

**ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ
УНИВЕРСИТЕТ имени академика С.П. КОРОЛЁВА
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)»**

**СОЗДАНИЕ И РЕДАКТИРОВАНИЕ
БАЗ ДАННЫХ
В СУБД MICROSOFT ACCESS**

САМАРА 2011

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ
УНИВЕРСИТЕТ имени академика С.П. КОРОЛЁВА
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)»

СОЗДАНИЕ И РЕДАКТИРОВАНИЕ
БАЗ ДАННЫХ
В СУБД MICROSOFT ACCESS

*Утверждено Редакционно-издательским советом университета
в качестве методических указаний*

САМАРА
Издательство СГАУ
2011

УДК СГАУ:004.422.8(075)
ББК 22.18

Составитель *М.С. Стенгач*

Рецензент д-р техн. наук, проф. *С.А. Ишков*

Создание и редактирование баз данных в СУБД Microsoft Access: метод. указания / сост. *М.С. Стенгач*. – Самара: Изд-во Самар. гос. аэрокосм. ун-та, 2011. – 20 с.: ил.

Приведены основные сведения по созданию структуры базы данных и установлению связей между таблицами в системе управления базами данных Microsoft Access.

Предназначены для студентов, изучающих дисциплину «Информатика».

СОДЕРЖАНИЕ

1. Цели лабораторной работы.....	4
2. Основные сведения.....	4
3. Задание к лабораторной работе	7
4. Создание структуры базы данных.....	8
5. Установление связей между таблицами	13
6. Создание форм для ввода данных в таблицы.....	15
7. Контрольные вопросы	17
Список литературы.....	19

1. Цели лабораторной работы

Целями лабораторной работы являются:

- получение навыков по созданию структуры базы данных (БД);
- отработка практических навыков по вопросам создания различных типов полей и задания их свойств;
- умение использовать мастер подстановок, маску ввода и поля со списком;
- умение устанавливать связи между таблицами;
- умение создавать формы для ввода информации в базу данных;
- умение импортировать данные в базу данных из внешних файлов.

2. Основные сведения

База данных – это совокупность взаимосвязанных данных, организованных на внешних носителях информации средствами системы управления базами данных (СУБД).

Microsoft Access – это система управления базами данных. Под системой управления базами данных понимается программа, которая не только позволяет хранить большие массивы данных в определённом формате, но и обрабатывать их, представляя в удобном для пользователей виде.

Для более эффективной работы с СУБД Access важно знать терминологию и понимать принцип построения и основные преимущества реляционной (многотабличной) базы данных.

Система баз данных поддерживает в памяти ЭВМ модель предметной области. Каждая СУБД предоставляет свой инструментарий для отображения предметной области. Этот инструментарий принято называть моделью данных. Модель данных определяется тремя компонентами:

- допустимой организацией данных;
- ограничениями целостности;
- множеством операций, допустимых над объектами модели данных.

Допустимая организация данных определяется разнообразием и количеством типов объектов модели данных, ограничениями на структуру данных.

Ограничения целостности поддерживаются средствами, предусмотренными в модели данных для выражения ограничений на значения данных и связи, которые характеризуют достоверные состояния БД. Ряд ограничений целостности поддерживается моделью данных по умолчанию и распространяется на все типовые ситуации, возникновение которых возможно при внесении изменений в БД. Другие ограничения целостности могут задаваться явно и также распространяться на множество однотипных ситуаций. Ограничения целостности могут распространяться и на значения отдельных полей. Например, если при описании поля записи задать область допустимых значений, то система не допустит обновления данного, если новое значение будет квалифицировано как некорректное.

Множество операций определяет виды обработок, которым могут подвергаться объекты модели данных. Сюда, в первую очередь, входят операции выборки данных и операции, изменяющие состояние БД. Рассмотрим подробнее организацию данных и ограничения целостности (допустимые операции будут рассмотрены при практической работе с базой данных).

Организация данных. Слово «реляционная» происходит от английского relation – отношение. Отношение – математическое понятие, но в терминологии моделей данных отношения удобно изображать в виде таблицы, в которой данные располагаются по столбцам (полям) и строкам (записям).

Столбец таблицы, у которого данные в строках уникальны (не повторяются), может быть определён как ключ. В СУБД Access можно определить также и составной ключ. У составного ключа данные в строках нескольких столбцов могут повторяться, но комбинация данных каждой строки этих столбцов является уникальной. Например, в таблице ПАССАЖИРЫ есть столбцы *Фамилия* и *Дата рождения*. В каждом из столбцов есть некоторые повторяющиеся данные, т.е. одинаковые фамилии и одинаковые даты рождения. Но если пассажиры, имеющие одинаковые фамилии, имеют разные даты рождения, то эти столбцы можно использовать в качестве составного ключа.

В Access можно выделить три типа ключевых полей: 1) простой ключ; 2) составной ключ; 3) счетчик.

Поле счетчика содержит номера строк, и Access обычно предлагает его в качестве ключа, если не выбрано другое ключевое поле.

Одно из важнейших достоинств реляционных баз данных состоит в том, что вы можете хранить логически сгруппированные данные в разных таблицах и задавать связи между ними, объединяя их в единую базу. Для задания связи таблицы должны иметь поля с одинаковыми именами или хотя бы с одинаковыми форматами данных. Связь между таблицами устанавливает отношения между совпадающими значениями в этих полях. Такая организация данных позволяет уменьшить избыточность хранимых данных, упрощает их ввод и организацию запросов и отчетов. Поясним это на примере. Допустим, что в базе должны храниться данные обо всех рейсах (номер рейса, аэропорт вылета, аэропорт прилета) и билетах, купленных на эти рейсы за последние четыре года (номер рейса, дата, информация о пассажире). Если хранить данные в одной таблице, то в строке с номером рейса будут храниться данные обо всех билетах, купленных на этот рейс за последние четыре года. Это является огромной избыточностью данных. Если же хранить данные о рейсах в одной таблице, а о купленных билетах – в другой и установить связь между полями «номер рейса», то избыточность хранимых данных многократно уменьшится.

В Access можно задать три вида связей между таблицами: «один-ко-многим», «многие-ко-многим» и «один-к-одному».

Связь «один-ко-многим» является наиболее часто используемым типом связи между таблицами. При такой связи каждой записи в таблице А могут соответствовать несколько записей в таблице В (эти записи называют внешними ключами), а запись в таблице В не может иметь более одной соответствующей ей записи в таблице А.

При связи «многие-ко-многим» одной записи в таблице А могут соответствовать несколько записей в таблице В, а одной записи в таблице В – несколько записей в таблице А. Такая схема реализуется только с помощью третьей (связующей) таблицы, ключ которой состоит из, по крайней мере, двух полей, одно из которых является общим с таблицей А, а другое – общим с таблицей В.

При связи «один-к-одному» запись в таблице А может иметь не более одной связанной записи в таблице В и наоборот. Этот тип связи

используют не очень часто, поскольку такие данные могут быть помещены в одну таблицу. Связь с отношением «один-к-одному» используют для разделения очень широких таблиц, для отделения части таблицы по соображениям защиты, а также для сохранения сведений, относящихся к подмножеству записей в главной таблице.

Целостность данных. Целостность данных означает систему правил, используемых в Access для поддержания связей между записями в связанных таблицах, а также обеспечивает защиту от случайного удаления или изменения связанных данных. Установить целостность данных можно, если выполнены следующие условия:

- связанное поле главной таблицы является ключевым полем или имеет уникальный индекс;
- связанные поля имеют один тип данных (здесь существует исключение – поле счетчика может быть связано с числовым полем, если в последнем в свойстве «Размер поля» указано значение «Длинное целое»);
- обе таблицы принадлежат одной базе данных Access;
- для установки целостности данных база данных, в которой находятся таблицы, должна быть открыта.

Невозможно определить условия целостности данных для присоединённых таблиц из баз данных других форматов.

3. Задание к лабораторной работе

Требуется разработать базу данных АЭРОПОРТ, содержащую данные об авиакомпаниях, прямых рейсах, пассажирах и билетах, купленных на все рейсы за последние четыре года.

Таким образом, база данных должна состоять из четырёх таблиц: АВИАКОМПАНИИ, ПАССАЖИРЫ, РЕЙСЫ, БИЛЕТЫ.

Таблица АВИАКОМПАНИИ должна содержать следующие поля:

Код авиакомпании

Название

Адрес

Телефон

Таблица ПАССАЖИРЫ должна содержать следующие поля:

Код пассажира

Фамилия

Имя

Отчество

Паспорт

Дата рождения

Пол

Телефон

Таблица РЕЙСЫ должна содержать следующие поля:

Номер рейса

Аэропорт вылета

Аэропорт прилета

Код авиакомпании

Таблица БИЛЕТЫ должна содержать следующие поля:

Код пассажира

Номер рейса

Цена билета

Дата вылета

4. Создание структуры базы данных

1. Создайте базу данных АЭРОПОРТ, выполнив следующие действия:

- загрузите Access, в появившемся окне выберите пункт **Файл=>Создать...** и в правой части экрана выберите пункт **Новая база данных...**;
- в окне **Файл новой базы данных** задайте в пункте **Имя файла** свою фамилию (по умолчанию Access предлагает имя базы db1) и выберите свою папку на диске;
- нажмите кнопку **Создать**.

2. Создайте структуру таблицы АВИАКОМПАНИИ. Для этого в появившемся окне базы данных дважды щёлкните на пункте **Создание таблицы** в режиме конструктора. В результате проделанных операций откроется окно таблицы в режиме конструктора.

Определите поля таблицы базы данных в соответствии с табл. 1.

Таблица 1

Имя поля	Тип данных	Размер поля
<i>Код авиакомпании</i>	Числовой	Целое
<i>Название</i>	Текстовый	20
<i>Адрес</i>	Текстовый	30
<i>Телефон</i>	Текстовый	7

Размер поля задаётся на вкладке **Общие** панели **Свойство поля**, которое занимает нижнюю часть экрана.

В качестве ключевого поля задайте поле *Код авиакомпании*. Для этого щёлкните правой кнопкой мыши по полю *Код авиакомпании* и выберите из открывшегося списка **Ключевое поле**. Слева от поля *Код авиакомпании* появится изображение ключа.

Сохраните созданную таблицу. Для этого:

- выберите пункт **Файл=>Сохранить**;
- в появившемся окне **Сохранение** задайте имя таблицы: **АВИАКОМПАНИИ** и нажмите **ОК**.

Щёлкните по полю **Телефон** и на панели **Свойство поля** на вкладке **Общие** введите в пункте **Маска ввода** следующую маску: **999-99-99**. Пункт **Размер поля** задайте равным **7**. Это позволит не набирать тире в номере телефона при его вводе и при этом номер телефона не сможет содержать более семи цифр.

Закройте окно **АВИАКОМПАНИИ: таблица**. В появившемся диалоговом окне на запрос **Сохранить изменения макета или структуры объекта "таблица 'АВИАКОМПАНИИ' "?** ответить **Да**.

В окне базы данных появится новый пункт: **АВИАКОМПАНИИ**.

Откройте таблицу **АВИАКОМПАНИИ** (дважды щёлкните на её названии) и добавьте в неё сведения об авиакомпаниях. **Обязательно** присвойте каждой авиакомпании уникальный код (любое целое число).

3. Таблица **ПАССАЖИРЫ** (табл. 2) будет импортирована из таблицы **EXCEL**. Файл **группаNNN.xls** (где **NNN** – номер группы) создан ранее во время выполнения лабораторной работы №6.

- выполните команду **Файл=>Внешние данные=>Импорт**;
- выберите тип файла **Microsoft Excel**;
- выберите файл **группаNNN.xls** и нажмите кнопку **Импорт**;
- выберите переключатель **листы, Лист1** и нажмите кнопку **Далее**;

- выберите переключатель Первая строка содержит заголовки столбцов и нажмите кнопку Далее;
 - данные сохраняем в новой таблице – нажмите кнопку Далее;
 - импортируются все поля – нажмите кнопку Далее;
 - выберите переключатель Определить ключ – в качестве ключевого задайте поле *Код пассажира* и нажмите кнопку Далее;
 - задайте имя таблицы: ПАССАЖИРЫ и нажмите кнопку Готово.
- Отредактируйте созданную таблицу в режиме конструктора (табл. 2).

Т а б л и ц а 2

Имя поля	Тип данных	Размер поля
<i>Код пассажира</i>	Числовой	Целое
<i>Фамилия</i>	Текстовый	15
<i>Имя</i>	Текстовый	12
<i>Отчество</i>	Текстовый	15
<i>Паспорт</i>	Текстовый	10 (маска)
<i>Дата рождения</i>	Дата/время	
<i>Пол</i>	Текстовый	1
<i>Телефон</i>	Текстовый	10 (маска)

При определении поля *Паспорт* используем маску для ввода серии и номера паспорта. Для этого щёлкните на поле *Паспорт* и на панели Свойство поля на вкладке Общие введите в пункте Маска ввода следующую маску: 99 99 999999. Пункт Размер поля задайте равным 10. Таким образом, серия и номер паспорта будут вводиться в стандартном виде, например, 36 09 147262 и при этом общее количество цифр в серии и номере паспорта не может быть более десяти.

В поле *Дата рождения* используем маску для удобного ввода даты (т.е. в датах точки будут вводиться автоматически). Установите курсор на поле Маска ввода – справа появится кнопка с тремя точками. Нажмите на эту кнопку. В появившемся окне создания масок выберите Краткий формат даты и нажмите кнопку Готово.

В поле *Пол* создайте поле со списком без ввода новых значений. Для этого выберите для поля *Пол* тип данных Мастер подстановок. В появившемся окне Создание подстановки выберите пункт Будет введен фиксированный набор значений и нажмите кнопку Далее. Запол-

ните только две ячейки ниже надписи Столбец 1: м и ж. Нажмите кнопку **Далее**. Подтвердите, что подпись столбца подстановки будет *Пол* и нажмите кнопку **Готово**.

В свойствах поля *Пол* на вкладке **Общие** задайте **Значение по умолчанию** равным “*м*” (кавычки обязательны).

Щёлкните по полю **Телефон** и на вкладке **Общие** введите в пункте **Маска ввода** следующую маску для сотового телефона: **8-999-999-99-99**. Это позволит не набирать первую восьмёрку и тире в номере телефона при его вводе.

4. Создайте структуру таблицы РЕЙСЫ аналогично п. 2 в соответствии с табл. 3.

Т а б л и ц а 3

Имя поля	Тип данных	Размер поля
<i>Номер рейса</i>	Числовой	Целое
<i>Аэропорт вылета</i>	Текстовый	20
<i>Аэропорт прилета</i>	Текстовый	20
<i>Код авиакомпании</i>	Числовой	Целое

В качестве ключевого задайте поле *Номер рейса*. Для этого щёлкните по полю *Номер рейса* и выполните команду **Правка=>Ключевое поле**.

Сохраните созданную таблицу. Для этого:

- выберите пункт **Файл=>Сохранить**;
- в появившемся окне **Сохранение** задайте имя таблицы: **РЕЙСЫ** и нажмите **ОК**.

Так как нужно создать таблицу прямых рейсов, в поле *Аэропорт вылета* задаём **Значение по умолчанию**: *Самара*.

В поле *Аэропорт прилета* используем мастер подстановок для того, чтобы не вводить, а выбирать из списка название города с возможностью ввода нового названия, которого нет в списке. Выберите для поля *Аэропорт прилета* тип данных **Мастер подстановок**. Появится окно **Создание подстановки**. Выберите пункт **Будет введен фиксированный набор значений** и нажмите кнопку **Далее**.

Заполните ячейки ниже надписи **Столбец 1** названиями аэропортов прилета:

Москва
Санкт-Петербург
Сочи
Владивосток
Минск

Нажмите кнопку **Далее**. Подтвердите, что подпись столбца подстановки будет *Аэропорт прилета*, и нажмите кнопку **Готово**.

Посмотрите, какие изменения появились на вкладке **Подстановка** панели **Свойство поля**. Проверьте строку **Ограничиться списком**, в которой должно стоять слово **Нет**.

Поле *Код авиакомпании* будет заполняться при помощи мастера подстановок из таблицы **АВИАКОМПАНИИ**. Для этого необходимо выполнить следующие действия:

- выберите для поля *Код авиакомпании* тип данных **Мастер подстановок**;
- в окне **Создание подстановки** выберите пункт **Объект “столбец подстановки”** будет использовать значения из таблицы или запроса и нажмите кнопку **Далее**;
- выберите пункт **Таблицы**;
- выберите таблицу **АВИАКОМПАНИИ** и нажмите кнопку **Далее**;
- из доступных полей таблицы **АВИАКОМПАНИИ** выберите *Код авиакомпании* и нажмите кнопку со значком **>**. Поле *Код авиакомпании* переместится в окно **Выбранные поля**;
- аналогично выберите остальные поля таблицы **АВИАКОМПАНИИ** и нажмите кнопку **Далее**;
- выберите в первом выпадающем списке *Название* (список авиакомпаний будет сортироваться по названию в порядке возрастания) и нажмите кнопку **Далее**;
- скройте ключевой столбец – после работы мастера при заполнении поля *Код авиакомпании* таблицы **РЕЙСЫ** будут отображаться названия авиакомпаний для выбора, но в таблице **АВИАКОМПАНИИ** поле *Код авиакомпании* будет оставаться числовым целым;
- нажмите кнопку **Далее**;
- подтвердите, что подпись столбца подстановки будет *Код авиакомпании*, и нажмите кнопку **Готово**.

Появится окно **Создание подстановки**. На вопрос о сохранении таблицы ответьте **Да**.

Закройте таблицу РЕЙСЫ в режиме конструктора.

Откройте таблицу РЕЙСЫ в окне базы данных. Добавьте в неё:

- уникальный код каждого рейса (любое целое число);
- значения поля *Код авиакомпании* выбирайте из списка подстановки.

5. Создайте структуру таблицы БИЛЕТЫ в соответствии с табл. 4.

Таблица 4

Имя поля	Тип данных	Размер поля
<i>Код пассажира</i>	Числовой	Целое
<i>Номер рейса</i>	Числовой	Целое
<i>Цена билета</i>	Денежный	
<i>Дата вылета</i>	Дата/время	

Для поля Цена билета на вкладке Общие панели Свойство поля установите Условие на значение равным >0 . Сообщение об ошибке задайте: *Бесплатным бывает только сыр в мышеловке*. Для проверки условия на введённое значение попробуйте ввести отрицательную или нулевую стоимость билета.

У таблицы БИЛЕТЫ ключ будет составной. Выделите два поля: *Код пассажира*, *Номер рейса* и задайте ключ.

Сохраните созданную таблицу. Заполните таблицу: «отправьте» себя и своих друзей в те города, где вам хотелось бы побывать.

5. Установление связей между таблицами

Разработайте схему данных, т.е. создайте связи между таблицами. Для этого:

- выполните команду Сервис=>Схема данных... На экране появится окно Схема данных;
- выполните команду Связи=>Добавить таблицу...;
- появится окно Добавление таблицы. Так как таблицы РЕЙСЫ и АВИАКОМПАНИИ уже имеются в схеме данных, выделите название таблицы БИЛЕТЫ и нажмите кнопку Добавить;
- выделите название таблицы ПАССАЖИРЫ и нажмите кнопку Добавить;
- закройте окно, нажав кнопку Закреть.

Создайте связь между таблицами РЕЙСЫ и БИЛЕТЫ. Для этого подведите курсор мыши к полю *Номер рейса* в таблице РЕЙСЫ, нажмите левую клавишу мыши и, не отпуская её, перетащите курсор на поле *Номер рейса* в таблице БИЛЕТЫ, а затем отпустите левую клавишу мыши. На экране откроется окно Изменение связей.

Щёлкните по ячейке Обеспечение целостности данных – в ней должна появиться галочка.

Щёлкните по ячейкам Каскадное обновление связанных полей и каскадное удаление связанных записей. Задание каскадного обновления связанных полей и каскадного удаления связанных записей позволит редактировать записи только в таблице РЕЙСЫ, а в таблице БИЛЕТЫ эти действия со связанными записями будут выполняться автоматически. Например, если вы удалите из таблицы РЕЙСЫ какой-либо рейс, то в таблице БИЛЕТЫ удалятся все строки, связанные с этим рейсом.

Нажмите кнопку Создать. Связь будет создана.

Аналогично создайте связи между полем *Код пассажира* в таблице ПАССАЖИРЫ и полем *Код пассажира* в таблице БИЛЕТЫ.

Связь между полем *Код авиакомпании* в таблице АВИАКОМПАНИИ и полем *Код авиакомпании* в таблице РЕЙСЫ была установлена ранее при помощи мастера подстановок.

В результате должна получиться схема данных, показанная на рис. 1.

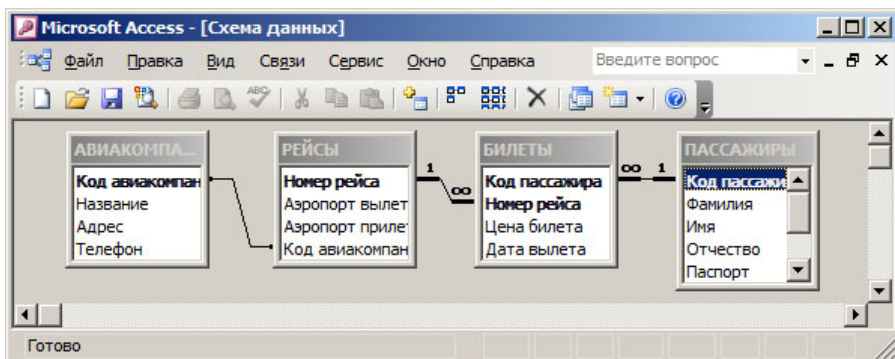


Рис. 1. Схема данных

Закройте окно схемы данных, ответив Да на вопрос о сохранении макета 'Схема данных'.

6. Создание форм для ввода данных в таблицы

Формы для ввода данных позволяют более наглядно представлять, вводить и обрабатывать информацию в таблицах БД.

В окне базы данных откройте вкладку **Формы** и нажмите кнопку **Создать**.

В открывающемся списке выберите таблицу **РЕЙСЫ**.

Выберите пункт **Автоформа: в столбец**.

Нажмите кнопку **ОК**. Форма для ввода данных создана (рис. 2).

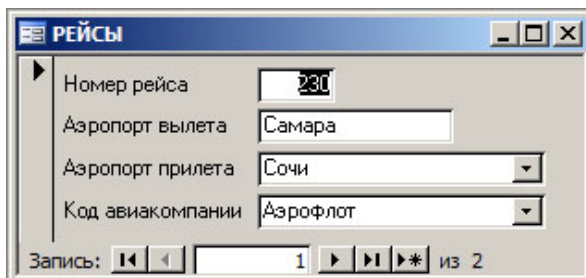


Рис. 2. Форма РЕЙСЫ

Закройте форму РЕЙСЫ. В появившемся окне на запрос **Сохранить изменение макета или структуру объекта?** ответьте **Да**. Сохраните форму с именем РЕЙСЫ. Теперь корректировать информацию в таблице РЕЙСЫ можно с помощью удобной формы.

Аналогично создайте форму для ввода данных в таблицу **АВИАКОМПАНИИ**.

Создайте ленточную автоформу на основе таблицы **ПАССАЖИРЫ** (рис. 3).

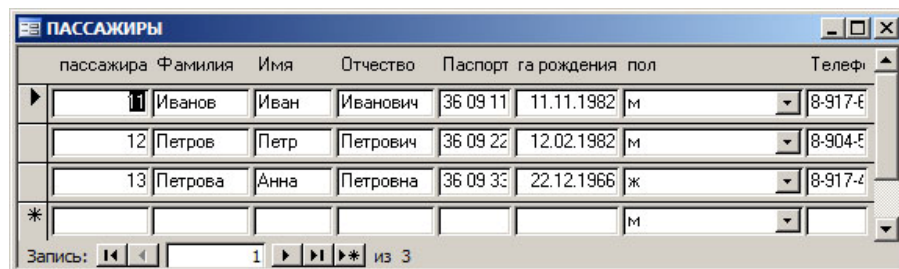


Рис. 3. Форма ПАССАЖИРЫ

Откройте форму РЕЙСЫ в режиме конструктора (рис. 4).

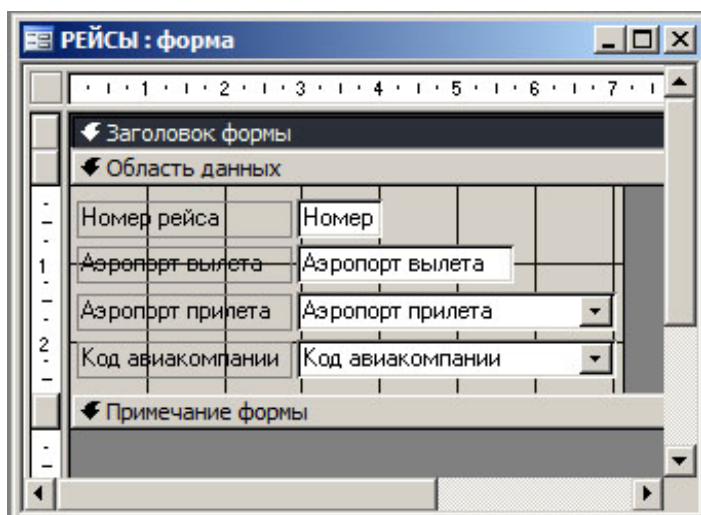


Рис. 4. Форма РЕЙСЫ в режиме конструктора

Одновременно откроется плавающая Панель элементов, с помощью которой на форму можно добавлять новые элементы управления (рис. 5).

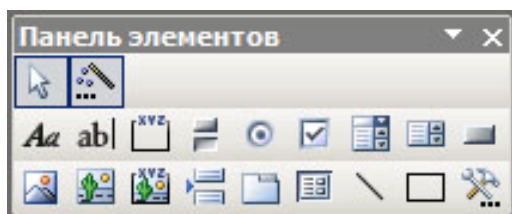


Рис. 5. Панель элементов

Преобразуйте в режиме конструктора форму РЕЙСЫ к виду, показанному на рис. 6.

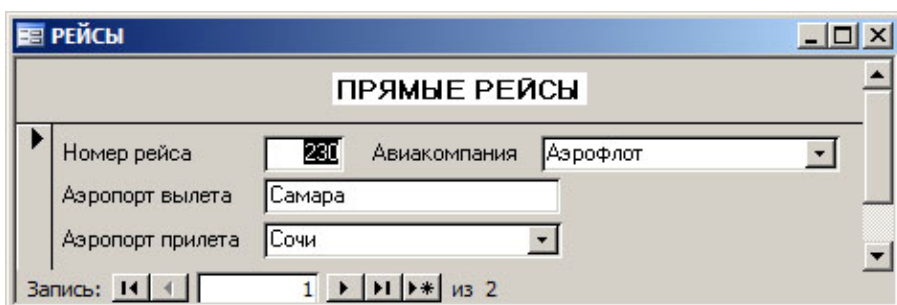


Рис. 6. Форма РЕЙСЫ

Для этого увеличьте область заголовка формы и разместите элемент Надпись с Панели элементов (щёлкните по значку Надпись на Панели элементов, а затем щёлкните в области заголовка). Наберите заголовок ПРЯМЫЕ РЕЙСЫ.

Для изменения размера или положения элемента его необходимо предварительно выделить. Если указатель мыши принимает форму раскрытой ладони, можно перетащить элемент в новое место, а также двойным щелчком раскрыть окно свойств данного элемента. Перетащить поле и присоединённую к нему надпись можно независимо друг от друга, если подвести курсор к чёрному квадратику в левом верхнем углу элемента (указатель мыши примет форму сжатой ладони с вытянутым указательным пальцем). Для выделения нескольких элементов необходимо удерживать нажатой клавишу Shift.

Для выравнивания элементов по размеру или расположению можно воспользоваться пунктом Формат главного меню.

Преобразуйте форму ПАССАЖИРЫ в режиме конструктора к виду, показанному на рис. 7.

Код пассажира	Фамилия	Имя	Отчество	Паспорт	Дата рождения	пол	Телефон
	Иванов	Иван	Иванович	36 09 111111	11.11.1982	м	8-917-666-66-66

Рис. 7. Форма ПАССАЖИРЫ

При помощи Мастера форм создайте форму для таблицы БИЛЕТЫ. Выбор внешнего вида сделайте самостоятельно.

7. Контрольные вопросы

1. К какому классу относится СУБД Access?
2. Чем определяется модель данных?
3. Каковы особенности реляционной модели данных?
4. Как создать новую базу данных в Access?
5. Как добавить новый объект в существующую базу данных?
6. Какие способы создания таблиц Вы знаете? В каких случаях следует использовать каждый из них?

7. Какие типы полей допустимы в Access? Каковы особенности работы с полями каждого из этих типов?
8. Какие способы создания полей подстановки Вы знаете? В каком случае следует использовать каждый из них?
9. Какие преимущества даёт использование полей подстановки?
10. Какие ограничения накладываются на имена полей?
11. Что называется ключом таблицы? Какие разновидности ключей Вы знаете?
12. Какими способами можно создать ключ?
13. В каких случаях задание ключа в Access является обязательным?
14. Какими специфическими особенностями обладает поле типа «счетчик»?
15. Какие свойства полей Вы знаете? Приведите примеры их использования.
16. Как можно изменить структуру существующей таблицы?
17. Как можно задать объединение таблиц? Какие способы объединения Вы знаете?
18. Что означает целостность данных в Access?
19. Что такое «ограничение целостности»?
20. Какие условия необходимы для установления целостности данных?
21. В чём важность задания ограничения целостности?
22. Что такое «ограничение целостности связи» и как оно может задаваться в Access?
23. Какие способы задания ограничения целостности в Access Вы знаете?
24. Чем определяется связь между таблицами «один-ко-многим»?
25. Какие записи называют внешними ключами?
26. Какие разновидности экранных форм Вы знаете?
27. Каково назначение экранных форм?
28. Какими способами можно создавать экранную форму?
29. Как можно скорректировать ранее созданную электронную форму?
30. Каким образом можно менять размещение элементов на экране?

Список литературы

1. Золотова, С.И. Практикум по Access / С.И. Золотова. – М.: Финансы и статистика, 2007. – 143 с.
2. Гринченко, Н.Н. Проектирование баз данных СУБД MICROSOFT ACCESS: учеб. для вузов / Н.Н. Гринченко, Е.В. Гусев, Н.П. Макаров. – М.: Горячая линия – Телеком, 2004. – 240 с.
3. Степанов, В.В. Microsoft Access 2003 для начинающих / В.В. Степанов. – М.: Аквариум; Киров: Дом печати – ВЯТКА, 2006. – 127 с.
4. Хомоненко, А.Д. Базы данных: учеб. для вузов / А.Д. Хомоненко. – М.: Бином-Пресс; СПб.: КОРОНА-Век, 2007. – 736 с.
5. Голицына, О.Л. Базы данных / О.Л. Голицына. – М.: ФОРУМ: ИНФРА-М, 2007. – 399 с.
6. Малыхина, М.П. Базы данных: основы, проектирование, использование / М.П. Малыхина. – СПб.: БХВ-Петербург, 2004. – 499 с.
7. Фуфаев, Э.В. Базы данных / Э.В. Фуфаев. – М.: Academia, 2005. – 320 с.
8. Власовец, А.М. Основы проектирования баз данных реляционных СУБД: учеб. пособие / А.М. Власовец. – СПб.: Изд-во СПбГУЭФ, 2001. – 144 с.

Учебное издание

**СОЗДАНИЕ И РЕДАКТИРОВАНИЕ
БАЗ ДАННЫХ
В СУБД MICROSOFT ACCESS**

Методические указания

Составитель *Стенгач Михаил Сергеевич*

Редактор Т.С. Петренко
Доверстка Т.С. Петренко

Подписано в печать 13.04.2011. Формат 60×84 1/16.

Бумага офсетная. Печать офсетная.

Печ. л. 1,25.

Тираж экз. Заказ . Арт. /2011.

Самарский государственный
аэрокосмический университет им. академика С.П. Королёва.
443086, Самара, Московское шоссе, 34.

Изд-во Самарского государственного
аэрокосмического университета им. академика С.П. Королёва.
443086, Самара, Московское шоссе, 34.