### МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Воронежский государственный технический университет»

Кафедра высшей математики и физико-математического моделирования

# ОСНОВНЫЕ ПРИЕМЫ РАБОТЫ С БАЗАМИ ДАННЫХ

#### МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к выполнению лабораторной работы по курсам «Информатика» и «Практикум по информационным технологиям» для студентов направлений 14.03.01 «Ядерная энергетика и теплофизика», 16.03.01 «Техническая физика», 21.03.01 «Нефтегазовое дело», 22.03.02 «Металлургия» и 28.03.01 «Нанотехнологии и микросистемная техника» очной и очно-заочной форм обучения

#### Составители:

С. А. Кострюков, В. В. Пешков, Г. Е. Шунин

Основные приемы работы с базами данных: методические указания к выполнению лабораторной работы по курсам «Информатика» и «Практикум по информационным технологиям» для студентов направлений 14.03.01 «Ядерная энергетика и теплофизика», 16.03.01 «Техническая физика», 21.03.01 «Нефтегазовое дело», 22.03.02 «Металлургия» и 28.03.01 «Нанотехнологии и микросистемная техника» очной и очно-заочной форм обучения / ФГБОУ ВО «Воронежский государственный технический университет»; сост.: С. А. Кострюков, В. В. Пешков, Г. Е. Шунин. – Воронеж: Изд-во ВГТУ, 2021. – 22 с.

В методических указаниях кратко описаны основные понятия баз данных и возможности систем управления базами данных (СУБД), рассмотрены приемы работы с ними. Приводится большое количество разобранных примеров и заданий для самостоятельной работы.

Издание предназначено для студентов очной и очнозаочной форм обучения.

Методические указания подготовлены в электронном виде и содержатся в файле MУ-Access.pdf.

Ил. 4. Табл. 2. Библиогр.: 3 назв.

УДК 004.65:681.3(07) ББК 32.97я7

Рецензент – И. М. Пашуева, канд. физ.-мат. наук, доц. кафедры высшей математики и физико-математического моделирования ВГТУ

Издается по решению редакционно-издательского совета Воронежского государственного технического университета

#### ВВЕДЕНИЕ

База данных (БД) — организованная в соответствии с определёнными правилами и поддерживаемая в памяти компьютера совокупность данных, характеризующая актуальное состояние некоторой предметной области и используемая для удовлетворения информационных потребностей пользователей. Для работы с БД служит специальная категория программных средств — системы управления базами данных (СУБД). Таким образом, СУБД — это совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных.

Изначально, когда понятие БД только формулировалось, в них действительно содержались только данные. Однако сегодня большинство СУБД позволяют хранить в своих структурах не только сами данные, но и методы (т.е. программный код соответствующих подпрограмм, с помощью которых происходит обработка данных и взаимодействие с пользователями или с другими программно-аппаратными комплексами). Следовательно, в настоящее время можно утверждать, что в существующих сейчас БД хранятся не только данные, но и информация (сравните с определением информации как продукта взаимодействия данных и адекватных им методов).

С любой БД пользователь взаимодействует через посредство соответствующей СУБД. С её помощью происходит создание структуры новой пустой БД, заполнение её содержимым, изменение и обновление содержимого, а также визуализация данных. Визуализация означает отбор данных в соответствии с заданным критерием, их упорядочение, оформление и последующая выдача на устройство вывода или передача по каналам связи.

В настоящее время существует большое количество СУБД нескольких основных производителей. Каждая СУБД основывается на какой-либо конкретной модели данных, и, следовательно, может по-своему работать с объектами БД и предоставлять пользователю различные функции и средства.

Наиболее популярными в настоящее время считаются СУБД, основанные на реляционной модели данных, поэтому здесь будут рассмотрены основные понятия, приёмы и методы работы с одной из них, а именно, с СУБД Microsoft Access, входящей в пакет Microsoft Office.

### Структура простейшей БД

Следует сразу подчеркнуть, что даже если в БД вообще нет никаких данных (пустая база), то это, тем не менее, полноценная БД. Несмотря на то, что данных в БД нет, информация в ней всё же имеется — это структура базы, определяющая формат вносимых данных и способ хранения их в базе. Бумажным аналогом такой пустой БД является знакомый всем ежедневник, где на каждый день приходится по странице. Если в него не внесено ни одной записи, он всё равно остается ежедневником, так как имеет структуру, которая отличает его от телефонной или записной книжки, рабочей тетради и т.п.

Основными объектами реляционной БД являются её таблицы, которые описывают отношения между элементами данных. Понятие «реляционный» происходит от англ. relation («отношение», «зависимость», «связь»). Простейшая реляционная БД имеет минимум одну таблицу, и, таким образом, её структура представляет собой структуру этой таблицы.

Известно, что двумерная таблица состоит из столбцов и строк. Их аналогами в структуре БД являются поля (атрибуты) и записи (кортежи). Если изменить набор полей базовой таблицы или их свойства, то изменится структура БД, и мы, соответственно, получим новую БД.

### Свойства полей базовых таблиц

Поля БД, помимо структуры базы, также задают групповые свойства данных, которые будут вводиться в соответствующие ячейки. Рассмотрим основные свойства полей таблиц БД на примере СУБД Microsoft Access.

Имя поля — определяет, как необходимо обращаться к данным этого поля при операциях с базой (имя поля по умолчанию будет служить заголовком столбца таблицы).

- Тип поля задаёт тип данных, которые будут находиться в этом поле.
- Размер поля задаёт максимальную длину в символах тех данных, которые будут содержаться в этом поле.
- Формат поля задаёт способ форматирования данных в ячейках поля.
- Маска ввода задаёт форму, в которой будут вводиться данные в поле (одно из средств автоматизации ввода).
- Подпись задаёт заголовок соответствующего столбца таблицы (если подпись отсутствует, то для заголовка столбца будет использовано свойство Имя поля).
- Значение по умолчанию значение, внедряемое в ячейки данного поля автоматически (одно из средство автоматизации ввода).
- Условие на значение ограничение, служащее для проверки корректности вводимых данных (используется обычно для числовых, денежных данных или даты/времени).
- Сообщение об ошибке текст, который будет выводиться при попытке ввода в ячейку поля некорректных данных (проверка будет выполняться автоматически, если было задано свойство Условие на значение).
- Обязательное поле свойство, которое определяет обязательность заполнения этого поля при вводе записей.
- Пустые строки свойство, которое разрешает ввод в ячейки поля пустых строк (в отличие от свойства Обязательное поле, относится не ко всем типам данных, а лишь к некоторым, например, к текстовым).
- Индексированное поле свойство, служащее для ускорения поиска и сортировки записей по значению, содержащемуся в данном поле. Также для индексированного поля можно исключить дублирование данных, т.е. сделать так, что значения в записях будут проверяться по этому полю на наличие повторов.

Набор свойств у разнотипных полей может различаться в зависимости от типа данных. Например, рассмотренный выше список свойств в основном относится к полям текстового типа. Поля других типов могут иметь или не иметь эти свойства, а

также могут добавлять к ним свои. Например, для данных, являющихся действительными числами, часто оказывается важным количество знаков после запятой. В то же время для полей, служащих для хранения рисунков, звукозаписей и других объектов, многие из перечисленных свойств не имеют смысла.

#### Типы данных

Базы данных Microsoft Access могут использовать поля следующих типов:

- Текстовый тип данных для хранения обычного текста небольшого размера (до 255 символов).
- Поле Мемо тип данных для хранения большого объема текста (максимум 65535 символов). Сам текст при этом не хранится в поле, он находится в другом месте БД, а поле лишь содержит ссылку (т.е. указатель) на него.
- Числовой тип данных для хранения целых или вещественных чисел, конкретный формат поля определяется отдельно в списке свойств поля.
- Дата/время тип данных для записи календарных дат и времени, конкретный формат поля определяется отдельно в списке свойств поля.
- Денежный тип данных для хранения денежных сумм. При операциях над ними существуют некоторые особенности, например, связанные с правилами округления, поэтому использование числовых типов нецелесообразно.
- Счетчик тип данных для порядковой нумерации строк (записей) с автоматическим увеличением номеров.
- Логический тип для хранения логических данных, которые могут принимать только два значения, например, Нет или Да.
- Поле объекта OLE тип данных, предназначенный для хранения различных объектов *OLE*, например, мультимедийных.
  Как и для полей МЕМО, реально в таблице хранятся не сами эти объекты, а только ссылки на них.
- Гиперссылка тип данных для адресов URL объектов Интернета. При клике на ссылке будет автоматически запущен браузер для воспроизведения объекта в его окне.

 Мастер подстановок – это не тип данных, а объект, настроив который, можно автоматизировать ввод данных в поле – вместо ручного ввода пользователь сможет выбирать их из раскрывающегося списка.

### Режимы работы с базой данных

Как правило, с БД работает две группы исполнителей. Первая — это *проектировщики*, задача которых — разработка структуры таблиц БД и согласование её с заказчиком. Помимо таблиц, они проектируют и другие объекты БД, служащие для автоматизации работы с базой, а также для ограничения функциональных возможностей работы с базой (если это необходимо по причинам безопасности). Подчеркнем, что проектировщики не заполняют базу реальными данными, которые, как правило, конфиденциальны и не предоставляются посторонним лицам. Вместо этого они используют произвольные модельные данные с целью отладки объектов БД.

Вторая группа исполнителей, работающих с БД — это *пользователи*. Получив готовую пустую базу от проектировщиков, они наполняют её реальными данными и занимаются обслуживанием БД. Как правило, пользователи не имеют доступа к управлению структурой БД, а только к данным, и то лишь к тем, обработка которых входит в их служебные обязанности.

В соответствии с этим, СУБД может работать в двух режимах: *проектировочном* и *пользовательском*. Первый режим служит для создания или модификации структуры БД и создания её объектов. Во втором режиме эти объекты используются для заполнения базы или извлечения данных из неё.

### Объекты базы данных

СУБД Microsoft Access позволяет создавать и использовать объекты следующих типов.

– **Таблицы.** Это основные объекты реляционной БД. В таблицах хранятся все данные, имеющиеся в базе, кроме того, таблицы хранят и структуру базы (поля, их типы и свойства).

- Запросы. Эти объекты предназначены для выборки данных из таблиц и представления их в удобном для пользователя виде. С их помощью производится отбор, фильтрация и сортировка данных, также с их помощью можно осуществлять преобразование данных по заданному алгоритму, создание новых таблиц, автоматическое наполнение таблиц данными, импортированными из других источников, простейшие вычисления в таблицах и др.

С точки зрения безопасности, чем более ограничен доступ конечных пользователей к базовым таблицам, тем лучше. При этом уменьшается риск повреждения данных в таблицах из-за их неумелых действий. Кроме того, если разным пользователям предоставить разные запросы, то можно эффективно разграничить их доступ к данным в соответствии с персональными обязанностями. При правильной организации работы с БД для совершения преступных действий требуется сговор нескольких сотрудников, что значительно повышает шансы своевременного раскрытия и предотвращения ущерба от этих действий.

Запросы отбирают данные из базовых таблиц и создают из них временную результирующую таблицу. Чтобы подчеркнуть её временность, эту таблицу ещё иногда называют моментальным снимком. Работая с основными таблицами БД, мы при этом имеем дело с жестким диском, т.е. с довольно медленным устройством. Когда же запрос сформирует результирующую таблицу, существующую лишь в оперативной памяти, то работа с ней будет происходить гораздо быстрее и эффективнее. В случае обработки больших объемов данных разница становится достаточно заметной.

Один из основных недостатков упорядоченных табличных структур – сложность обновления, так как при добавлении новых записей нарушается упорядоченность, и приходится снова упорядочивать всю таблицу. В СУБД эта проблема решается также с помощью запросов.

Ключевым принципом здесь является то, что базовые таблицы всегда хранятся в неупорядоченном виде. Записи в

них всегда вносятся только в естественном порядке (по мере их поступления). Если же пользователь хочет отсортировать или отфильтровать данные по тому или иному принципу, то он должен применить соответствующий запрос. Если такого запроса нет в наличии, он должен обратиться к проектировщику с просьбой предоставить этот запрос.

— Формы. В противоположность запросам как средствам для отбора и анализа данных, формы представляют собой средства ввода данных. Их основная цель — чтобы пользователь располагал средствами для заполнения лишь тех полей, которые ему положено заполнять. Кроме этого, форма может содержать дополнительные элементы управления для автоматизации ввода — переключатели, раскрывающиеся списки, флажки, счетчики, и т.д. Особенно удобны формы в тех случаях, когда ввод данных происходит с заполненных бланков. Для этого внешний вид формы делают таким, чтобы она соответствовала виду бланка — при этом существенно упрощается работа наборщика, уменьшается количество ошибок и повышается производительность.

Формы применяются не только для ввода данных, но и для их отображения. При этом могут использоваться дополнительные средства оформления, в то время как запросы выводят данные как результирующую таблицу, которая таких средств практически не имеет.

- Отчеты. По структуре и свойствам отчеты во многом сходны с формами, но служат лишь для вывода данных, причем не на экран, а на печатающее устройство (т.е. на бумагу). В соответствии с этим отчеты содержат специальные средства для группирования выводимых данных и для вывода элементов оформления, характерных для печатных документов (номера страниц, сведения о времени создания отчета, верхний и нижний колонтитулы, и т.д.).
- **Страницы.** Эти объекты БД реализованы в последних версиях MS Access. Более верно их называть *страницами доступа к данным*. Физически это особый объект, созданный на

языке HTML, размещённый на web-странице и предоставляемый пользователю вместе с ней. Этот объект сам по себе не является БД, но включает компоненты, которые устанавливают связь этой web-страницы с БД, хранящейся на сервере. Используя эти компоненты, клиент сайта может просматривать данные из базы в полях страницы. Можно сказать, что страницы доступа к данным обеспечивают интерфейс между клиентом, сервером и БД, хранящейся на сервере. Эта БД может быть не только базой данных MS Access, страницы доступа, созданные с помощью MS Access, дают возможность работать также с БД Microsoft SQL Server.

– Макросы и модули. Эти типы объектов предназначены для автоматизации повторяющихся действий при работе с СУБД, а также для создания новых функций с помощью программирования. Макросы в Microsoft Access – это последовательность внутренних команд СУБД, они являются одним из средств автоматизации работы с БД. Модули же создаются с помощью внешнего языка программирования, в данном случае – Visual Basic for Applications (VBA). С их помощью разработчик БД может добавлять в неё нестандартные функциональные возможности, удовлетворять специфическим требованиям клиентов, увеличивать быстродействие системы управления, а также надёжность её защиты.

#### Лабораторная работа

# Упражнение 1

#### Создание базовых таблиц в MS Access

Рассмотрим ситуацию, когда руководитель малого предприятия, выполняющего сборку ПК из готовых компонентов, заказал разработку БД, основанной на двух базовых таблицах. Одна таблица должна содержать данные, которые могут отображаться для клиентов при отборе компонентов ПК, — в ней указаны розничные цены на компоненты. Вторая таблица должна служить для анализа результатов деятельности предприятия — в ней будут содержаться оптовые цены на компоненты и краткая информация о поставщиках (клиенты не имеют доступа к данным этой таблицы).

- 1. Запустите СУБД Microsoft Access. Кликните на кнопке Новая база данных.
- **2**. В правом нижнем углу окна MS Access выберите папку своей группы (либо другую по указанию преподавателя) и дайте файлу имя: Комплектующие. Кликните на кнопке Создать.
- 3. Открывшуюся в режиме таблицы Таблицу1 закройте с помощью клика правой кнопкой мыши на ярлычке таблицы (команда Закрыть).
- **4**. Перейдите в меню Создание и кликните на Конструктор таблиц откроется бланк создания структуры таблицы.
  - 5. Для первой таблицы задайте следующие поля (табл. 1).

Таблица 1

Имя поля	Тип данных		
Компонент	Текстовый		
Модель	Текстовый		
Основной параметр	Числовой		
Цена	Числовой		

Обратите внимание на то, что в этом случае поле Цена задано числовым типом, а не денежным. Данные денежного типа имеют размерность, выраженную в рублях (если Вы работаете с версией Microsoft Access, локализованной в России). Но стоимость компонентов вычислительной техники выражать в

этой единице измерения не принято. Для сравнимости цен разных поставщиков обычно используют «условные единицы». В таких случаях более удобно использовать поле числового типа, чтобы не перенастраивать всю СУБД.

- **6**. Кликните на поле Цена. В нижней части бланка задайте тип Одинарное с плавающей точкой и свойство Число десятичных знаков -2.
- 7. Для связи с будущей таблицей поставщиков нужно задать ключевое поле. Так как здесь ни одно поле по отдельности не может быть признано уникальным, будем использовать комбинацию полей Компонент и Модель. Выделите оба поля в верхней части бланка, для этого кликните в ячейке пустого столбца слева от имени поля Компонент, а затем при нажатой клавише Shift в ячейке слева от имени поля Модель. С помощью меню Конструктор и кнопки Ключевое поле либо контекстного меню (клик правой кнопкой мыши) сделайте эти поля ключевыми (в ранее пустом столбце появится изображение ключа).
- 8. Закройте окно Конструктора. При закрытии окна дайте таблице имя Комплектующие.
- **9**. Повторив действия пунктов 4–8, создайте таблицу Поставщики, в которую входят следующие поля (табл. 2).

Таблица 2

Имя поля	Тип данных
Компонент	Текстовый
Модель	Текстовый
Цена оптовая	Числовой
Поставщик	Текстовый
Телефон	Текстовый
Адрес	Текстовый
Примечание	Поле МЕМО

Для оптовой цены также задайте тип Одинарное с плавающей точкой и свойство Число десятичных знаков -2.

Обратите внимание на то, что поле номера телефона является текстовым, несмотря на то, что обычно номера телефонов записывают цифрами. Это связано с тем, что они не имеют числового содержания, причем их почти всегда записывают через черточку (дефис). Это типичное текстовое поле.

**Ключевое поле** для второй таблицы можно не задавать – для текущей задачи оно не требуется.

**10**. Закончив работу, закройте созданные базовые таблицы. Теперь требуется связать их между собой и наполнить конкретными данными.

### Упражнение 2

#### Создание межтабличных связей и ввод данных

- 1. В окне Microsoft Access найдите панель Все таблицы. Убедитесь, что на ней присутствуют значки ранее созданных таблиц Комплектующие и Поставщики.
- 2. В меню Работа с базами данных найдите и кликните на кнопке Схема данных. Откроется окно Схема данных. Одновременно с открытием этого окна будет открыто диалоговое окно Добавление таблицы, на вкладке Таблицы которого можно выбрать таблицы, между которыми создаются связи.
- 3. Кликом на кнопке Добавить выберите таблицы Комплектующие и Поставщики в окно Схема данных добавятся списки полей этих таблиц.
- **4**. При нажатой клавише **Shift** мышью выделите в таблице Комплектующие ключевые поля Компонент и Модель.
- 5. Перетащите выделенные поля на список полей таблицы Поставщики. При отпускании кнопки мыши автоматически откроется диалоговое окно Изменение связей.
- 6. На правой панели окна Изменение связей выберите поля Компонент и Модель таблицы Поставщики, включаемые в связь. Не устанавливайте флажок Обеспечение целостности данных в данном упражнении это не требуется, но может препятствовать постановке учебных опытов с таблицами.
- 7. Закройте диалоговое окно Изменение связей и в окне Схема данных рассмотрите образовавшиеся связи. Убедитесь, что линию связи можно выделить кликом левой кнопки мыши, а с помощью правой кнопки мыши открывается контекстное меню, позволяющее разорвать связь или отредактировать её. Закройте окно Схема данных.
- **8**. На панели Все таблицы откройте таблицу Комплектующие и наполните ее экспериментальными данными, например:

	1	Комплектующие				
		Компонент	Модель	Основной параметр	Цена	Доб
	+	Жесткий диск	WD Blue WD5000AZRZ	500	2599	
	+	Жесткий диск	Seagate BarraCuda ST1000DM010	1000	2999	
	+	Жесткий диск	Toshiba P300 HDWD220UZSVA	2000	3899	
	+	Жесткий диск	WD Blue WD40EZAZ	4000	6799	
	+	Процессор	AMD A8-9600 OEM 4x3100	3100	5099	
	+	Процессор	Intel Core i3-10100F BOX 4x3600	3600	7099	
	+	Процессор	Intel Core i3-10105F BOX 4x3700	3700	7599	
	+	Процессор	Intel Core i3-10320 OEM 4x3800	3800	13599	
	+	Процессор	AMD Ryzen 5 5600G OEM 6x3900	3900	25499	
344						

Рис. 1

**9**. Через связь между таблицами (знак «+» слева от поля Компонент) заполните те поля второй таблицы, которых нет в первой (рис. 2). Таблица Поставщики показана на рис. 3.

	1	Ком	плектующие							
	Компонент			Модель		Основной параметр		Цена	L	
	早	Жесткий диск Цена опт		WD Blue WD5000AZRZ		500		259	99	
	L			Поставщик	Телефон	A,	дрес	Примечан		
			27,65	ООО "Арго"	2-11-22-44	ул. Мира	a, 7			
		*								
	早	🗏 Жесткий диск		Seagate BarraCuda ST1000DM010			1000	299	99	
	L		Цена опт	Поставщик	Телефон	A,	дрес	Примечан		
			31,25	ООО "Арго"	2-11-22-44	ул. Мира	a, 7			
		*								
	+	Же	есткий диск	Toshiba P300	HDWD220U	ZSVA		2000	389	99
	+	Жесткий диск Процессор		WD Blue WD4	0EZAZ			4000	679	99
	+			AMD A8-9600 OEM 4x3100		3100		509	99	
	+	Процессор		Intel Core i3-10100F BOX 4x3600		3600		709	99	
	+	Процессор		Intel Core i3-10105F BOX 4x3700		3700		759	99	
	+	Процессор		Intel Core i3-10320 OEM 4x3800		3800		1359	99	
	+	Процессор AMD Ryzen 5 5600G OEM 6x3		6x3900		2549	99			
ale				·						

Рис. 2

Поставщики						
Компонент	Модель	Цена опт	Поставщик	Телефон	Адрес	Примечан
Жесткий диск	WD Blue WD5000AZRZ	27,65	ООО "Арго"	2-11-22-44	ул. Мира, 7	
Жесткий диск	Seagate BarraCuda ST1000DM010	31,25	ООО "Арго"	2-11-22-44	ул. Мира, 7	
Жесткий диск	Toshiba P300 HDWD220UZSVA	39,5	000 "ЧИП"	2-22-33-44	ул. Репина, 4	Предоплата
Жесткий диск	WD Blue WD40EZAZ	73	000 "ЧИП"	2-22-33-44	ул. Репина, 4	Предоплата
Процессор	AMD A8-9600 OEM 4x3100	56,5	ООО "Лидер"	2-33-44-55	ул. Ленина, 7	Предоплата
Процессор	AMD Ryzen 5 5600G OEM 6x3900	291,65	ООО "Лидер"	2-33-44-55	ул. Ленина, 7	Предоплата
Процессор	Intel Core i3-10320 OEM 4x3800	152,75	000 "РиК"	2-55-66-77	пр. Труда, 20	
Процессор	Intel Core i3-10105F BOX 4x3700	81,5	000 "РиК"	2-55-66-77	пр. Труда, 20	
Процессор	Intel Core i3-10100F BOX 4x3600	74,25	000 "РиК"	2-55-66-77	пр. Труда, 20	
Ψ.						

Рис. 3

10. Закройте все открытые таблицы базы данных.

# Упражнение 3 Создание запроса на выборку

В этом упражнении требуется создать запрос на выборку данных о жестких дисках, имеющих емкость не менее 1500 Гб при цене менее 40 у. е. Результирующая таблица также должна содержать фирму-поставщика, её адрес и номер телефона.

- 1. Если это упражнение выполняется не сразу за предыдущим, запустите СУБД Microsoft Access и откройте ранее созданную базу Комплектующие.
- 2. Перейдите в меню Создание. Кликните на кнопке Конструктор запросов откроется бланк запроса по образцу. Одновременно откроется диалоговое окно Добавление таблицы.
- 3. В окне Добавление таблицы выберите поочередно таблицы Комплектующие и Поставщики и добавьте их в верхнюю часть бланка запроса с помощью кнопки Добавить. Закройте окно Добавление таблицы.
- 4. В таблице Поставщики выберите поля, включаемые в результирующую таблицу: Компонент, Модель, Цена оптовая, Поставщик, Телефон. Для этого нужно выполнить двойные клики на именах полей.
- 5. Чтобы из таблицы выбирались не все изделия, а только жесткие диски, задайте условие отбора для поля Компонент, для чего в соответствующую строку введите: Жесткий диск.
- **6**. Далее задайте условие отбора для поля Цена оптовая. В соответствующую строку введите: <40. Из таблицы будут выбираться только изделия, имеющие цену менее 40 у.е.
- 7. Задайте условие отбора по основному потребительскому параметру ёмкости жесткого диска. Двойным кликом на поле Основной параметр в списке полей таблицы Комплектующие введите это поле в бланк запроса по образцу. Далее в строке Условие отбора столбца Основной параметр введите условие: >1500 (т.е. емкость диска более 1500 Гбайт).
- **8**. Закройте бланк запроса. При закрытии запроса введите его имя Выбор HDD.
- 9. На панели Все таблицы найдите и откройте только что созданный запрос и рассмотрите результирующую таблицу. Ее

содержание зависит от того, что было введено в таблицы Комплектующие и Поставщики при их заполнении. Если ни одно изделие не соответствует условиям отбора и получившаяся результирующая таблица пуста, откройте базовые таблицы и наполните их модельными данными, позволяющими проверить работу запроса.

10. Закройте все открытые объекты базы данных.

## Упражнение 4 Создание запросов «с параметром»

В предыдущем упражнении было показано, как действуют условия отбора. Существенным недостатком такого подхода является невозможность для пользователя изменять условия отбора. Если, например, разработчик предусмотрел запрос для отбора изделий ценой менее 40 у.е., то пользователь базы уже не в состоянии отобрать изделия, цена которых менее, чем 30 у.е., поскольку у него нет соответствующего запроса.

Специальный тип запросов, который называется запросами с параметром, позволяет пользователю самому ввести критерий отбора данных на этапе запуска запроса. Это существенно повышает эффективность и гибкость работы с базой.

Создадим простой запрос для отбора процессоров, максимальную цену которых пользователь может задать сам при запуске запроса.

- **1**. Запустите, если надо, программу Microsoft Access и откройте ранее созданную базу Комплектующие.
- **2**. Перейдите в меню Создание и кликните на кнопке Конструктор запросов.
- 3. Согласно предыдущему упражнению, создайте запрос на выборку, основанный на таблице Поставщики, в который войдут поля:
  - Компонент;
  - Модель;
  - Цена оптовая;
  - Поставщик;
  - Телефон;

- **4**. В строке Условие отбора поля Компонент введите: Процессор.
- 5. Строку Условие отбора для поля Цена оптовая надо заполнить таким образом, чтобы при запуске запроса пользователь получал предложение ввести нужное значение. Если бы мы хотели отобрать процессоры, цена которых меньше 80

у.е., мы бы написали <80. Но если мы хотим дать пользователю возможность выбрать значение, мы должны написать: <[Введите максимальную цену:]. Текст, обращенный к пользователю, заключается в квадратные скобки.

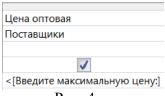


Рис. 4

- **6**. Закройте запрос. При закрытии сохраните его под именем Процессоры.
- 7. На панели Все таблицы найдите и откройте запрос Процессоры на экране появится диалоговое окно Введите значение параметра.
- **8**. Введите какое-либо число и кликните на кнопке ОК. В зависимости от того, что реально содержится в таблице Поставщики, по результатам запроса будет сформирована результирующая таблица.
- 9. Закройте все объекты базы данных. Закройте программу Microsoft Access.

## Упражнение 5 Создание итогового запроса

Если полностью заполнить данными таблицу Комплектующие, введя параметры всех компонентов, входящих в сборочную спецификацию персонального компьютера, то можно узнать, во что обходится себестоимость комплектующих узлов. Запросы, выполняющие вычисления по всем записям для какого-либо числового поля, называются *итоговыми запросами*. В итоговом запросе может рассчитываться сумма значений или величина среднего значения по всем ячейкам поля, может выбираться максимальное или минимальное значение данных в

поле, может также исполняться иная *итоговая функция*. Итоговые запросы, как и запросы на выборку, готовятся с помощью бланка запроса по образцу.

Предположим, что малое предприятие собирает компьютеры трех классов: «Элитный», «Деловой» и «Экономичный». Несмотря на то, что архитектура у всех компьютеров близка, их компоненты заметно отличаются по цене и техническим параметрам. Соответственно, имеются значительные отличия в цене этих трех моделей, что важно для захвата разных секторов рынка. Наша задача — подготовить итоговый отчет, с помощью которого можно определять цену каждой модели компьютеров и динамично ее отслеживать при изменении входящих компонентов или их поставщиков.

- 1. Запустите программу Microsoft Access и откройте ранее созданную базу Комплектующие.
- 2. Откройте таблицу Комплектующие в режиме конструктора нам это необходимо для создания дополнительного поля Класс, в котором будут храниться данные о том, для какого класса изделий предназначены компоненты.
- **3**. В начало структуры таблицы вставьте новое поле. Для этого выделите поле Компонент и нажмите клавишу **Insert**.
- Введите имя нового поля Класс и определите его тип Текстовый.
- **5**. Закройте окно Конструктора. При закрытии подтвердите необходимость изменить структуру таблицы.
- **6**. Откройте таблицу Комплектующие и наполните ее содержанием, введя для каждого класса данные по следующим изделиям:
  - Материнская плата;
  - Процессор;
  - O3У:
  - Жесткий диск;
  - Корпус;
  - DVD-ROM;
  - Видеокарта.

Цены на эти изделия для каждого класса проставьте произвольно. Прочие поля таблицы можно не заполнять – в формировании итогового запроса они участвовать не будут.

- 7. Закройте таблицу Комплектующие.
- 8. В меню Создание кликните на кнопке Конструктор запросов. В диалоговом окне Добавление таблицы выберите таблицу Комплектующие, на основе которой будет создаваться итоговый запрос. Закройте окно Добавление таблицы.
- **9**. В бланк запроса введите следующие поля таблицы Комплектующие: Класс, Компонент, Цена.
- **10**. Для поля **Класс** включите сортировку по возрастанию. Включите также сортировку по полю **Цена** по убыванию.
- 11. На панели инструментов MS Access щелкните на кнопке Итоги. Это необходимо для создания в нижней части бланка строки Групповые операции. Именно на ее базе и создаются итоговые вычисления. Все поля, отобранные для запроса, получают в этой строке значение Группировка.
- 12. Для поля, по которому производится группировка записей (в нашем случае Класс), оставьте в строке Групповые операции значение Группировка. Для остальных полей щелкните в этой строке появится кнопка раскрывающегося списка, из которого можно выбрать итоговую функцию для расчета значений в данном поле.
- 13. Для поля Цена выберите итоговую функцию Sum для определения стоимости изделия как суммы стоимостей комплектующих.
- 14. Для поля Компонент выберите итоговую функцию Count, определяющую общее количество записей, вошедших в группу. В нашем случае это количество узлов, из которых собран компьютер.
- **15**. Закройте бланк запроса по образцу и дайте ему имя: Расчет стоимости. Запустите запрос и убедитесь, что он правильно работает.
- **16**. Закройте все объекты БД и завершите работу с программой Microsoft Access.

#### Задание для самостоятельной работы

С помощью СУБД MS Access создать базу данных для хранения сведений о студентах факультета, состоящую из двух таблиц. Одна содержит сведения о студентах (фамилия, инициалы, группа, номер зачетки, дата рождения, адрес, учебное заведение, которое окончил студент до поступления в вуз, форма обучения в вузе — платное или бюджетное). Другая — сведения о сдаче сессии (номер зачетки, предметы, по которым были сданы экзамены — 3-4 предмета). При выполнении задания ввести сведения о 15-20 студентах, принадлежащих к двум-трем группам.

Далее с помощью созданной базы данных вывести на экран следующие сведения:

- 1) список студентов какой-либо группы (например, ЯЭ-171) с указанием номера зачетки и сведений о форме обучения;
- 2) список студентов, которые по итогам сессии будут получать стипендию, с указанием группы (учащиеся на платной основе стипендию не получают);
  - 3) итоги экзамена по математике в группе ЯЭ-171;
- 4) сведения о сдаче сессии указанным студентом (фамилия вводится при запуске запроса);
- 5) список студентов группы ЯЭ-171, имеющих хотя бы одну неудовлетворительную оценку;
- 6) список студентов, которые по итогам сессии будут получать повышенную стипендию (бюджетная форма обучения, без троек, хотя бы одна пятерка).

### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- 1. Информатика. Базовый курс: Учеб. пособие для втузов / Под ред. С.В. Симоновича. 3-е изд. СПб.: Питер, 2016. 637 с.
- 2. Лабораторный практикум по информатике: Учеб. пособие / Под ред. В.А. Острейковского. М.: Высш. шк., 2003. 376 с.
- 3. Днепров А. Microsoft Access 2007. СПб.: Питер, 2008. 400 с.

## ОГЛАВЛЕНИЕ

Введение	3
Структура простейшей БД	4
Свойства полей базовых таблиц	4
Типы данных	6
Режимы работы с базой данных	7
Объекты базы данных	7
Лабораторная работа	11
Задание для самостоятельной работы	20
Библиографический список	20

# ОСНОВНЫЕ ПРИЕМЫ РАБОТЫ С БАЗАМИ ДАННЫХ

#### МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к выполнению лабораторной работы по курсам «Информатика» и «Практикум по информационным технологиям» для студентов направлений 14.03.01 «Ядерная энергетика и теплофизика», 16.03.01 «Техническая физика», 21.03.01 «Нефтегазовое дело», 22.03.02 «Металлургия» и 28.03.01 «Нанотехнологии и микросистемная техника» очной и очно-заочной форм обучения

Составители:

Кострюков Сергей Александрович, Пешков Вадим Вячеславович, Шунин Геннадий Евгеньевич

Отпечатано в авторской редакции

Подписано к изданию 30.11.2021. Уч.-изд. л. 1,4

ФГБОУ ВО «Воронежский государственный технический университет» 394026 Воронеж, Московский проспект, 14