**, и щелкнуть по кнопке **Далее**:

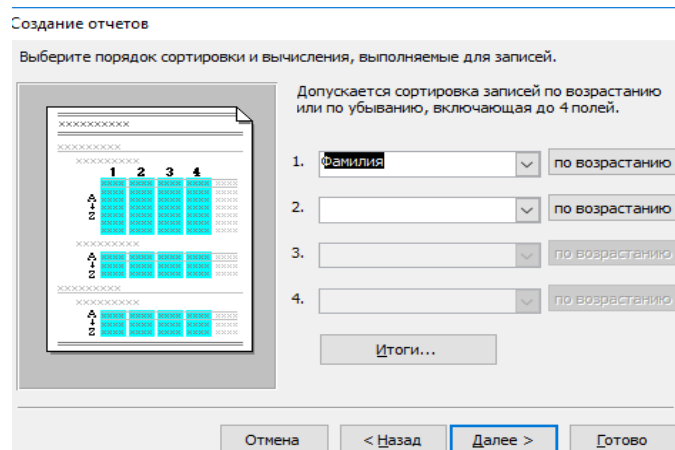


На следующем шаге Мастер предлагает указать поле группировки, в нашем примере *Пол*.



В результате список пациентов будет разбит на две группы: женщин и мужчин. В каждой группе будут приведены следующие данные пациентов: фамилия, вес, рост, ИМТ.

Следующий диалог устанавливает сортировку (пусть по фамилии, то есть в алфавитном порядке) и заказ итоговых вычислений (кнопка Итоги):



Мастер предложит список полей, для которых имеет смысл вычисление итогов, заказываем оценку среднего (Avg – от average) для числовых данных поля *ИМТ*:

Итоги

Какие итоговые значения необходимо вычислить?

Поле	Sum	Avg	Min	Max
Вес	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Рост	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ИМТ	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Показать

данные и итоги

только итоги

Вычислить проценты

OK

Отмена

Осталось подобрать элементы форматирования и указать заголовок отчёта –
Готово.

средний ИМТ по полу

<i>Пол</i>	<i>Фамилия</i>	<i>Вес</i>	<i>Рост</i>	<i>ИМТ</i>
<i>ж</i>				
	Антонова	80	1,5	35,555556
	Березина	65	1,7	22,4913482
	Волкова	60	1,6	23,4374993
	Петрова	70	1,6	27,3437492
	Соловьёва	60	1,5	26,6666667
<i>Итоги для 'Пол' = ж (5 записей)</i>				
Avg				27,0989638
<i>м</i>				
	Дубов	85	1,8	26,2345693
	Иванов	80	1,8	24,6913593
	Кошкин	75	1,6	29,2968741
	Лисицын	110	1,7	38,0622816
	Серов	90	1,9	24,9307485
	Сидоров	95	1,8	29,3209892
	Собакин	100	1,8	30,8641992
<i>Итоги для 'Пол' = м (7 записей)</i>				
Avg				29,0572888

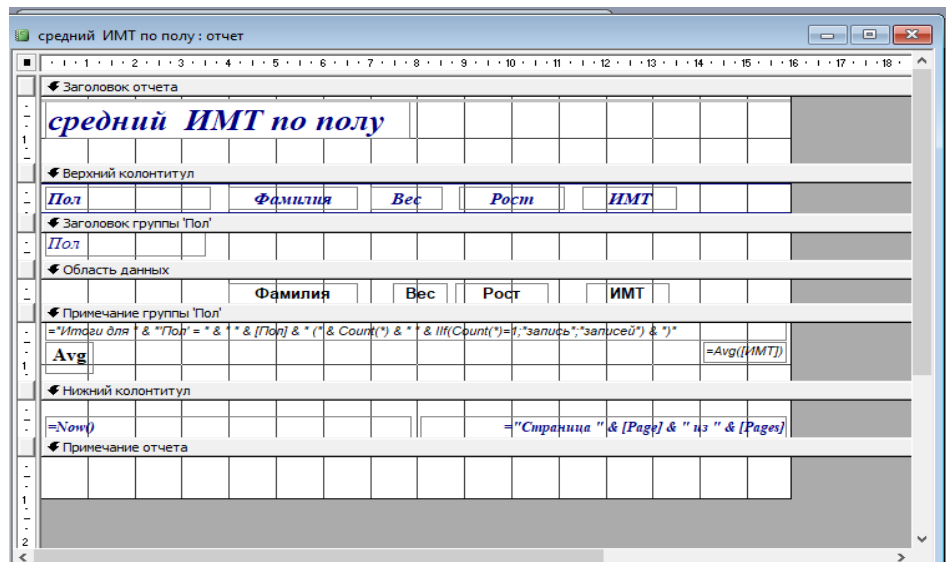
Из приведённых итоговых данных видно, что индекс массы тела у женщин оказался в среднем ниже, чем у мужчин. Дальше нужно оценивать результаты персонально: если у конкретного пациента ИМТ находится в пределах от 25 до 30, то налицо избыток масса тела. А избыточная масса тела увеличивает риск возникновения сердечнососудистых заболеваний и увеличивает нагрузку на все органы.

При значениях ИМТ свыше 30 можно говорить об ожирении. Ожирение провоцирует целый ряд серьёзных заболеваний, таких как артрит, диабет, различные

опухоли. В частности, при ожирении риск летального исхода от сердечнососудистых заболеваний увеличивается на 30%.

Таким образом, данные приведённого отчёта наглядно демонстрируют результаты анализа информации из БД и вполне пригодны для содержательного заключения.

Любой готовый отчет можно открыть и отредактировать в режиме Конструктора:



Из приведённого рисунка видно, что структуру как формы, так и отчёта в Конструкторе представляют одни и те же характерные области:

- *заголовок отчета* расположен на первой странице и выводится только один раз в самом начале отчета;
- *верхний колонтитул* обычно содержит заголовки колонок (имена полей), то есть представляет шапку отчета, которая выводится в верхней части каждой страницы;
- *заголовок группы* содержит имя поля группировки, в нашем примере это *Пол*,
- *область данных* содержит собственно записи, то есть значения полей *Фамилия*, *Вес*, *Рост*, *ИМТ*;
- *нижний колонтитул* по умолчанию содержит нумерацию страниц отчета и дату его генерации, выводится в конце каждой страницы;
- *примечание отчета* содержит какую-то итоговую информацию и выводится только один раз в конце отчета (перед нижним колонтитулом последней страницы).

Границы всех перечисленных областей можно перетаскивать мышью, если запланированное содержание не помещается в стандартный размер.

Напротив, если какая-то область отчёта оказалась ненужной (простой отчёт в одну страницу, например), то такую область можно скрыть, сдвинув её границы до нулевой ширины.

Существенно улучшить внешний вид отчета можно в режиме макета (воспользоваться кнопкой «Режим» на вкладке *Главная* и переключиться в режиме Макета). Стиль работы по оформлению и украшению отчёта в режиме Макета полностью аналогичен правке внешнего вида формы. Прежде чем вносить изменения, выделите нужный объект на макете, щелкнув по нему. Вокруг этого объекта появится сплошная рамка, а вокруг всего поля – пунктирная рамка. Далее можно выполнять коррекцию выделенного объекта.

Чтобы посмотреть, как будет выглядеть отчет после форматирования, нужно переключиться в режим *Представление отчета* (кнопка «Режим» на вкладке *Формат*). При необходимости можно вернуться обратно в режим Макета и внести изменения, либо отредактировать отчет в режиме Конструктора.

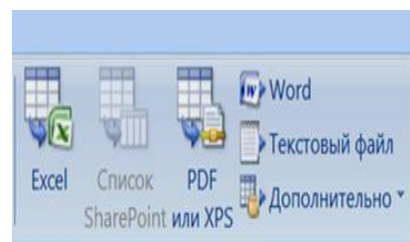
На вкладке *Формат* имеются инструменты для форматирования значений полей, для изменения внешнего вида всего макета, для добавления таких элементов, как эмблема, заголовков, номер страницы, итоговые поля.

В частности, завершить формирование отчета можно введением в него привычного элемента – адреса фирмы (можно логотип, структурное подчинение...). Для этого в области нижнего колонтитула формируем надпись (кнопка **Aa** на панели инструментов):

Наш адрес: улица Большая Пироговская, 2, кор. 2

Печать отчета можно выполнить как из окна БД, так и из окна Просмотра: *Режим – Предварительный просмотр*. Вкладка *Параметры страницы* этого же окна отвечает за форматирование печатной страницы.

В правой части окна предварительного просмотра размещены кнопки, с помощью которых можно выполнить экспорт отчёта в другие приложения Microsoft Office. Действительно, в практике делового оборота документов довольно часто возникает необходимость публикации отчета в качестве составной части документа Microsoft Word.



6. База данных из двух взаимосвязанных таблиц

Реальные базы данных представляют собой *совокупность взаимосвязанных* таблиц, между которыми разумным образом распределены данные.

Такой подход к организации БД представляется оптимальным по многим причинам: 1) каждая таблица в БД должна отвечать одной определенной тематике. Действительно, разные стороны (т.е. таблицы) моделируемого в БД объекта могут существенно отличаться друг от друга как по структуре, так и по содержанию. 2) распределение информации по нескольким таблицам исключает дублирование данных. Тогда при редактировании каждое исправление или дополнение можно производить только в одном месте базы данных. Кроме того, исключение дублирования приводит к сокращению (в реальных базах данных весьма значительному) объёмов оперативной памяти компьютера. И вообще, исключение дублирования данных является один из главных принципов технологии баз данных. 3) разнесение информации по таблицам отвечает принципу модульности в программировании: отдельный модуль решает свою конкретную задачу, поэтому каждый из них можно модернизировать и заменять по отдельности. Кроме того, таблицы-модули, подготовленные для одной базы данных, можно вставлять (копировать) в другие базы данных, схожие по тематике с первой.

Так, в нашем учебном примере БД **СТАЦИОНАР** мы предполагаем, как минимум, две составные части: **Пациенты** и **Госпитализация**. Понятно, что соответствующие им таблицы будут иметь разную структуру и разное наполнение – слишком разные сущности они представляют. С другой стороны, и **Пациенты**, и **Госпитализация** представляют две стороны одного итого же объекта - **СТАЦИОНАРА**. Следовательно, данные в этих таблицах *связаны* между собой по сути моделируемого объекта.

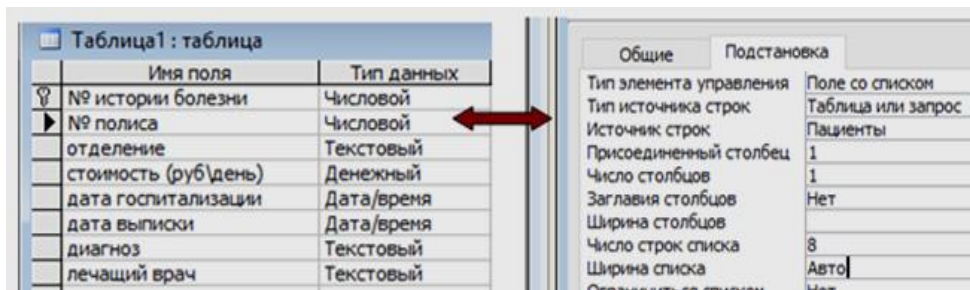
Средствами СУБД Access создадим таблицу **Госпитализация**. В ней будут указаны: номер истории болезни (*поле № истории болезни*), номер полиса пациента (*№ полиса*), в какое отделение он был госпитализирован (*Отделение*), стоимость одного дня пребывания в этом отделении (*Стоимость (руб.\ день)*), дата госпитализации (*Дата госпитализации*), дата выписки (*Дата выписки*), диагноз при госпитализации (*Диагноз*), лечащий врач (*Лечащий врач*).

Таблица 5

Госпитализация

<i>Имя поля</i>	<i>Тип данных</i>	<i>Размер поля</i>
№ истории болезни	Числовой	Целое
№ полиса	Числовой	Целое
Отделение	текстовый	15
Стоимость (руб\ день)	Денежный	Основной
Дата госпитализации	Дата/Время	Краткий формат даты
Дата выписки	Дата/Время	Краткий формат даты
Диагноз	текстовый	20
Лечащий врач	текстовый	15

С помощью Конструктора реализуем схему отношения **Госпитализация** в БД **СТАЦИОНАР**. Заметим, что номер полиса медицинского страхования (поле *№ полиса*) уже присутствует в таблице **Пациенты**. Поэтому при описании свойств этого поля в таблице **Госпитализация** выберем вариант поля подстановки, а в качестве источника строк укажем ранее созданную таблицу **Пациенты**.



Присутствие в таблице **Госпитализация** ключевого поля «№ полиса» из таблицы **Пациенты** отображает классический подход к оформлению межтабличной связи в реляционных базах данных. В этой «связке» таблица **Госпитализация** является подчинённой (*дочерней*), так как она содержит чужой ключ «№ полиса» (его называют *внешний ключ*). Следовательно, таблица **Пациенты** будет главной (*материнской*), так как она поставляет свой первичный ключ в дочернюю таблицу для организации межтабличной связи.

Связанные поля могут иметь различные имена, однако они должны содержать данные одного типа с одинаковыми свойствами (в особенности размер поля). При наличии связи между таблицами Access будет автоматически выбирать данные из обеих таблиц.

Таким образом, установление связи между таблицами приводит к тому, что при построении запросов, отчетов, форм можно выбирать данные из нескольких источников, объединённых фактом связи.

Наполним подходящими данными таблицу **Госпитализация**. При этом, чтобы отличать визуально первичные ключи разных таблиц, значения ключевого поля таблицы **Госпитализация** *№ истории болезни* обозначим двухзначными целыми числами.

Для простоты рассмотрим только три отделения: хирургия, терапия и неврология.

Чтобы не ограничивать общность, будем считать, что некоторые пациенты могли быть госпитализированы несколько раз. Этот факт найдёт проявление в том, что номера полисов таких пациентов в таблице **Госпитализация** будут повторяться несколько раз. Так, в нашем примере Иванов (№ полиса 1) и Серов (№ полиса 7) находились на излечении по три раза каждый.

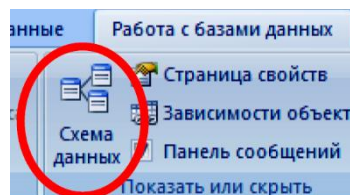
№ истории бол	№ полиса	отделение	стоимость (руб)	дата госпитали	дата выписки	диагноз	лечащий врач
11	2	хирургия	800	22.12.2017	05.01.2018	грыжа	Посев
12	7	терапия	500	11.01.2018	02.02.2018	ХОБЛ	Медведев
13	5	неврология	1000	08.01.2018	20.01.2018	остеохондроз	Зайцев
14	1	хирургия	800	03.02.2018	23.02.2018	грыжа	Посев
15	7	хирургия	800	22.02.2018	02.03.2018	вывих ноги	Волков
16	1	неврология	1000	01.03.2018	22.03.2018	инсульт	Зайцев
17	11	терапия	500	02.01.2018	31.01.2018	язва ДПК	Медведев
18	1	неврология	1000	25.03.2019	06.04.2018	инсульт	Зайцев
19	7	неврология	1000	01.03.2018	15.03.2018	остеохондроз	Зайцев
20	12	терапия	500	13.02.2018	02.03.2018	бронхит	Медведев
21	8	терапия	500	03.02.2018	01.03.2018	стенокардия	Медведев
22	3	хирургия	800	31.01.2018	12.02.2018	перелом руки	Волков
23	10	неврология	1000	03.02.2018	04.03.2018	инсульт	Зайцев
▶	0		0				

Наполнение таблицы **Госпитализация** данными можно заметно ускорить, если помимо поля *№ полиса* использовать подстановку в полях *отделение* и *лечащий врач* - там много повторных значений, варианты которых следует перечислить в списке источника строк (по аналогии с полем *Пол* в таблице **Пациенты**).

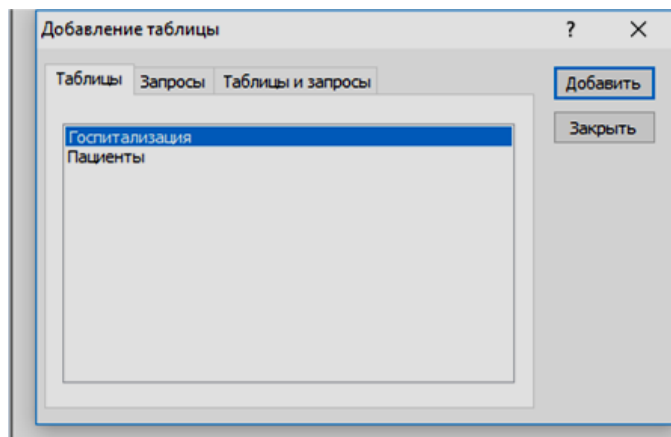
6.1. Установление связи между таблицами базы данных

Перейдём к процедуре связывания таблиц **Пациенты** и **Госпитализация**, но! перед этим их следует закрыть.

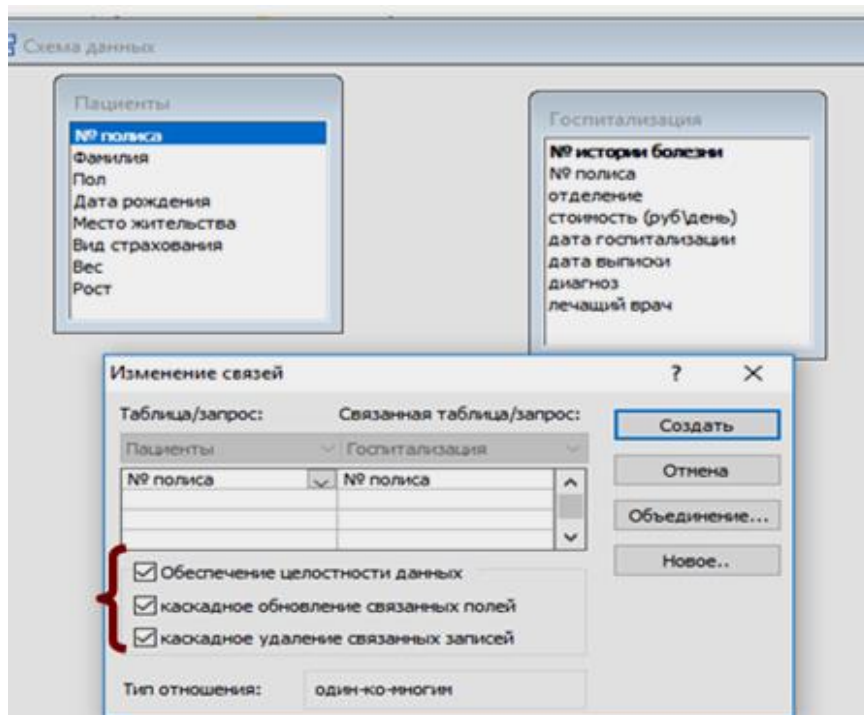
Установить связь между таблицами в первый раз, а также просмотреть и отредактировать ранее установленные связи можно в окне **Схема данных** (на вкладке **Работа с базами данных** щелкнуть по кнопке «Схема данных»).



Если связь устанавливают первично, то откроется окно «Добавление таблицы»:



Следует поочередно выделить и добавить имена таблиц **Пациенты** (сначала материнская), затем **Госпитализация**, затем кнопка «Заккрыть». В диалоге появятся два списка полей: один из таблицы **Пациенты**, другой из таблицы **Госпитализация**.

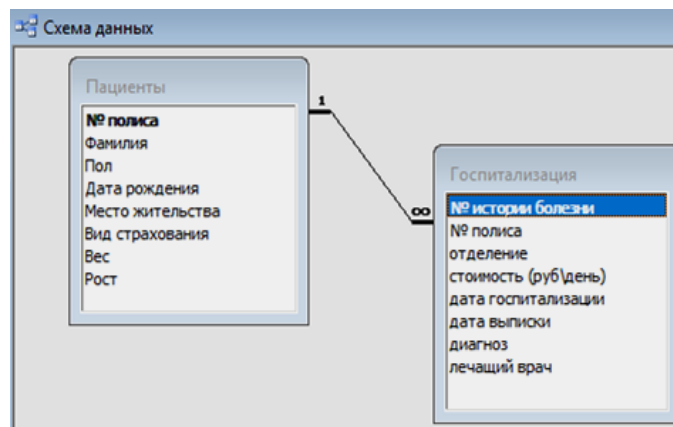


Захватим мышью поле *№ полиса* из материнской таблицы **Пациенты** и перетащим его на одноимённое поле в дочерней таблице **Госпитализация**. Поле *№ полиса* – это первичный ключ в таблице **Пациенты**, а в таблице **Госпитализация** оно же (*№ полиса*) – это внешний (вторичный) ключ. Ключ называется «внешний», так как пришёл извне, а «вторичный», так как у таблицы **Госпитализация** уже есть свой первичный ключ – это поле *№ истории болезни*.

Отпустим мышью – появится окно для указания параметров связи. А именно, следует установить флаги в опциях «Обеспечение целостности данных», «Каскадное обновление связанных полей» и «Каскадное удаление связанных полей». В результате СУБД будет автоматически контролировать ссылочную целостность внешних ключей, например, при удалении (выбыл!) некоторой записи из таблицы **Пациенты** будут удалены соответствующие записи в таблице **Госпитализация** с совпадающим значением в поле *№ полиса*.

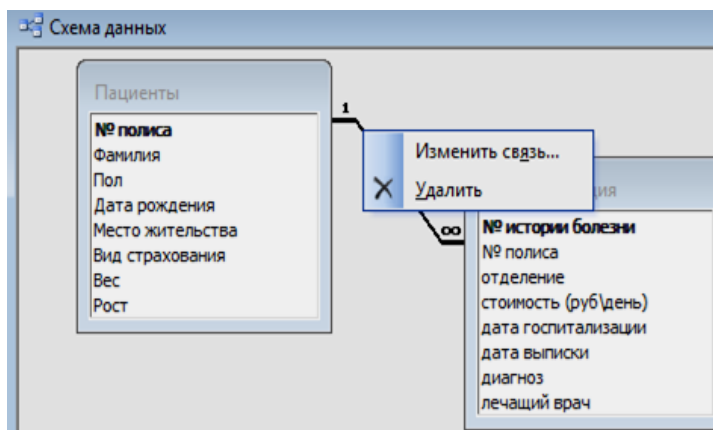
Внизу окна указан тип связи *один-ко-многим* (одной записи в таблице **Пациенты** могут соответствовать несколько записей в таблице **Госпитализация**). Действительно, так отображается факт возможной повторной госпитализации одного и того же пациента.

Фиксируем все заказанные параметры связи кнопкой «Создать» в окне «Изменение связи». В ответ получим графическое изображение связи между таблицами **Пациенты** и **Госпитализация** в окне Схемы данных.



Готовую схему данных закрываем, подтвердив при этом ее сохранение.

Иногда нужно отредактировать связь между таблицами. Для этого достаточно щёлкнуть мышью по линии связи и выбрать подходящий вариант (удалить или изменить связь).



Теперь наличие межтабличной связи можно проследить визуально в таблицах БД. Откроем с этой целью таблицу **Пациенты**. Отметим в ней некоторые изменения: слева от первого поля *№ полиса* появился дополнительный столбец с плюсиками – типичный признак скрытого списка. Раскроем щелчком несколько списков-плюсиков и увидим, что каждого пациента теперь сопровождает соответствующие ему записи из таблицы **Госпитализация**. Так, пациент Иванов был госпитализирован трижды, Сидоров – один раз, а Березина вообще не была госпитализирована. Очевидно, что каждая встроенная запись даёт полную информацию по параметрам госпитализации (отделение, диагноз, лечащий врач и так далее).

Пациенты : таблица								
	№ полиса	Фамилия	Пол	Дата рождения	Место жительства	Вид страхования	Вес	Рост
-	1	Иванов	м	21.11.1952	Москва	ОМС	80	1,8
	№ истории бол	отделение	стоимость (руб)	дата госпитали	дата выписки	диагноз	лечащий врач	
	14	хирургия	800	03.02.2018	23.02.2018	грыжа	Лосев	
	16	неврология	1000	01.03.2018	22.03.2018	инсульт	Зайцев	
	18	неврология	1000	25.03.2018	06.04.2018	инсульт	Зайцев	
	0		0					
+	2	Петрова	ж	12.01.1991	Сочи	ОМС	70	1,6
-	3	Сидоров	м	04.10.1988	Москва	ОМС	95	1,8
	№ истории бол	отделение	стоимость (руб)	дата госпитали	дата выписки	диагноз	лечащий врач	
	22	хирургия	800	31.01.2018	12.02.2018	перелом руки	Волков	
*	0		0					
-	4	Березина	ж	08.03.1955	Видное	ОМС	65	1,7
	№ истории бол	отделение	стоимость (руб)	дата госпитали	дата выписки	диагноз	лечащий врач	
*	0		0					
+	5	Дубов	м	15.05.1945	Берлин	ММС	85	1,8

Замечание на будущее. Важно отметить, что оптимальный путь связывания выполняется на пустых (не заполненных данными) таблицах. Такой вариант упрощает контроль ссылочной целостности данных. Например, добавление или удаление записей в одной из таблиц не приводит к появлению “осиротевших” записей в другой таблице (тех, которые ссылаются на несуществующую связанную с ней запись).

Подведём итог. В результате проделанной работы:

1. определены схемы отношений **Пациенты** и **Госпитализация** (обе таблицы описаны с точки зрения их структуры);
2. выстроена схема данных в БД (между таблицами установлена связь типа *один-ко-многим*);
3. база данных получила информационное наполнение (в таблицы **Пациенты** и **Госпитализация** были внесены записи).

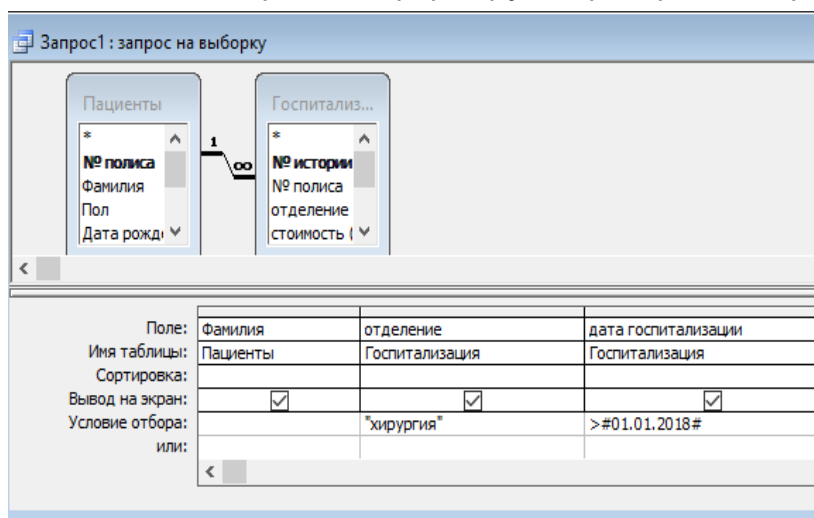
Вывод: для некоторого реального стационара была создана его компьютерная модель в виде базы данных **СТАЦИОНАР**. Далее можно приступить к её эксплуатации с целью удовлетворения информационных потребностей, которые возникают в процессе работы реального объекта - больничного стационара.

6.2. Операции над данными взаимосвязанных таблиц

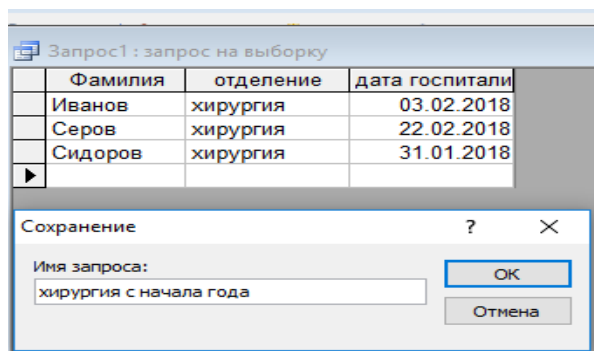
Установление связи между таблицами в БД приводит к организации единого информационного пространства, которое отображает в результате состояние дел во всех подразделениях предприятия. Формально связывание таблиц означает, что теперь в информационные запросы можно включать данные из нескольких таблиц одновременно. Продемонстрируем этот факт на примерах.

Запросы на выборку

Пусть была поставлена задача: получить список пациентов, госпитализированных в хирургическое отделение с начала 2018 года. Очевидно, что для реализации этого запроса потребуются данные как из таблицы **Пациенты** (*Фамилия*), так и из таблицы **Госпитализация** (*отделение, дата госпитализации*). Добавим обе таблицы в окно запроса и сформируем критерий отбора:



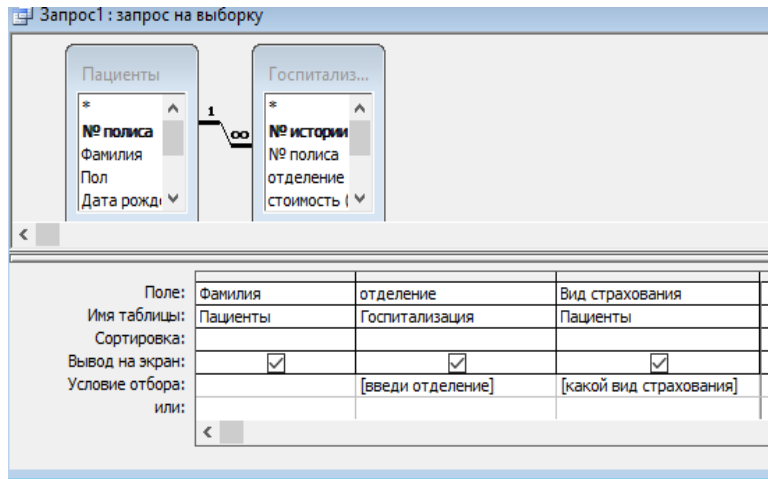
Выполним запрос и сохраним в базе данных для последующей эксплуатации.



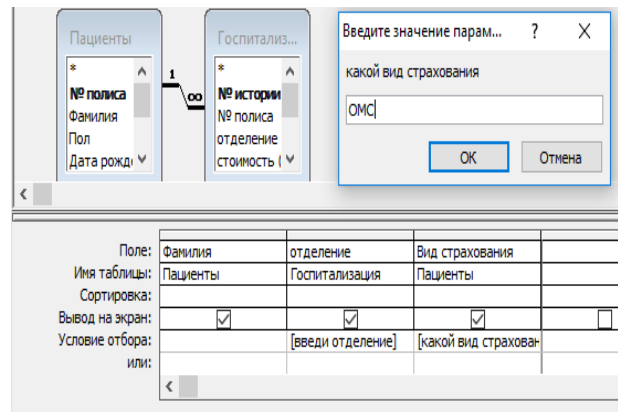
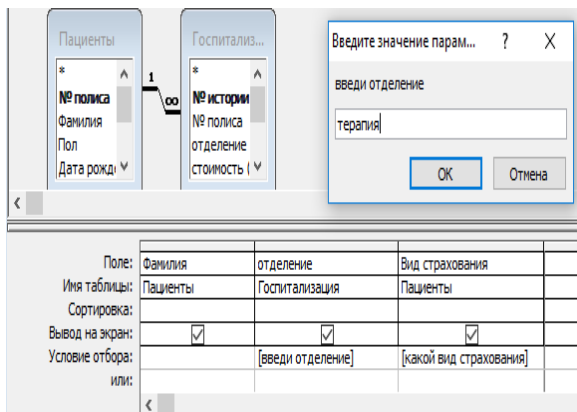
Аналогичным путём можно сформировать список пациентов, которые находились в *данном отделении* в период *между двумя указанными датами*.

Параметрические запросы на выборку

Протестируем параметрический запрос, в котором параметры относятся к разным таблицам. Например, получим список пациентов для заданного отделения (первый параметр) с определённым полисом страхования (второй параметр).



После запуска этого запроса на исполнение СУБД даст диалоговое окно для ввода названия интересующего нас отделения (пусть это будет терапия, например). Тут же придёт второй запрос на определение интересующего нас значения в поле *вид страхования* (пусть ОМС), и только после указания последнего параметра запрос будет исполнен.



Результат:

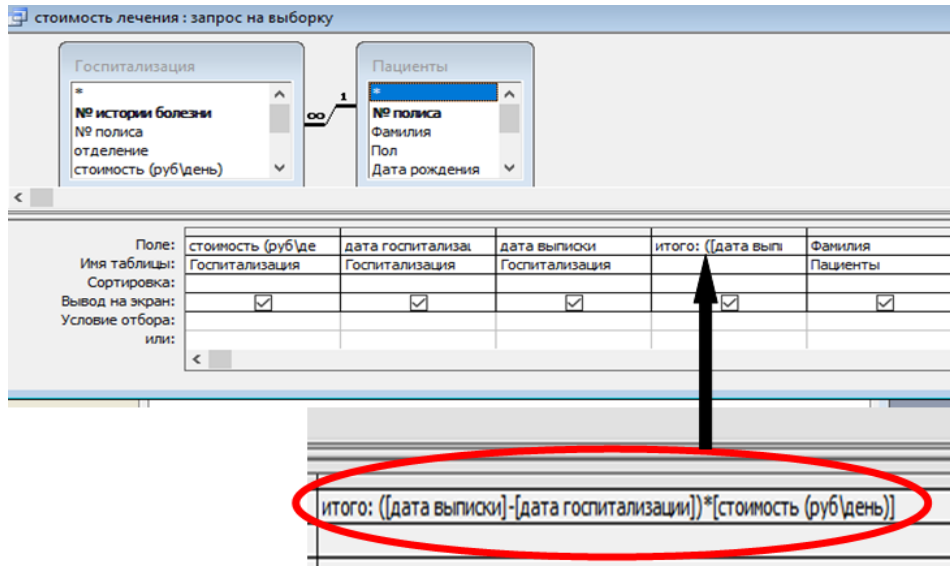
	Фамилия	отделение	Вид страхования
	Антонова	терапия	ОМС
	Волкова	терапия	ОМС

Аналогичным путём можно сформировать запрос на выдачу списка пациентов с данным видом страхования, которые были выписаны позже заданной даты (два параметра).

Запросы на выборку с вычисляемыми полями

Пусть поставлена задача на вычисление стоимости лечения пациентов. Для этого потребуются сформировать дополнительное поле (назовём его для краткости «итого»), в котором размер оплаты будет вычисляться по следующей формуле:

$$= (\text{дата выписки} - \text{дата госпитализации}) * \text{стоимость (руб/день)}$$



стоимость лечения : запрос на выборку

Госпитализация

- № истории болезни
- № полиса
- отделение
- стоимость (руб\день)

Пациенты

- № полиса
- Фамилия
- Пол
- Дата рождения

Поле: стоимость (руб\де

Имя таблицы: Госпитализация

Сортировка:

Вывод на экран:

Условие отбора:

или:

стоимость (руб\де	дата госпитализа	дата выписки	итого: ((дата вып	Фамилия
Госпитализация	Госпитализация	Госпитализация	Госпитализация	Пациенты
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

итого: ((дата выписки) - [дата госпитализации]) * [стоимость (руб/день)]

Результат:

стоимость (руб	дата госпитали	дата выписки	итого	Фамилия
800	03.02.2018	23.02.2018	16000	Иванов
1000	01.03.2018	22.03.2018	21000	Иванов
1000	25.03.2018	06.04.2018	12000	Иванов
800	22.12.2017	05.01.2018	11200	Петрова
800	31.01.2018	12.02.2018	9600	Сидоров
1000	08.01.2018	20.01.2018	12000	Дубов
500	11.01.2018	02.02.2018	11000	Серов
800	22.02.2018	02.03.2018	6400	Серов
1000	01.03.2018	15.03.2018	14000	Серов
500	03.02.2018	01.03.2018	13000	Волкова
1000	03.02.2018	04.03.2018	29000	Лисицын
500	02.01.2018	31.01.2018	14500	Соловьёва
500	13.02.2018	02.03.2018	8500	Антонова

Полученная таблица позволяет выставить счёт на оплату каждому пациенту, прошедшему госпитализацию.

Заметим, при наличии повторной госпитализации для одного и того же пациента иногда полезно вычислить суммарные затраты на лечение (для налоговой службы, например). С этой целью на базе полученной выше таблицы строят запрос с групповой операцией суммирования вслед за организацией групп по полю *Фамилия* и логическую операцию к оплате. Эту задачу оставляем для самостоятельной реализации.

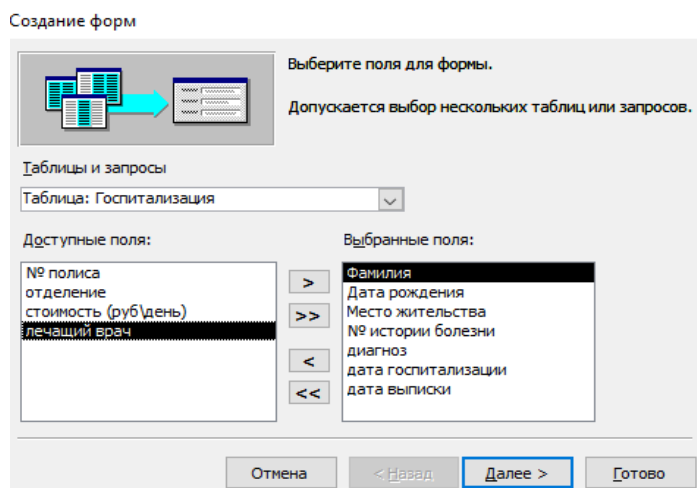
Создание составной (подчинённой) формы

От рассмотренной ранее простой формы составная форма отличается тем, что создается на основе связанных таблиц.

Пусть нужно сделать бланк справки, которая для любого пациента укажет причину (диагноз) и продолжительность госпитализации в стационаре. В такой форме будет задействован не один источник данных, а два, причём они не равноправные – есть главный (таблица **Пациенты**) и подчинённый ему источник данных (таблица **Госпитализация**). Отсюда и название – подчинённая форма. По виду табличных связей СУБД автоматически распределяет роли между таблицами-источниками данных.

Начинаем с команды **Создание формы с помощью мастера** на вкладке **Создание**.

В появившемся диалоге «Создание форм» производим отбор полей из таблиц-источников, сначала из материнской таблицы **Пациенты** (*фамилия, дата рождения, место жительства*). Далее нужно поменять источник: вместо имени материнской таблицы **Пациенты** в окошке «Таблицы и запросы» установим имя дочерней таблицы **Госпитализация** с последующим выбором из неё полей *№ истории болезни, диагноз, дата госпитализации, дата выписки*.



В результате выполненных действий СУБД автоматически определит составной тип создаваемой формы (индикация кнопкой «Подчиненные формы»): главная часть формы обусловлена таблицей **Пациенты**, а подчиненная – таблицей **Госпитализация**. Нажимаем кнопку «Далее».

Создание форм

Выберите вид представления данных:

- Пациенты
- Госпитализация

Фамилия, Дата рождения, Место жительства

№ истории болезни, диагноз, дата госпитализации, дата выписки

Подчиненные формы Связанные формы

Отмена < Назад Далее > Готово

Остаётся задать параметры оформления:

- внешний вид формы – табличный, «Далее»;
- стиль формы – стандартный, «Далее»;
- имя формы: **Справка**, «Далее».

Открываем форму для просмотра и ввода данных.

Справка

Фамилия: Волкова

Дата рождения: 04.10.1953

Место жительства: Сочи

Госпитализация

№ истории болезни	диагноз	дата госпитализации	дата выписки
21	стенокардия	03.02.2018	01.03.2018

Запись: 2 из 2

Запись: 8 из 12

Готовый проект составной формы представляет собой композицию двух бланков: верхний служит для указания данных пациента, а нижний предназначен для информации о причине и сроке госпитализации. Кроме того, в нижней части формы расположены две отдельные панели для навигации по записям из разных таблиц-источников.

Эту форму можно использовать при решении следующих задач эксплуатации БД:

- просмотра записей из двух таблиц (в том числе и автономного);
- печати заполненной данными справки о госпитализации;
- внесения новых данных из *одного* окна формы *по разным* источникам, СУБД автоматически распределит информацию по соответствующим таблицам.

Заметим, что более сложная форма предполагает предварительную заготовку подходящего запроса.

Ещё форму можно “украсить” с помощью Конструктора форм. Например, в приведённом выше примере можно дополнить форму «Справка» подходящим рисунком и указанием текущей даты и времени в области примечания формы. В простейшем варианте рисунок в форму добавляют с помощью кнопки «Вставка рисунка», расположенной на инструментальной ленте Конструктора форм. Далее следует указать нужную картинку из вашей коллекции изображений.

Создание отчёта по связанным таблицам

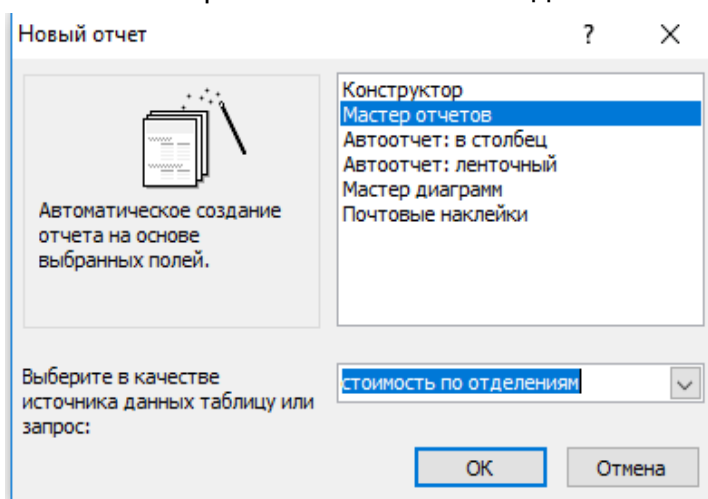
Отчёт по связанным таблицам требует предварительной подготовки. А именно, сначала рекомендуется создать запрос - информационную основу отчёта. Затем, вторым заходом, следует выстраивать задуманный (или заданный заказчиком) отчёт по изложенной в разделе 5 схеме. Предложенная стратегия избавляет пользователя от множества ошибок и тщетных попыток в построении информативного по содержанию и наглядного по форме отчёта.

Зададимся целью создания печатного отчёта, в котором указана суммарная выручка каждого отделения и общая выручка стационара за предоставленные им услуги.

В базе данных **СТАЦИОНАР** уже есть подходящий запрос с именем **Стоимость лечения**, нужно только отредактировать его, добавив туда название каждого отделения стационара.

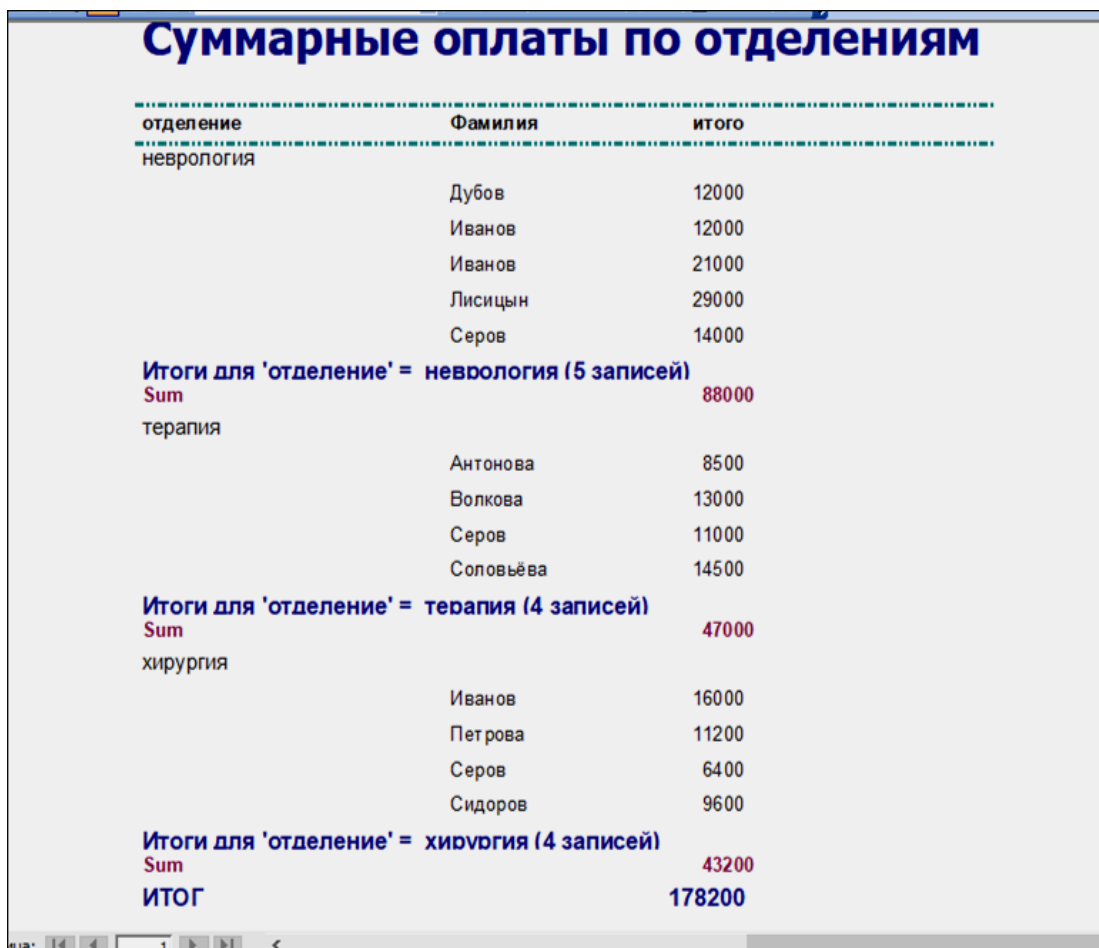
Воспользуемся стандартной процедурой копирования нужного запроса через буфер обмена. СУБД запросит новое имя для копии, пусть это будет **Стоимость по отделениям**. В режиме конструктора добавим сюда дополнительное поле *отделение* из таблицы **Госпитализация**. Тем самым мы подготовили запрос – основу отчёта, причём этот запрос извлекает данные из двух связанных между собой таблиц.

Запускаем Мастер отчётов (вкладка *Создание*), в качестве информационного источника отчёта указываем запрос «Стоимость по отделениям»:



Внесём в отчёт все нужные поля (*отделение, Фамилия, итого*), выполним группировку по полю *отделение*, а для суммарных вычислений укажем поле *итого*.

Полученный документ в целях наглядности отформатируем, увеличив размер, цвет и начертание шрифта, чтобы выделить зрительно итоговые строки отчёта.



The screenshot shows a report window titled "Суммарные оплаты по отделениям". The report contains a table with three columns: "отделение", "Фамилия", and "итого". The data is grouped by department. For each department, there is a list of patient names and their payment amounts, followed by a summary row labeled "Итого для 'отделение' = ..." and "Sum". The final row of the report is labeled "ИТОГ" and shows the total sum of 178200. The report is displayed in a window with a standard toolbar at the bottom.

отделение	Фамилия	итого
неврология	Дубов	12000
	Иванов	12000
	Иванов	21000
	Лисицын	29000
	Серов	14000
Итого для 'отделение' = неврология (5 записей)		
	Sum	88000
терапия	Антонова	8500
	Волкова	13000
	Серов	11000
	Соловьёва	14500
Итого для 'отделение' = терапия (4 записей)		
	Sum	47000
хирургия	Иванов	16000
	Петрова	11200
	Серов	6400
	Сидоров	9600
Итого для 'отделение' = хирургия (4 записей)		
	Sum	43200
ИТОГ		178200

Результат имеет ожидаемый вид: для каждого отделения указан список пациентов, сумма оплаты для каждого из них и общий итог по отделению (**Sum**). В последней строке отчёта приведена общая сумма выручки по стационару (**ИТОГ**).

На самостоятельное решение можно предложить аналогичную задачу, в которой сумма выручки стационара учитывает льготы по оплате лечения для некоторых категорий больных (пенсионеров, например). Потребуется, конечно, ввести в исходную таблицу **Пациенты** дополнительное поле *Скидка* для величины или процента скидки на стоимость лечения для разных категорий пациентов.

СЛОВАРЬ ТЕРМИНОВ И ОПРЕДЕЛЕНИЙ (ГЛОССАРИЙ)

- ASCII текстовый файл** — Файл, в котором применяются только символы кода ASCII (American Standard Code for Information Interchange – Американский Стандартный Код для Обмена Информацией), именно у этих файлов расширение *.TXT.
- CD-ROM** — Устройство для чтения компакт дисков. Имеет большую емкость (640 Мбайт). В основном применяется для хранения и переноса больших объемов информации (чаще в энциклопедиях и играх)
- Абзац** — Текст, находящийся между двумя нажатиями клавиши <ENTER>. Может иметь красную строку (отступ).
- Автоформат** – Встроенный набор форматов ячеек, таких как размер шрифта, узоры и выравнивание, которые можно применять к диапазонам данных. В Microsoft Excel автоматически определяются уровни итогов и подробных данных в выделенном диапазоне для применения к ним соответствующих форматов.
- Адрес** – Путь к объекту, документу, файлу, странице и т. п. Адрес может быть адресом URL (веб-адресом) или путем UNC (сетевым адресом), а также может включать указание на расположение в файле, например, закладку Microsoft Word или диапазон ячеек Microsoft Excel.
- Активация** – Выбор обычного листа или листа диаграммы для работы с ним. Тип активированного листа определяет отображаемые вкладки. Чтобы активировать лист, щелкните его ярлычок в книге.
- Активная ячейка** – – Выделенная ячейка, в которую будет помещаться текст при вводе с клавиатуры. В каждый момент времени активной может быть только одна ячейка. Активная ячейка выделяется жирной границей.
- Активное окно** — Окно документа или программы, в котором в данный момент идет работа. В активном окне документа находится точка вставки. Полоса заголовка активного окна подсвечивается.
- Активный лист** – Лист, на котором выполняется работа в книге. Имя этого листа на ярлычке выделено жирным начертанием.
- Анализ «что-если»** – Процесс изменения значений ячеек и анализа влияния этих изменений на результат вычисления формул на листе. Например, изменение процентной ставки, используемой в таблице амортизации для определения сумм платежей.
- Аргумент** – Значения, используемые функцией для выполнения операций или вычислений. Тип аргумента, используемого функцией, зависит от конкретной функции. Обычно аргументы, используемые функциями, являются числами, текстом, ссылками на ячейки и именами.
- Архивирование** – Процесс сжатия файлов с целью хранения их в более компактном виде. С технической точки зрения архивирование представляет собой анализ

значений и частоты появления байт в файле, выполняемый специальной программой-архиватором. Архивирование производится (в упрощенном изложении) примерно следующим образом. Например, для хранения строки "sssss" (длиной 5 байт, как вы понимаете) после архивирования потребуется всего лишь 2 байта: один – для запоминания символа "s" и второй – для запоминания количества его повторений. В результате такого анализа создается архивный вариант файла, гораздо более компактный по размеру.

Архивный файл – Набор из одного или нескольких файлов, помещенных в сжатом виде в единый файл, имеющий расширение по имени программы архиватора, например: Norton4.arj, файлы программы NC в архивированном виде.

База данных – Набор данных, относящихся к одной теме или использующихся для одной цели. В базе данных сведения об отдельных объектах, например о сотрудниках или заказах, сгруппированы по таблицам, записям и полям.

Базовый адрес – Относительный путь, используемый Microsoft Excel для определения конечного адреса при вставке гиперссылки. Базовым адресом может быть адрес в Интернете (URL), путь к папке на локальном жестком диске или сетевой путь.

Буфер обмена — Понятие Windows. Область памяти, в которую временно помещается вырезанный или скопированный фрагмент документа или графическое изображение при выполнении команд Копировать (Copy) или Вырезать (Cut). Содержимое Буфера обмена (Clipboard) можно вставить в тот же самый документ, либо в документ другой Windows-программы. Буфер обмена хранит информацию до следующей операции помещения в него данных, в течение одного сеанса работы с Windows. При выходе из Windows, а также копировании или вырезании нового фрагмента старое содержимое буфера обмена теряется. В Windows 95 действуют новые, гораздо более удобные и демократичные правила присвоения имен файлам и папкам. Теперь ограничение на длину имени файла составляет 255 символов (вместо 8). В имени можно использовать пробелы. Тем не менее сохранился запрет на использование следующих символов: " | " - вертикальная черта "\" - обратная черта ":" - двоеточие "?" - знак вопроса "*" - звездочка "\"" - кавычка "<" - треугольная скобка ">" - треугольная скобка. При использовании файла с длинным именем в программе, не являющейся приложением Windows 95, имя файла будет усечено до первых шести символов длинного имени без пробелов.

Вершины – Черные квадратные перетаскиваемые точки, отображающиеся на концах и в точках пересечения прямых и кривых линий в некоторых автофигурах, таких как полилинии, рисованные кривые и кривые.

Влияющая ячейка – Ячейка, на которую ссылается формула из другой ячейки. Например, если ячейка D10 содержит формулу =B5, ячейка B5 является влияющей на ячейку D10.

Внедренная диаграмма – Диаграмма, помещенная на обычный лист, а не на отдельный лист диаграммы. Внедренные диаграммы удобны для просмотра или

печати отчета сводной диаграммы вместе с исходными данными и другими сведениями, содержащимися на листе.

Внешняя ссылка – Ссылка на данные, помещаемые в документ из другой программы.

Внутреннее объединение – В Microsoft Query используемый по умолчанию тип объединения двух таблиц, при котором отбираются только записи, имеющие совпадающие значения в объединяемых полях обеих таблиц. В списке результатов отображается одна запись, получаемая при объединении двух совпадающих записей из разных таблиц.

Выбор – Выделение ячейки или диапазона ячеек на листе. Следующая команда или действие будут применены к выделенным ячейкам.

Выделение несмежных ячеек – Выделение двух и более ячеек или диапазонов, расположенных вразброс. При построении диаграммы по выделенным несмежным ячейкам убедитесь, что комбинированный выделенный блок имеет форму прямоугольника.

Выделение фрагмента — Операция пометки определенной части документа (например, фрагмента текста MS EXCEL или определенного количества смежных ячеек - интервала MS Excel - и т.д.), либо всего документа для последующего проведения действия именно с этой частью документа или выполнения команды, относящейся только к этой части документа (например, удаления, копирования). Выделенная часть документа помечается реверсивным способом - белым текстом на черном фоне. Выделение может производиться с помощью мыши или клавиатуры. Для выделения с помощью мыши необходимо установить курсор мыши в начало помечаемого фрагмента, нажать и удерживать ее левую клавишу, затем передвинуть мышь к концу выделения и отпустить клавишу. Для выделения с помощью клавиатуры необходимо установить курсор в начало помечаемого фрагмента, нажать и удерживать клавишу <Shift>, затем обозначить фрагмент с помощью клавиш управления курсором (со стрелками) и отпустить <Shift>.

Выражение – Комбинация операторов, имен полей, функций, литералов и констант, возвращающая одно значение. С помощью выражений могут задаваться условия (например, СуммаЗаказа>10000) или выполняться расчеты по значениям полей (например, Цена*Количество).

Вычисляемое поле (база данных) – Поле, заданное в запросе для вывода результата расчета выражения, а не для получения данных из базы.

Вычисляемое поле (сводный отчет) – Поле в сводном отчете или отчете сводной диаграммы, использующее созданную формулу. Вычисляемые поля могут выполнять вычисления, используя содержимое других полей сводного отчета или отчета сводной диаграммы.

Вычисляемый столбец – В таблице Microsoft Excel вычисляемый столбец использует одну формулу, применяемую к каждой строке. Он автоматически распространяется

на дополнительные строки, чтобы немедленно применить к ним соответствующую формулу.

Вычисляемый элемент – Элемент в поле сводной таблицы или в поле сводной диаграммы, использующий созданную формулу. Вычисляемые элементы могут выполнять вычисления, используя содержимое других элементов этого же поля сводного отчета или отчета сводной диаграммы.

Группа – В структуре или отчете сводной таблицы группой называется одна или несколько соседних строк или столбцов, подчиненных итоговой строке или столбцу.

Данные сводной таблицы – В сводном отчете итоговые данные, вычисленные на основе полей исходного списка или таблицы.

Данные структуры – Данные, содержащиеся в структуре листа. К данным структуры относятся строки и столбцы структуры.

Деления и подписи делений – Делениями называются маленькие метки единиц измерения, пересекающие ось и похожие на деления линейки. Подписи делений идентифицируют категории, значения и ряды на диаграмме.

Диалоговое окно — Окно, появляющееся на экране при выборе команды меню, содержащей после своего имени многоточие. Служит для указания дополнительных параметров, необходимых для выполнения данной команды.

Диапазон – Две или более ячеек листа. Ячейки диапазона могут быть как смежными, так и несмежными.

Диапазон внешних данных – Диапазон данных, представленных на листе, но расположенных вне Microsoft Excel, например в базе данных или текстовом файле. В Microsoft Excel эти данные можно форматировать и использовать в вычислениях, как и любые другие данные.

Диапазон ячеек, содержащий результаты подстановки различных значений в одну или несколько формул. Существует два типа таблиц данных: таблицы с одним входом и таблицы с двумя входами.

Дополнительные вычисления – Способ сложения значений в области данных сводного отчета путем использования значений из других ячеек области данных. Для создания дополнительных вычислений служит список в группе «Дополнительные вычисления» диалогового окна «Вычисление поля сводной таблицы».

Драйвер — Программа, управляющая каким-либо модулем компьютера или периферийным устройством (мышью, принтером, памятью и т. д.).

Другая папка автозагрузки – Папка, дополняющая стандартную папку автозагрузки XLStart, содержащую книги или другие файлы, которые должны открываться автоматически при запуске Microsoft Excel, и шаблоны, которые должны быть доступны при создании книг.

- Жесткий диск (Винчестер)** — Устройство компьютера, представляющее собой магнитный диск со считывающей, головкой и другими механическими компонентами, помещенное в металлический защитный корпус. На жестком диске хранится постоянная информация: программы, файлы данных и т. д. Емкость жесткого диска, как правило, весьма велика, но не безгранична: от 1 до 16 Гбайт.
- Зависимость** – Ситуация, в которой ячейка содержит формулу, ссылающуюся на другие ячейки. Например, если ячейка D10 содержит формулу =B5, ячейка D10 является зависимой от ячейки B5.
- Заголовки для печати** – Заголовки строк или столбцов, которые при печати выводятся вверх или слева на каждой странице.
- Заголовок столбца** – Помеченная буквами или цифрами затененная область, расположенная сверху каждого столбца. Чтобы выделить столбец, щелкните мышью его заголовок. Чтобы изменить ширину столбца, перетащите линию рамки справа от его заголовка.
- Заголовок строки** – Помеченная цифрами затененная область, расположенная слева от каждой строки. Чтобы выделить строку, щелкните мышью ее заголовок. Чтобы изменить высоту строки, перетащите линию рамки снизу от ее заголовка.
- Запрос** – В Microsoft Query или Microsoft Access средство поиска данных, отвечающих на определенный вопрос о данных, хранящихся в источнике данных.
- Запрос с параметрами** – Тип запроса, запрашивающий при выполнении один или несколько параметров (условий), так что один и тот же запрос может использоваться несколько раз для разных целей или в разных ситуациях.
- Значение R-квадрат** – Число от 0 до 1, которое отражает близость значений линии тренда к фактическим данным. Линия тренда наиболее соответствует действительности, когда значение R в квадрате близко к 1. Оно также называется квадратом смешанной корреляции.
- Значения** – Отдельные значения, отображаемые на диаграмме. Соседние значения образуют ряд данных. Значения отображаются в виде полос, столбцов, линий, секторов, точек или других объектов. Эти объекты называются маркерами данных.
- Идентификатор** – Имя поля, используемое в формулах. Например, «Сумма по доходу» является идентификатором (именем) поля, содержащего суммы по доходу. Вместо идентификатора можно использовать формулу (например, Цена*Количество).
- Имя** – Слово или строка знаков, представляющие ячейку, диапазон ячеек, формулу или константу. Понятные имена, такие как «Продукты», используют для ссылок на диапазоны, названия которых трудно запомнить, например, «Продажи!C20:C30».
- Имя файла** — Служит для обозначения и идентификации файла (документа, программы и пр.). Присваивается при первом сохранении нового документа.
- Индекс** – Компонент базы данных, повышающий скорость поиска данных. Если таблица имеет индекс, данные в таблице могут быть найдены путем просмотра индекса.

Интернет — Глобальная сеть (иначе объединение разнородных сетей). С помощью Интернет возможен доступ к мировым информационным ресурсам. Для подключения к Интернет по коммутируемой линии необходим модем.

Информация — Динамический продукт взаимодействия данных и адекватных им методов представления и обработки.

Информация, получаемая с помощью букв и цифр, называется символьной. Информация отличная от символьной (изображения, рисунки, фотографии, карты), называется видеоинформацией.

Исходные данные – Список или таблица, используемые для создания сводного отчета или отчета сводной диаграммы. Исходные данные могут быть взяты из таблицы или диапазона Microsoft Excel, внешней базы данных или куба либо из другого сводного отчета.

Исходные области – Диапазоны ячеек, консолидация которых выполняется в указанной конечной области. Исходные области могут располагаться на любом листе или книге, на других открытых листах или книгах, а также на листах Lotus 1-2-3.

Итоги – Один из пяти типов вычислений, возможных в Microsoft Query: сумма, среднее, количество, минимум и максимум.

Итоговая функция – Тип вычислений при объединении данных в сводном отчете, в таблице консолидации или при вычислении итогов в списках или базах данных. Примерами итоговых функций могут служить функции СУММ, СЧЕТ и СРЕДНЕЕ.

Итоговые данные – Все строки или столбцы, в которых итоговые данные подсчитываются по исходным данным. Итоговые данные обычно примыкают к соответствующим исходным данным.

Клавиши управления курсором — Клавиши со стрелками на клавиатуре, используемые для перемещения курсора и передвижения по документу, выделения его частей (при одновременно нажатой клавише <Shift>, обозначения точки вставки, а также работы с меню и указания команд.

Ключи легенды – Символы в легенде, показывающие узоры и цвета, соответствующие рядам (или категориям) данных на диаграмме. Ключи легенды отображаются слева от строк легенды. Форматирование ключа легенды также применяется и к связанному с ним маркеру данных.

Кнопка — Элемент экрана, как правило, диалогового окна или панели инструментов. Является графическим изображением команды и в большинстве случаев дублирует ее. Нажатие кнопки приводит к выполнению соответствующей команды. Например, кнопка ОК любого диалогового окна приводит к выполнению команды при заданных в окне установках и равнозначна нажатию клавиши <Enter>; а кнопка Копировать (Copy) стандартной панели инструментов равнозначна команде Копировать (Copy) меню Правка (Edit). Кнопки панелей инструментов содержат характерные графические образы, например, на кнопке Вырезать (Cut) изображены ножницы.

Нажатие на кнопку осуществляется путем установки на нее стрелки мыши и затем щелчком левой ее клавишей.

Кнопка «Выделить все» – Серый прямоугольник в левом верхнем углу листа, на пересечении заголовков строк и столбцов.

Конечная область – Диапазон ячеек, в который будут помещены итоговые данные после консолидации. Конечная область может располагаться как на том же листе, что и исходные данные, так и на другом листе. На одном листе может быть выполнена только одна консолидация.

Консолидации – Таблица объединенных результатов, появляющаяся в конечной области. В Microsoft Excel таблица консолидации создается путем применения итоговой функции к исходным значениям.

Константа – Постоянное (не вычисляемое) значение. Например, число 210 и текст «Квартальная премия» являются константами. Формула и результат вычисления формулы константами не являются.

Конструктор запросов – Все элементы в окне запроса, такие как таблицы, условия, порядок полей и т. п. Конструктор также определяет, включен ли автоматический режим и допускается ли изменение исходных данных.

Коридор колебания – На плоских графиках коридор колебания отображается линиями, идущими от максимального до минимального значения в каждой категории. Коридор колебания часто используется на биржевых диаграммах.

Корневой каталог — Каталог самого верхнего уровня, который создается при форматировании диска. В нем можно хранить файлы и создавать другие каталоги и подкаталоги. В отношении корневого все другие каталоги являются подкаталогами. Корневой каталог нельзя удалить.

Критерий сравнения – Набор ограничений, используемых для поиска данных. Критерием сравнения может быть набор знаков, которые требуется найти, например, «Предприятие 675»; или выражение, такое как «>300».

Легенда – Подпись, связывающая узоры или цвета с рядами или категориями данных на диаграмме.

Линии проекции – В линейчатых диаграммах и диаграммах с накоплением линиями проекции называются линии, идущие от точек данных к оси X (оси категорий). Удобны в диаграммах с областями для отделения одного маркера данных от другого.

Линии рядов – На плоской линейчатой диаграмме с накоплением и гистограмме линиями рядов называются линии, соединяющие маркеры данных в каждом из рядов данных и, таким образом, делающие разницу между рядами более заметной.

Линия объединения – В Microsoft Query это линия, соединяющая поля двух таблиц и показывающая связь данных для Microsoft Query. Тип объединения показывает, какие записи выбраны для включения в список результатов запроса.

Линия тренда – Графическое представление направления изменения ряда данных, например повышающаяся линия обозначает увеличение продаж за определенное количество месяцев. Линии тренда используются для анализа ошибок предсказания, что также называется регрессионным анализом.

Лист – Основной документ, используемый в Microsoft Excel для хранения данных и работы с ними. Он также называется электронной таблицей. Лист состоит из ячеек, упорядоченных в строки и столбцы. Листы всегда хранятся в книге.

Лист диаграммы – Лист книги, содержащий только диаграмму. Листы диаграммы позволяют просматривать диаграмму или отчет сводной диаграммы отдельно от данных листа или отчета сводной таблицы.

Лист изменений – Отдельный лист, содержащий список изменений в общей книге, включая имя пользователя, внесшего изменения, измененную ячейку, дату изменения, удаленные или измененные данные, а также сведения о разрешении конфликтов.

Маркер данных – Полоса, область, точка, сегмент или другой объект на диаграмме, соответствующий одному значению одной ячейки листа. Маркеры данных одного цвета на диаграмме образуют ряд данных.

Маркер заливки – Небольшой черный квадрат в правом нижнем углу выделенного блока. При наведении на маркер заливки указатель принимает вид черного креста.

Массив – Объект, используемый для получения нескольких значений в результате вычисления одной формулы или для работы с набором аргументов, расположенных в различных ячейках и сгруппированных по строкам или столбцам. Диапазон массива использует общую формулу; константа массива представляет собой группу констант, используемых в качестве аргументов.

Матрица – Прямоугольный массив значений или диапазон ячеек, объединенный с другими массивами или диапазонами для вычисления нескольких сумм или произведений. В Microsoft Excel имеются встроенные функции для работы с матрицами, осуществляющие сложение и умножение.

Мегабайт — Единица измерения объема информации (емкости памяти) в компьютерных системах. Сокращенно - Мбайт. 1 Мбайт равен 1 048 576 байт. Байт - одна из низовых специальных единиц измерения. 1 байт может содержать объем информации, равный одной букве или цифре. В свою очередь в 1 Байте всего 8 Бит (то есть восемь 0 или 1).

Многоуровневые подписи категорий – Подписи категорий на диаграмме, отображаемые в виде структуры в зависимости от данных в ячейках на листе. Например, заголовок «Фрукты» может отображаться над строкой с заголовками «Сливы», «Яблоки» и «Груши».

Надстройка Microsoft Excel – Компонент, который можно установить на компьютере для добавления команд и функций в Microsoft Excel. Такие надстройки

предназначены для использования только в Microsoft Excel. Другие надстройки для Microsoft Excel или Microsoft Office являются надстройками для модели COM.

Названия на диаграммах – Описательный текст, располагающийся вдоль осей или по центру в верхней части диаграммы.

Направляющие на диаграммах – Линии, которые можно добавить на диаграмму для улучшения восприятия и оценки отображаемых данных. Направляющие начинаются от делений на оси и пересекают область построения.

Неактивное окно — Любое открытое окно, в котором в данный момент не ведется работа. Заголовок неактивного окна не подсвечивается.

Неявное пересечение – Ссылка на диапазон ячеек вместо одной ячейки, из которого выбирается одна соответствующая ячейка. Например, если в ячейке C10 содержится формула =B5:B15*5, значение в ячейке B10 будет умножено на 5, потому что ячейки B10 и C10 находятся в одной строке.

Область вставки – Конечная область, в которую из буфера обмена Microsoft Office вставляются скопированные или вырезанные ячейки.

Область данных – Диапазон ячеек, содержащих данные и окруженных пустыми ячейками или границами листа.

Область диаграммы – Область размещения диаграммы и всех ее элементов.

Область значений – Часть сводного отчета, содержащая итоговые значения. Значение каждой ячейки области значений представляет собой итог данных, содержащихся в исходных записях или строках.

Область копирования – Ячейки, копируемые для вставки в другой области. После копирования ячеек вокруг них появляется бегущая рамка, показывающая, что они скопированы.

Область печати – Один или несколько диапазонов ячеек, выделенных для печати, если не требуется печатать весь лист. Если на листе задана область печати, будет напечатана только она.

Область построения – На плоских диаграммах ограниченная осями область, содержащая все ряды данных. На объемных диаграммах ограниченная осями область, содержащая все ряды данных, имена категорий, подписи делений и названия осей.

Область сводной таблицы – Область на листе, в которую перетаскиваются поля сводной таблицы или сводной диаграммы для изменения макета отчета. В новом отчете область сводной таблицы показана синей пунктирной рамкой.

Область условий – Область окна, в которой отображаются условия, используемые для ограничения записей, включаемых в список результатов запроса.

Обновление (сводный отчет) – Повторное построение сводного отчета или отчета сводной диаграммы для отражения изменения исходных данных. Если отчет создан на основе внешних данных, при обновлении будет запущен запрос на получение новых или измененных данных.

Общая книга – Книга, для которой настроена возможность одновременного просмотра и изменения по сети несколькими пользователями. Каждый пользователь, сохраняющий такую книгу, видит изменения, внесенные другими пользователями.

Общие итоги сводной таблицы – Итоговые значения для всех ячеек в строке или столбце сводного отчета. Значения в строке или столбце общих итогов вычисляются с использованием тех же итоговых функций, что и в области данных сводного отчета.

Объединение – Связь между несколькими таблицами, в которых записи из совпадающих соответствующих таблиц объединяются и отображаются как одна запись. Несовпадающие записи могут быть как включены, так и исключены в зависимости от типа объединения.

Объединенные ячейки – Одна ячейка, созданная путем объединения двух или нескольких выделенных ячеек. Ссылкой на объединенную ячейку является адрес верхнего левого угла объединенного диапазона.

Ограничения – Ограничения на значения изменяемых ячеек, конечных ячеек или других ячеек, прямо или косвенно связанных друг с другом, задаваемые при постановке задачи.

Окно — Прямоугольная область на экране, в которой отображается прикладная программа или документ. Окна программы или документа в MS Windows можно перемещать, изменять в размере, открывать, закрывать.

Операнд – Объекты, над которыми выполняются действия с помощью операций. В Microsoft Excel операндами являются значения, ссылки на ячейки, имена, надписи и функции.

Оператор сравнения – Знак, используемый в условиях для сравнения двух значений. Имеются шесть стандартных операторов: = (равно), > (больше), < (меньше), >= (больше или равно), <= (меньше или равно), <> (не равно).

Ось – Линия, ограничивающая область построения диаграммы, используемая как система координат в выбранных единицах измерения. Ось Y обычно расположена вертикально, а вдоль нее строятся данные. Ось X обычно расположена горизонтально, а вдоль нее строятся категории. Ось диаграммы, представляющая категорию для каждого элемента данных. На ней отображаются произвольные текстовые значения, например Кв1, Кв2 и Кв3; на ней не могут отображаться масштабированные числовые значения.

Ось значений – Ось диаграммы, на которой отображаются масштабированные числовые значения.

Ось рядов – Ось диаграммы, представляющая измерение глубины на действительно объемной диаграмме. На ней отображаются имена рядов данных в виде произвольных текстовых значений; на ней не могут отображаться масштабированные числовые значения.

Относительная ссылка – Адрес ячейки в формуле, определяемый на основе расположения этой ячейки относительно ячейки, содержащей ссылку. При копировании ячейки относительная ссылка автоматически изменяется. Относительные ссылки задаются в форме A1.

Отчет сводной диаграммы – Диаграмма, выполняющая интерактивный анализ данных, как и отчет сводной таблицы. Пользователь имеет возможность изменять представление данных, просматривать данные с различными уровнями детализации, перестраивать макет диаграммы, перетаскивая поля и показывая или скрывая элементы полей.

Палитра — Набор цветов в графических пакетах. Существуют специальные палитры, например, WEB. Палитра, основанная на наибольшем показателе встречающихся или стоящих рядом цветов, называется оптимизированной.

Память — Под памятью компьютера обычно понимают Оперативную память - ОЗУ или RAM (Оперативное Запоминающее Устройство), в которой хранится текущая необходимая для работы компьютера информация.

Панель задач — Горизонтальная полоса в нижней части экрана Windows. Каждой загруженной программе присваивается определенная кнопка в полосе задач с именем этой программы. Кнопка используется для переключения в данную программу.

Панель управления — Панель управления (Control Panel) - встроенное приложение Windows, предназначенное для изменения конфигурации аппаратных средств и настройки системы при работе в Windows. Все параметры, которые вы можете изменить, находятся в окне Панели управления и представлены в виде пиктограмм.

Папка — Понятие, используемое в MS Windows и являющееся аналогом каталога. Подобно каталогу, папка может содержать другие папки (подкаталоги).

Параллельный порт — Порт LPT, как правило - LPT1. Разъем с тыльной стороны компьютера, используемый для подключения периферийного устройства, как правило, принтера.

Параметр – В Microsoft Excel можно добавить, изменить или удалить параметры, чтобы определить ячейки, которые можно редактировать в просматриваемых данных листа служб Excel Services. При сохранении книги изменения автоматически отражаются на сервере.

Переключатель — Элемент окна Windows-приложения, как правило - диалогового, служащий для включения или выключения режима, напротив которого находится переключатель.

Пиксель (pixel) — Минимальная единица для отображения на экране монитора. Применяется для указания размеров графики.

Пиктограмма (значок) — Условное графическое изображение программного продукта или свернутого группового окна программ. Пиктограмма конкретного приложения служит для его запуска.

- По умолчанию** — Определение, обозначающее, что при открытии документа или выполнении какой-либо команды будут автоматически применены кем-то (либо разработчиком, либо пользователем) установленные ранее параметры при отсутствии дополнительных указаний (действий) пользователя. Установки "по умолчанию" можно изменить в зависимости от конкретных потребностей.
- Подпись данных** – Подпись с дополнительными сведениями о маркере данных, представляющем одну точку данных или значение ячейки листа.
- Подпись линии тренда** – Текст, который может сопровождать линию тренда. В качестве этого текста может выступать как уравнение, используемое для линии тренда, так и значение R-квадрат, а также и то, и другое. Подпись линии тренда нельзя отформатировать или переместить, нельзя также изменить ее размер.
- Подробные данные** – В таблицах с итогами и структурах — строки или столбцы с исходными данными, по которым вычисляются итоги. Подробные данные обычно расположены сверху или слева от итогов и примыкают к ним.
- Подстановочный символ (звездочка)** — Символ для представления (замещения) одного или нескольких других символов в шаблонах имен файлов. Например, в имени файла (при поиске, замене и т. д.) используется подстановочный символ ? (вопросительный знак) для замещения одного любого символа, находящегося на той же позиции в именах других файлов. Подстановочный символ * (звездочка) используется для представления группы символов, расположенных на тех же позициях. Например, запись *.DOC представляет все файлы с расширением .DOC.
- Поле (сводный отчет)** – В сводном отчете или отчете сводной диаграммы категория данных, получаемых из поля исходных данных. Отчеты сводной таблицы содержат поля строк, столбцов, страниц и данных. Отчеты сводной диаграммы содержат поля рядов, категорий, страниц и данных.
- Поле значений** – Поле исходного списка, таблицы или базы данных, которое является источником данных для вычислений в отчете сводной таблицы или в отчете сводной диаграммы. Поле значений обычно содержит числа, такие как статистические данные или объемы продаж.
- Поле имени** – Поле слева от строки формул, идентифицирующее выделенную ячейку, элемент диаграммы или графический объект. Чтобы присвоить имя ячейке или диапазону ячеек, введите это имя в поле имени и нажмите клавишу ENTER. Чтобы перейти к именованной ячейке, щелкните ее имя в поле имени.
- Поле категории** – Поле, выводящееся в области категорий отчета сводной диаграммы. Элементы поля категории отображаются как подписи по оси категорий.
- Поле категорий сводной диаграммы** – Поле, назначенное оси категорий в отчете сводной диаграммы. Категории на диаграмме обычно отображаются вдоль оси X (горизонтальной оси).
- Поле ряда** – Поле, отображаемое в области рядов отчета сводной диаграммы. Элементы в поле рядов указаны в легенде и дают имена отдельным рядам данных.

- Поле рядов сводной диаграммы** – Поле, назначенное оси рядов в отчете сводной диаграммы. На диаграмме ряды обычно показываются в легенде.
- Поле с раскрывающимся списком** – Элемент управления в меню, диалоговом окне или на панели инструментов, выводящий список параметров при нажатии кнопки со стрелкой справа от него.
- Поле столбца** – Поле, которому в сводном отчете соответствуют столбцы. Элементы, связанные с полем столбца, отображаются в виде подписей столбцов.
- Порядок сортировки** – Способ упорядочивания данных в зависимости от их значений и типа. Данные могут быть отсортированы по алфавиту, по числовым значениям или по дате. Порядок сортировки может быть по возрастанию (от 1 до 9, от А до Я) или по убыванию (от 9 до 1, от Я до А).
- Последовательный порт** – Порт COM. Разъем с тыльной стороны компьютера, используемый для подключения периферийного устройства (принтера, мыши, модема и пр.).
- Представление** – Набор параметров отображения и печати, которые можно именовать и применять к книге. В одной книге можно создавать несколько представлений без необходимости сохранения каждого в отдельной копии книги. Представление листа, в котором отображаются области печати и разрывы страниц. Область печати отображается белым цветом, автоматические разрывы страницы пунктиром, а вставленные пользователем разрывы страницы непрерывными линиями.
- Промежуточные итоги сводной таблицы** – Строка или столбец, отображающие итог по элементам поля сводной таблицы после использования выбранной итоговой функции.
- Пункт** — Единица измерения размера шрифта (в том числе в компьютерных системах, например, в MS EXCEL). Размер шрифта называется кеглем. 1 пункт приблизительно равен 0,354 мм.
- Пункт** – Единица измерения, равная 1/72 дюйма (около 0,353 мм).
- Раздел** – Любая комбинация листа, представления и сценария, выбираемая при создании отчета. Отчет может содержать несколько разделов.
- Разрешение** — Показатель качества графического изображения. Измеряется в точках на дюйм - dpi (dots per inch). Количество dpi - показатель разрешения, с которым печатает принтер или вводит изображение сканер. С увеличением разрешения качество изображения возрастает.
- Разрыв страницы** – Разделитель страниц для печати. В Microsoft Excel разрывы строки вставляются автоматически в зависимости от размера бумаги, полей, масштаба и расположения разрывов строки, вставленных пользователем.
- Раскрывающийся список** — Элемент экрана, как правило, диалогового окна или панели инструментов. Открывается путем нажатия кнопки.

Режим прокрутки листа – Если включен режим прокрутки листа, клавиши перемещения курсора позволяют осуществлять панорамное перемещение по листу, а не переход к другой ячейке. Чтобы отключить или включить режим прокрутки листа, нажмите клавишу SCROLL LOCK.

Ряд данных – Набор связанных между собой элементов данных, соответствующий одному столбцу или одной строке таблицы данных. Каждому ряду данных на диаграмме соответствует отдельный цвет и/или способ обозначения. Диаграммы всех типов, кроме круговой, могут содержать несколько рядов данных.

Сводный отчет – Интерактивный перекрестный отчет Microsoft Excel, содержащий итоговые данные и выполняющий анализ таких данных, как записи базы данных из разных источников, в том числе внешних по отношению к Microsoft Excel.

Связанный сводный отчет – Отчет сводной таблицы, предоставляющий данные для отчета сводной диаграммы. Он создается автоматически при создании отчета сводной диаграммы. При изменении макета любого из этих отчетов также изменяется и другой макет.

Символы структуры – Символы, используемые для изменения вида структурированного листа. Подробные данные можно показать или скрыть, щелкнув знак «плюс», «минус» или цифры 1, 2, 3, 4, показывающие уровень структуры.

Системный файл — Файл, содержащий определенную информацию, необходимую для работы операционной системы.

Сканер – Устройство ввода. Служит для ввода в компьютер разнообразных изображений. С помощью программ распознавания текста, позволяет получать текст, пригодный для редактирования.

Список сводной таблицы – Список сводной таблицы. Веб-компонент Microsoft Office, позволяющий создавать структуру, похожую на сводный отчет Microsoft Excel. Пользователи могут просматривать список сводной таблицы в веб-обозревателе и изменять его так же, как и сводный отчет таблицы в Microsoft Excel.

Ссылка на ячейку – Набор координат, по которым расположена ячейка на листе. Например, ссылка на ячейку, находящуюся на пересечении столбца В и строки 3, выглядит как В3.

Стандартный шаблон книги – Шаблон Книга.xlt, создаваемый для переопределения используемого по умолчанию формата вновь создаваемой пустой книги Microsoft Excel. Именно этот шаблон будет использоваться Microsoft Excel для создания пустой книги при запуске или создании книги без указания шаблона.

Стандартный шаблон листа – Шаблон Лист.xlt, создаваемый для переопределения используемого по умолчанию формата вновь создаваемого пустого листа Microsoft Excel. Именно этот шаблон будет использоваться Microsoft Excel для создания пустого листа при добавлении листа в книгу.

- Стандартный шрифт** – Стандартный шрифт для текста на листе. Этот шрифт установлен по умолчанию для обычного стиля ячеек.
- Строка вставки** – Специальная строка в таблице Microsoft Excel, облегчающая ввод данных. Строка вставки обозначается звездочкой.
- Строка итогов** – Особая строка в списке, в которой можно выбирать статистическую функцию для обработки числовых данных.
- Строка формул** – Панель в верхней части окна Microsoft Excel, которая используется для ввода или изменения значений или формул в ячейках или на диаграммах. На ней отображается константа или формула, содержащаяся в активной ячейке.
- Структура** – Режим электронной таблицы, при которой строки или столбцы сгруппированы так, что можно видеть итоги. В структуре возможно подведение итогов как по всему листу, так и по его определенному диапазону.
- Таблица Microsoft Excel** – Чтобы организовать данные на своем листе, можно создать, отформатировать и расширить таблицу Microsoft Excel, раньше называвшуюся списком Microsoft Excel.
- Таблица данных на диаграмме** – Сетка, которую можно добавить на некоторые диаграммы, содержащая числовые данные, отображаемые на диаграмме. Такая таблица данных обычно примыкает к горизонтальной оси диаграммы и заменяет на ней подписи делений.
- Текстовое поле** – Прямоугольный объект на листе, в который можно ввести текст.
- Текущая область** – Прямоугольная область заполненных ячеек, включающая текущие выбранные ячейки или одну ячейку. Эта область простирается во все стороны до первой пустой строки или столбца.
- Условие** – Ограничение, заданное для отбора записей, включаемых в список результатов запроса. Например, следующее условие отбирает только записи, значение поля «Сумма по доходу» которого больше 30 000: Сумма по доходу > 30000
- Условный формат** – Формат (например, узор ячейки или цвет шрифта), который в Microsoft Excel автоматически применяется к ячейке, если выполняется указанное условие.
- Файл** – С технической точки зрения - последовательность байтов. С практической – документ, созданный в какой-либо из программ и сохраненный с определенным именем (см. Имя файла). Информация в компьютере представлена в виде файлов. Файл – это в простейшем случае какой-либо документ либо программа, существующие на диске и имеющие собственное имя.
- Файл рабочей области** – Файл, сохраняющий сведения об отображении открытых книг, чтобы позже можно было возобновить работу с теми же размерами и положениями окон, областями печати, масштабом и другими параметрами отображения. Файл рабочей области не содержит сами книги.

Фильтр – Средство отображения только строк списка, соответствующих заданным условиям. Для отображения строк, соответствующих одному из заданных значений, вычисленных значений или условий, служит функция Автофильтр.

Фильтр отчета – Поле, используемое для фильтрации подмножества данных в сводном отчете или отчете сводной диаграммы на одной странице для последующего макетирования и анализа. Можно выводить как итоги по всем элементам в фильтре отчета, так и элементы по отдельности, отфильтровывая данные для всех других элементов.

Форма данных – Диалоговое окно, поочередно отображающее полные записи. В форме данных можно добавлять, изменять, удалять записи и проводить их поиск.

Формула – Совокупность значений, ссылок на другие ячейки, именованных объектов, функций и операторов, позволяющая получить новое значение. Формула всегда начинается со знака равенства (=).

Библиографический список

1. Gluschenko V.M., Pronkin N.N., Simakov A.I. Analysis of the problems of ensuring information security of the megalopolis. *International Journal of Professional Science*. 2021. № 1. С. 43-49.
2. Glushchenko V.M., Pronkin N.N. Principles for assessing potential and prevented damage to the information security of a megalopolis. *International Journal of Professional Science*. 2021. № 8. С. 5-8.
3. Movchun V., Lushkov R., Pronkin N. Prediction of individual learning style in e-learning systems: opportunities and limitations in dental education. *Education and Information Technologies*. 2020.
4. Panfilova E., Lukyanova A., Pronkin N.N., Zatsarinnaya E. Cloud technology development alongside public life digitalization. *Talent Development and Excellence*. 2020. Т. 12. № S2. С. 3324-3334.
5. Pashanova O.V., Ermakov D.A., Philippova A.V., Tikhonova Yu.A., Pronkin N.N. Analysis methods for medications improving cerebral circulation. *Research Journal of Pharmacy and Technology*. 2021. Т. 14. № 1. С. 115-121.
6. Tyurina Yu., Troyanskaya M., Babaskina L., Choriyeв R., Pronkin N.N. E-learning for SMES. *International Journal of Emerging Technologies in Learning*. 2021. Т. 16. № 2. С. 108-119.
7. Глущенко В.М., Елизаров В.С., Новиков А.Н., Пронькин Н.Н. Информационные технологии в управленческой деятельности. Учебное пособие для государственных гражданских служащих г. Москвы, обучающихся по образовательной программе повышения квалификации. – Московский городской университет управления Правительства Москвы. Москва, 2010.
8. Глущенко В.М., Новиков А.Н., Пронькин Н.Н. Особенности формирования и содержания модели управления московским мегаполисом. *Информационные и телекоммуникационные технологии*. 2019. № 44. С. 32-37.
9. Грейбо С.В., Новосёлова Т.Е., Пронькин Н.Н., Семёнычева И.Ф. Дистанционные технологии обучения в сеченовском университете. их преимущества и недостатки. *International Journal of Professional Science*. 2020. № 4. С. 20-36.
10. Елизарова М.И., Уразова К.М., Ермашов С.Н., Пронькин Н.Н. Искусственный интеллект в медицине. *International Journal of Professional Science*. 2021. № 5. С. 81-85.
11. Иконникова И.А., Новосёлова Т.Е., Пронькин Н.Н., Семёнычева И.Ф. Информационная технология баз данных в медицине. Москва, 2020. Том Часть I.
12. Иконникова И.А., Новосёлова Т.Е., Пронькин Н.Н., Семёнычева И.Ф. Инженерная графика. Москва, 2020.

13. Калачанов В.Д., Ефимова Н.С., Новиков А.Н., Пронькин Н.Н. Внедрение систем диспетчирования производства на высокотехнологичных предприятиях (на примере предприятий авиастроения). *Инновации и инвестиции*. 2019. № 3. С. 269-273.
14. Калачанов В.Д., Новиков А.Н., Калачанов В.В., Пронькин Н.Н. Критерии оптимального управления финансированием производственной деятельности предприятий высокотехнологичных отраслей промышленности (на примере авиастроения). *Организатор производства*. 2016. № 1 (68). С. 61-68.
15. Калачанов В.Д., Новиков А.Н., Калачанов В.В., Пронькин Н.Н. Разработка комплексной системы критериев оптимизации финансирования производственной деятельности промышленных предприятий (на примере авиастроения). *Организатор производства*. 2016. № 3 (70). С. 50-61.
16. Мяснянкина О.П., Пронькин Н.Н. Достижения и перспективы искусственного интеллекта в медицине. *International Journal of Professional Science*. 2021. № 4. С. 27-32.
17. Новиков А.Н., Пронькин Н.Н. Внедрение инструментальных методов в управление экономикой предприятий ОПК. *Вестник Академии военных наук*. 2014. № 3 (48). С. 148-152.
18. Пронькин Н.Н. Инструментарий управления системой производства продукции ОПК. *Вестник Академии военных наук*. 2015. № 1 (50). С. 147-150.
19. Пронькин Н.Н. Новые подходы к построению системы дистанционного обучения в МГУУ Правительства Москвы. *Вестник МГУУ*. 2011. № 1. С. 135-147.
20. Пронькин Н.Н. Практика внедрения системы дистанционного обучения в МГУУ Правительства Москвы на основе E-learning 3000. *Информационные и телекоммуникационные технологии*. 2011. № 12. С. 72-78.
21. Пронькин Н.Н., Новиков А.Н. Программно-математические методы обоснования потребности в экономических ресурсах для выполнения государственного оборонного заказа. *Вестник Академии военных наук*. 2014. № 4 (49). С. 122-125.
22. Пронькин Н.Н., Новиков А.Н. Совершенствование информационных технологий в управлении столичным мегаполисом. *Вестник МГУУ*. 2012. № 1. С. 84-90.

ISBN 978-1-387-25950-2



9 781387 259502

Усл. печ. л. 2.9.

Объем издания 13.3 МВ

Оформление электронного издания:

НОО Профессиональная наука, mail@scipro.ru

Дата размещения: 10.10.2021 г.

URL: <http://scipro.ru/conf/database.pdf>