

Приемы работы в СУБД Microsoft Access

1. Базы данных

1.1. Базы данных и системы управления базами данных

1.2. Определения

1.3. Структура базы данных

1.4. Свойства полей баз данных

1.5. Типы данных

2. Объекты базы данных

3. Проектирование базы данных

3.1. Этапы проектирования базы данных

3.1.1. Разработка технического задания

3.1.2. Разработка схемы данных

4. Работа с СУБД Microsoft Access

4.1. Создание таблиц

4.2. Создание межтабличных связей

4.3. Создание запросов

4.4. Создание форм

4.5. Создание страниц

4.6. Создание отчетов



1. Базы данных

1.1. Базы данных и системы управления базами данных

Современный мир информационных технологий трудно представить без использования баз данных. Практически все системы в той или иной степени связаны с функциями долговременного хранения и обработки информации. Фактически информация становится фактором, определяющим эффективность любой сферы деятельности. Увеличились информационные потоки, повысились требования к скорости обработки данных и теперь уже большинство операций не может быть выполнено вручную, они требуют применения современных компьютерных технологий. Любые административные решения требуют четкой и точной оценки текущей ситуации и возможных перспектив ее изменения. Следовательно, обойтись сегодня без информационной модели производства, хранимой в базе данных, невозможно.

1.2. Определения

Банк данных (БНД) – это система специальным образом организованных данных – баз данных, программных, технических, языковых и организационно – методических средств, предназначенных для обеспечения централизованного накопления и коллективного многоцелевого использования данных.

База данных (БД) – именованная совокупность данных, отражающая состояние объектов и их отношений в рассматриваемой предметной области.

Система управления базами данных (СУБД) – совокупность языковых и программных средств, предназначенных для создания, ведения и совместного использования БД многими пользователями.

Программы, с помощью которых пользователи работают с базой данных, называются приложениями. В общем случае с одной БД могут работать множество различных приложений.

1.3. Структура базы данных

Если в базе нет никаких данных, то это всё равно полноценная база данных. Этот факт имеет методическое значение. Хотя данных в базе и нет, но информация в ней всё таки есть - это структура базы. Она определяет методы занесения данных и хранения их в базе. Простейший «некомпьютерный» вариант базы данных - деловой ежедневник, в котором каждому календарному дню выделено по странице. Даже если в нём не записано ни строки, он не перестаёт быть ежедневником, поскольку имеет структуру, чётко отличающую его от записных книжек. Базы данных могут содержать различные объекты. Основными объектами любой базы данных являются её таблицы. Простейшая база данных имеет хотя бы одну таблицу. Соответственно, структура простейшей базы данных равна структуре её таблицы. Структуру двумерной таблицы образуют столбцы и строки. Их аналогами в структуре

простейшей базы данных являются поля и записи. Если записей в таблице пока нет, значит, её структура образована только набором полей. Изменив состав полей базовой таблицы, мы изменяем структуру базы данных и, соответственно получаем новую базу данных.

1.4. Свойства полей баз данных

Поля базы данных не просто определяют структуру базы, они ещё определяют групповые свойства данных, записываемых в ячейки, принадлежащие каждому из полей. Ниже перечислены основные свойства полей таблиц баз данных на примере СУБД Microsoft Access.

Имя поля	Тип данных	Описание
Воздушное судно	Текстовый	
Год выпуска	Дата/время	

Свойства поля	
Общие	Подстановка
Размер поля	50
Формат поля	
Маска ввода	
Подпись	
Значение по умолчанию	
Условие на значение	
Сообщение об ошибке	
Обязательное поле	Нет
Пустые строки	Да
Индексированное поле	Да (Совпадения не допускаются)
Сжатие Юникод	Да
Режим IME	Нет контроля
Режим предложений IME	Нет

- Имя поля - определяет, как следует обращаться с данными этого поля при автоматических операциях с базой.
- Тип данных - определяет тип данных, которые могут содержаться в данном поле.
- Размер поля - определяет предельную длину данных, которые могут размещаться в данном поле.
- Формат поля - определяет способ форматирования в данных ячейках, принадлежащих полю.
- Маска ввода - определяет форму, в которой вводятся данные в поле.
- Подпись - определяет заголовок столбца таблицы для данного поля.
- Значение по умолчанию - то значение, которое вводится в ячейки поля автоматически.
- Условие на значение - ограничение, используемое для проверки правильности ввода данных.
- Сообщение об ошибке - текстовое сообщение, которое выдаётся автоматически при попытке ввода в поле ошибочных данных.
- Обязательное поле - свойство, определяющее обязательность заполнения данного поля при наполнении базы.
- Пустые строки - свойство, разрешающее ввод пустых строковых данных.
- Индексированное поле - если поле обладает этим свойством, все операции, связанные с поиском или сортировкой записей по значению, хранящемуся в данном поле, существенно ускоряются. Кроме того, для индексированных полей можно сделать так, что значения в записях будут проверяться по этому полю на наличие повторов, что позволяет автоматически исключить дублирование данных.

В связи с тем, что в разных полях БД могут содержаться данные разного типа, то и свойства у полей могут различаться в зависимости от типа данных. Список вышеуказанных полей относится в основном к полям текстового типа. Поля других типов могут иметь или не иметь эти свойства, но могут добавлять к ним и свои. Например, для данных представляющие действительные числа, важным свойством является количество знаков после десятичной запятой. С другой стороны, для полей, используемых для хранения рисунков, звукозаписей, видео клипов и других объектов OLE, большинство вышеуказанных свойств не имеет смысла.

1.5. Типы данных

С основными типами данных мы уже знакомы. Так, например, при изучении электронных таблиц мы видели, что они работают с тремя типами данных: текстами, числами и формулами. Таблицы без данных, как правило, допускают работу с гораздо большим количеством разных типов данных. Так, например, базы данных Microsoft Access работают со следующими типами данных (рис.1)

№п/п	Модель	Размер	Шаг маски	Цена	Наличие	Ожидается	Примечание
1	LG 575N	15"	0,28	3 980р	+		Гарантия фирм
2	NEC FE 771SB	17"	0,25	5 977р	+		Гарантия фирм
3	LGFlatronF700B	17"	0,24	5 504р	+		Гарантия фирм
4	Philips 109P	19"	0,25	13 870р		10.06.2004	
(Счетчик)					+		

Рис.1 Таблица с полями некоторых типов

1. **Текстовый** - тип данных, используемый для хранения обычного неформатированного текста ограниченного размера (до 255 символов)
2. **Поле Мемо** - специальный тип данных для хранения большого объема текста (до 65 535 символов). Физически текст не хранится в поле. Он хранится в другом месте базы данных, а в поле хранится указатель на него, но для пользователя такое разделение заметно не всегда.

Имя поля	Тип данных
Воздушное судно	Текстовый
Год выпуска	Текстовый
Дата капремонта	Поле МЕМО
Ремонтный завод	Числовой
Адрес завода	Дата/время
Телефон	Денежный
Интернет адрес	Счетчик
	Логический
	Поле объекта OLE
	Гиперссылка
	Мастер подстано

3. **Числовой** - тип данных для хранения действительных чисел
4. **Дата/время** - тип данных для хранения календарных дат и текущего времени.
5. **Денежный** - тип данных для хранения денежных сумм.
6. **Счетчик** - специальный тип данных для уникальных, натуральных чисел с автоматическим нарачиванием. Естественное использование - для порядковой нумерации записей.
7. **Логический** - тип для хранения логических данных.
8. **Поле объекта OLE** - специальный тип данных, предназначенный для хранения объектов OLE, например мультимедийных. Реально, конечно, такие объекты в таблице не хранятся. Как и в случае полей Мемо, они хранятся в другом месте внутренней структуры файла базы данных, а в таблице хранятся только указатели на них.
9. **Гиперссылка** - специальное поле для хранения адресов URL для Web-объектов Интернета. При щелчке на ссылке автоматически происходит запуск браузера и воспроизведение объекта в окне.
10. **Мастер подстановок** – это не специальный тип данных. Это объект, настройкой которого можно автоматизировать ввод данных в поле так, что не вводить их вручную, а выбирать из раскрывающегося списка.

2. Объекты базы данных

В СУБД MS Access имеется возможность создания и использования объектов 7 различных типов.

Таблицы. Это основной объект любой БД. В таблицах хранятся все данные, имеющиеся в БД, и структура БД.

Запросы. Это объект, предназначенный для извлечения данных из таблиц и представления их пользователю в удобном виде. С помощью запросов выполняются следующие операции: отбор данных, их сортировка, фильтрация, преобразование данных по заданному алгоритму, создание новой таблицы, автоматическое наполнение таблицы данными и т. д.

Формы. Это объект, предназначенный для ввода данных. Форма позволяет предоставить пользователю средства для заполнения только тех полей, которые ему заполнять положено.

Отчеты. По своим свойствам и структуре отчеты во многом похожи на формы, но предназначены только для вывода, причем для вывода не на экран, а печатающее устройство.

Страницы. Это специальные объекты, реализованные в последних версиях СУБД MS Access называемый страницами доступа к данным. Физически он выполнен в коде HTML, размещаемый на Web-странице и передаваемый вместе с ней. Он содержит компоненты, через которые осуществляется связь переданной Web-странице с БД, остающейся на сервере.

Макросы и модули. Эти объекты предназначены для автоматизации повторяющихся операций при работе с СУБД и для создания новых функций путем программирования.

3. Проектирование базы данных

Методически правильно начинать работу по проектированию БД на листе бумаги с карандашом в руках, не используя компьютера.

3.1. Этапы проектирования базы данных:

3.1.1. Разработка технического задания

Техническое задание на проектирование базы данных должен предоставить заказчик. Однако для этого он должен владеть соответствующей терминологией и знать, хотя бы в общих чертах, технические возможности основных систем управления базами данных. К сожалению, на практике такое положение встречается не всегда. Поэтому обычно используют следующие подходы: • демонстрируя заказчику работу аналогичной базы данных, после чего согласовывают спецификацию отличий; • если аналога нет, выясняют круг задач и потребностей заказчика, после чего помогают ему подготовить техническое задание. При подготовке технического задания составляют:

- 1). Список исходных данных, с которыми работает заказчик;
- 2). Список выходных данных, которые необходимы заказчику для управления структурой своего предприятия;

Список выходных данных, которые не являются необходимыми для заказчика, но которые он должен предоставлять в другие организации (в вышестоящие структуры, в органы статистического учета, прочие административные и контролирующие организации). При этом очень важно не ограничиваться взаимодействием с головным подразделением заказчика, а провести обсуждение со всеми службами и подразделениями, которые могут оказаться поставщиками данных в базу или их потребителями. Так, например, при подготовке базы данных для учета абитуриентов и студентов в высшем учебном заведении необходимо не только изучить документооборот ректората и всех деканатов, но и понять, что хотели бы получить от базы данных службы. Следует изучить работу подразделений, распределяющих учебную нагрузку преподавателей, отвечающих за распределение аудиторного фонда, за проживание студентов в общежитии и других. В расчет должны приниматься и такие службы, как библиотека, отдел кадров и прочие. В любой момент может выясниться, например, что администрация библиотеки должна периодически поставлять кому-то отчеты, характеризующие читательскую активность студентов в зависимости от пола, возраста и социального положения. К возможным пожеланиям заказчика следует готовиться на этапе проектирования, до создания базы.

3.1.2. Разработка схемы данных

Выяснив основную часть данных, которые заказчик потребляет или предоставляет, можно приступить к созданию структуры базы, то есть структуру ее основных таблиц. Разработка схемы включает:

- 1). Составление генерального списка полей (может насчитывать 10-ки позиций).
- 2). Определение наиболее подходящего типа для каждого поля.
- 3). Распределение полей генерального списка по базовым таблицам.

Цель – обеспечить, чтобы ввод данных в одну таблицу производился, по возможности, в рамках одного подразделения, а еще лучше – на одном рабочем месте. Наметив столько таблиц, сколько подразделений охватывает база данных, приступают к дальнейшему делению таблиц. Критерием необходимости деления является факт множественного повтора данных в соседних записях на рис. 1 показана таблица, у которой в поле Адрес наблюдается повтор данных. Это явное свидетельство того, что таблицу надо поделить на две взаимосвязанных таблицы.



Компонент	Модель	Цена	Поставщик	Адрес
Pentium4	1, B	135 000*Самшит	123456, Москва, ул. Индустриальная, д. 15	123456, Москва, ул. Индустриальная, д. 15
Pentium4	1, 8A	145 000*Самшит	123456, Москва, ул. Индустриальная, д. 15	123456, Москва, ул. Индустриальная, д. 15
Pentium4	2, 4A	178 000*Самшит	111222, Москва, ул. Промышленная, д. 22	111222, Москва, ул. Промышленная, д. 22
Pentium4	2, 4B	182 000*Самшит	111222, Москва, ул. Промышленная, д. 22	111222, Москва, ул. Промышленная, д. 22
Pentium4	2, BC	319 000*Самшит	135719, Москва, Западный проспект, д. 134	135719, Москва, Западный проспект, д. 134
Pentium4	3, DC	463	123456, Тверь, ул. Созидателей, д. 159	123456, Тверь, ул. Созидателей, д. 159

- 4). Назначение ключевого поля в каждой таблице.

В качестве такового выбирают поле, данные в котором повторяться не могут. Например, для таблицы данных о студентах таким полем может служить индивидуальный шифр студента. Для таблицы, в которой содержатся расписания занятий, такого поля можно и не найти, но его можно создать искусственным комбинированием полей «Время занятия» и «Номер аудитории». Эта комбинация неповторима, так как в одной аудитории в одно и то же время не принято проводить два различных занятия. Если в таблице вообще нет никаких полей, которые можно было бы использовать как ключевые, всегда можно ввести дополнительное поле типа Счетчик – оно не может содержать повторяющихся данных по определению.

- 5). Определение связи между таблицами (с помощью карандаша и бумаги).

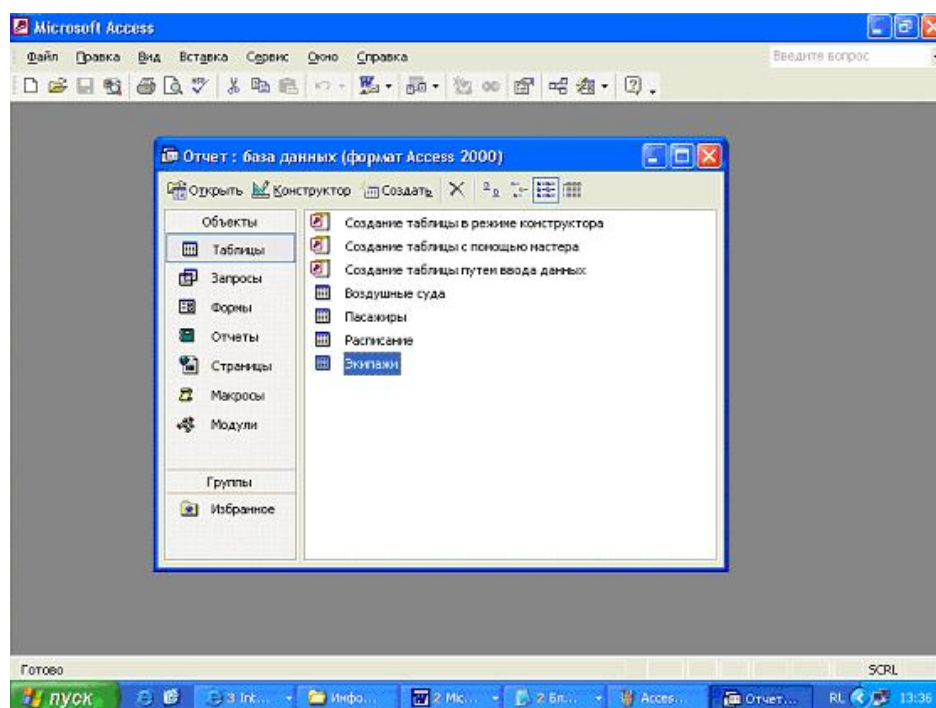
На рис. 2 показан пример взаимосвязи между группой таблиц, составляющих одну базу данных. Такой чертеж называется схемой данных.

Рис. 2 Типы связей между таблицами - «один ко многим» и «один к одному». Связь между таблицами организуется на основе общего поля, причем в одной из таблиц оно обязательно должно быть ключевым, то есть на стороне «один» должно выступать ключевое поле, содержащее уникальные, неповторяющиеся значения. Значения на стороне «многие» могут повторяться. Рассмотрим таблицу Клиенты (рис. 3). Здесь поле Код клиента является ключевым. Это понятно, поскольку у каждого клиента должен быть свой уникальный код, идентифицирующий его однозначно. Если мы рассмотрим таблицу Заказы, то увидим, что в ней код клиента не может быть уникальным, поскольку каждый клиент мог сделать сколько угодно много заказов. На схеме данных эти поля соединены линией связи. С одной стороны эта линия маркирована знаком «1», с другой стороны – знаком «бесконечность». Это графический метод изображения связи «один ко многим». Ключевым полем в таблице заказов является Код заказа – он однозначно идентифицирует, кто, когда, что заказал и на какую сумму. Здесь же можно указать, какой сотрудник принял заказ к исполнению. Поскольку один сотрудник может принять множество заказов, поле Код сотрудника в таблице заказов не является ни уникальным, ни ключевым, зато в таблице Сотрудники это поле уникально. Про подобные таблицы говорят, что они связаны реляционными отношениями. Соответственно, системы управления, способные работать со связанными таблицами, называют системами управления реляционными базами данных, а схему данных в технической литературе могут называть схемой реляционных отношений. Разработкой схемы данных заканчивается «бумажный» этап работы над техническим предложением. Эту схему можно согласовать с заказчиком, после чего приступить к непосредственному созданию базы данных. Следует помнить, что по ходу разработки проекта заказчику непременно будут приходить в голову новые идеи. На всех этапах проектирования он стремится охватить единой системой все новые и новые подразделения и службы предприятия. Возможность гибкого исполнения его пожеланий во многом определяется квалификацией разработчика базы данных. Если схема данных составлена правильно, подключать к базе новые таблицы нетрудно. Если структура базы нерациональна, разработчик может испытать серьезные трудности и войти в противоречия с заказчиком. Противоречия исполнителя с заказчиком всегда свидетельствуют о недостаточной квалификации исполнителя. Именно поэтому этап предварительного проектирования базы данных следует считать основным. От его успеха зависит, насколько база данных станет удобной и будут ли с ней работать пользователи. Если отмечается, что пользователи базы «саботируют» ее эксплуатацию и предпочитают работать традиционными методами, это говорит не о низкой квалификации пользователя, а о недостаточной квалификации разработчика базы. На этом этапе завершается предварительное проектирование базы данных, и на следующем этапе начинается ее непосредственная разработка. С этого момента следует начать работу с системой управления базами данных.

4. Работа с СУБД Microsoft Access

4.1. Создание таблиц

Работа с любыми объектами начинается с окна База данных (рис. 4).



На левой панели данного окна сосредоточены элементы управления для вызова всех семи типов объектов программы (Таблица, Запрос, Форма, Отчет, Страницы, Макросы, Модули). Создание таблиц начинается с выбора элемента управления Таблицы. Окно Конструктора таблиц представлено на рис. 5.

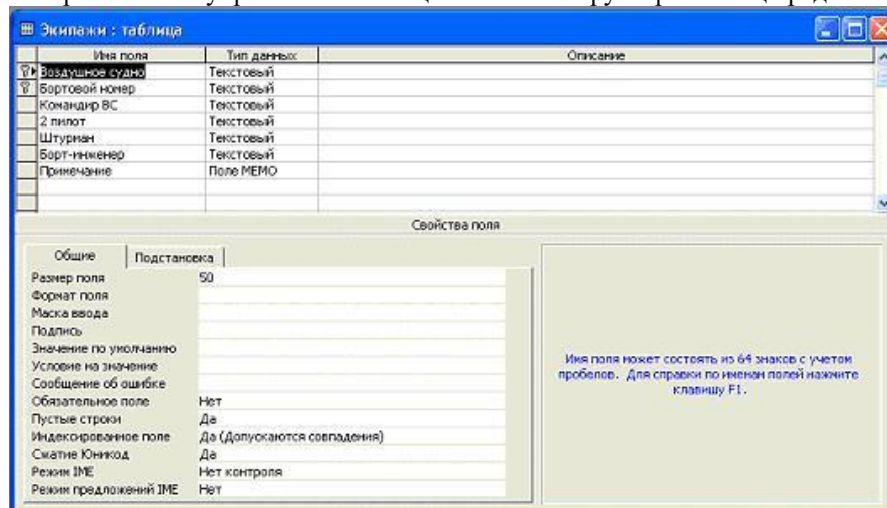


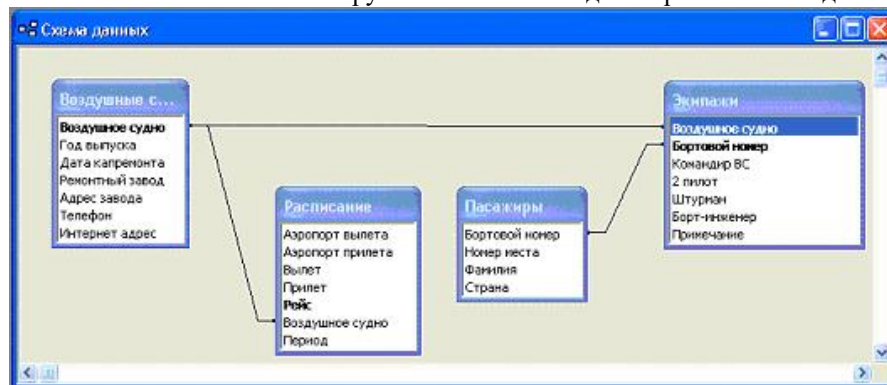
Рис. 5

Нижняя часть бланка содержит список свойств поля, выделенного в верхней части. Некоторые из свойств уже заданы по умолчанию. Свойства полей не являются обязательными. Их можно настраивать по желанию, а можно и не трогать.

После наполнения таблицы данными сохранять их не надо – все сохраняется автоматически. Однако если при работе с таблицей произошло редактирование ее макета (например, изменялась ширина столбцов), СУБД попросит подтвердить сохранение этих изменений. Если на этапе проектирования базы данных была четко разработана структура таблиц, то создание таблиц с помощью Конструктора происходит очень быстро и эффективно. Даже без использования автоматизированных средств создание основы для достаточно крупных проектов происходит в считанные минуты – это ценное свойство СУБД Microsoft Access, но оно реализуется при обязательном условии тщательной предварительной подготовки.

4.2. Создание межтабличных связей

Осуществляется в специальном окне Схема данных и выполняется с помощью мыши. Окно Схема данных открывают кнопкой на панели инструментов или командой Сервис – Схема данных



Образовавшаяся межтабличная связь отображается в окне Схема данных в виде линии, соединяющей два поля разных таблиц. При этом одна из таблиц считается главной, а другая – связанной. Главная – это таблица, которая участвует в связи своим ключевым полем (название этого поля на схеме данных отображается полужирным шрифтом).

Связь между таблицами позволяет:

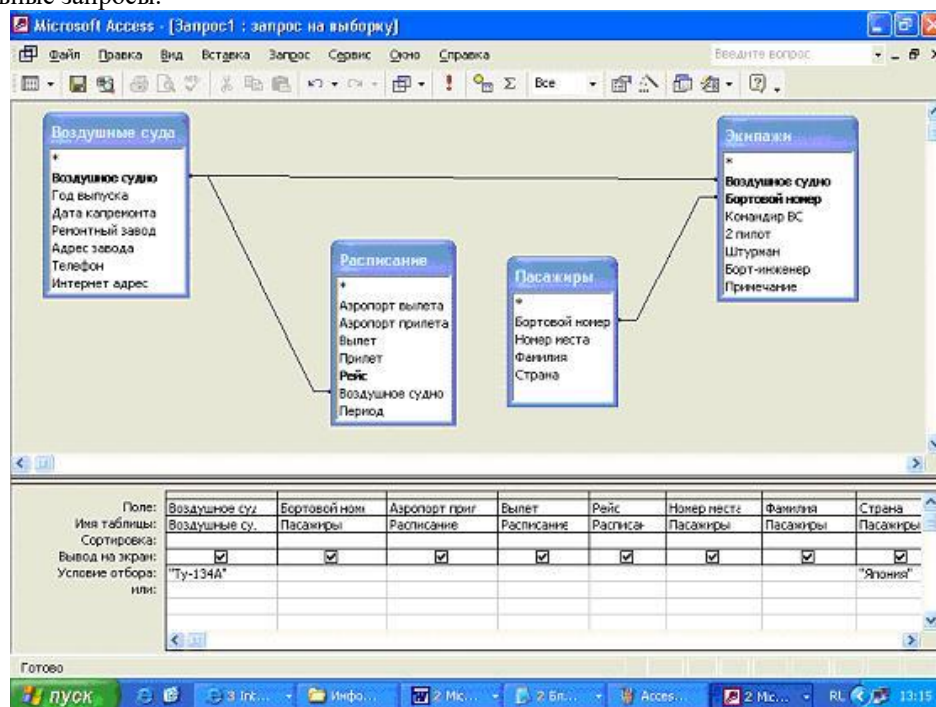
- Либо исключить возможность удаления или изменения данных в ключевом поле главной таблицы, если с этим полем связаны какие-либо поля других таблиц;
- Либо сделать так, что при удалении (или изменении) данных в ключевом поле главной таблицы автоматически (и абсолютно корректно) произойдет удаление или изменение соответствующих данных в полях связанных таблиц.

Таким образом, смысл создания реляционных связей между таблицами состоит, с одной стороны, в защите данных, а с другой стороны – в автоматизации внесения изменений сразу в несколько таблиц при изменениях в одной таблице.

4.3. Создание запросов

Запросы используются для получения данных из базы. Запрос готовится с помощью Конструктора в специальном бланке запроса по образцу. Виды запросов:

- 1).запрос на выборку
- 2).запрос с параметром;
- 3).итоговый запрос;
- 4).запросы по изменению;
- 5).перекрестные запросы;
- 6).специальные запросы.



Вид полученного запроса

	Воздушное судно	Бортовой номер	Аэропорт прилета	Вылет	Рейс	Номер места	Фамилия	Страна
▶	Ту-134А	75643	Иркутск	21.45.00	15 3а	Ли К.	Япония	Япония
	Ту-134А	75643	Иркутск	21.45.00	15 5а	Лимбья С.	Япония	Япония

4.4. Создание форм

Существует два вида организации структуры форм:

1. на основе таблицы;
2. на основе запроса.

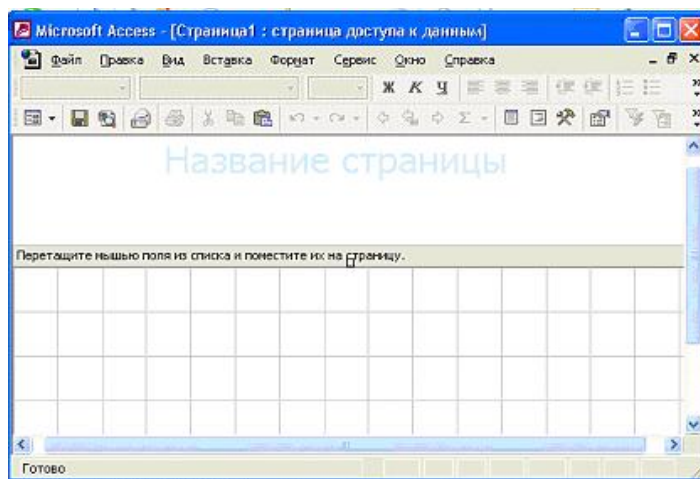
Запросы создаваться средствами автоматизации с использованием мастера форм.

Существует три вида автоформ: в столбец, ленточные и табличные.

Воздушное судно	Ту-134А
Бортовой номер	75643
Аэропорт прилета	Иркутск
Вылет	21.45.00
Рейс	15
Номер места	3а
Фамилия	Ли К.
Страна	Япония

4.5. Создание страниц

Создание страниц осуществляется с помощью Мастера страниц.



4.6. Создание отчетов

Средством автоматизированного создания отчетов является Мастер отчетов.

Мастер отчетов позволяет:

- 1). выбор базовых таблиц или запросов;
- 2). выбор полей, отображаемых в отчете;
- 3). выбор полей группировки;
- 4). выбор полей и методов сортировки;
- 5). выбор форм печатного макета и стиля оформления.

