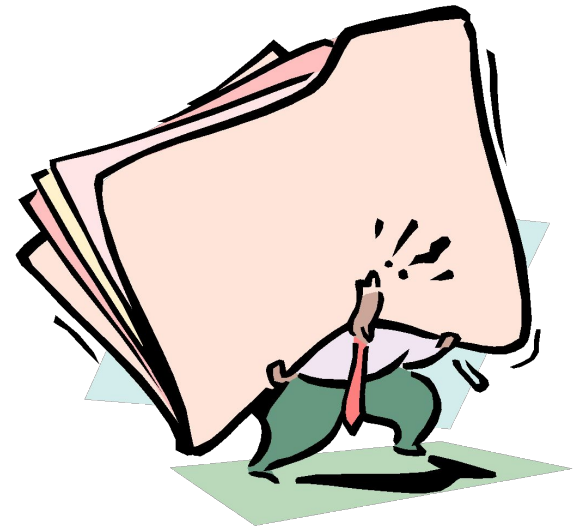
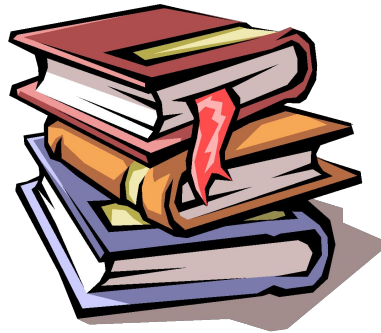


Базы данных

План:

1. Информационные системы. Структурирование данных
2. БД и СУБД.
3. Модели данных. Основные виды моделей данных: (иерархическая, сетевая, реляционная)
4. Основные структуры данных реляционной модели:
 - а) реляционные таблицы и их свойства
 - б) потенциальные, первичные и внешние ключи
 - в) типы связей между реляционными таблицами
5. Целостность данных
6. Реляционные операторы и язык SQL

- С возникновением письменности и изобретением бумаги человек обрел возможность накапливать, хранить и передавать информацию больших объемов в более надежном виде.



- библиотеки и архивы - поиск нужной информации.

Что такое информационная система?

- Информационные системы – системы, предназначенные для хранения и обработки **больших объемов информации об объектах, процессах, явлениях реального мира.**
- Библиотека – информационная система

Определение ИС

- **Информационные системы (ИС)** – взаимосвязанная совокупность средств, методов и персонала, используемых для хранения, пополнения, обработки и выдачи необходимой информации в интересах достижения поставленной цели.

Структурирование данных

- Данные представленные в крупной ИС должны быть определенным образом структурированы.
- В противном случае невозможно будет обеспечить быстрый поиск и представления информации в доступном виде.

Структурирование данных

Неструктурированные данные

- 1996, «Практическое руководство по SQL», 320 с., 5-88782-132-9; «Диалектика», 97 год, ВНУ, Введение в системы баз данных, 704 стр., 89 руб., тысяча девятьсот девяносто седьмой год ;96 р., 130 рублей, 996-506-094-5, Питер пресс, Эффективная работа с СУБД.

Структурирование данных

Структурированные данные

| Название | Издательство | Страниц | Издана | Цена |
|---------------------------------|--------------|---------|--------|------|
| Практическое руководство по SQL | Диалектика | 320 | 1997 | 130 |
| Введение в системы баз данных | Диалектика | 980 | 1998 | 96 |
| Эффективная работа с СУБД | Питер пресс | 704 | 1997 | 89 |

Структурирование данных

- Структура несет **новое содержание, новое качество, новую информацию.**
- Одним из самых ярких примеров удачного структурирования информации является таблица Д. И. Менделеева.
- Она показывает, как много полезной информации можно предоставлять хорошо организованная структура данных.

БД и СУБД

- **База данных** (БД) – это совокупность структурированных **данных**, относящаяся к определенной предметной области и отражающая **свойства объектов** и их **отношения**.
- **Система управления базами данных** (СУБД) – это **комплекс программных** и языковых **средств**, необходимых для создания БД, поддержания их в актуальном состоянии и организации поиска в них необходимой информации.

- БД – это файл данных



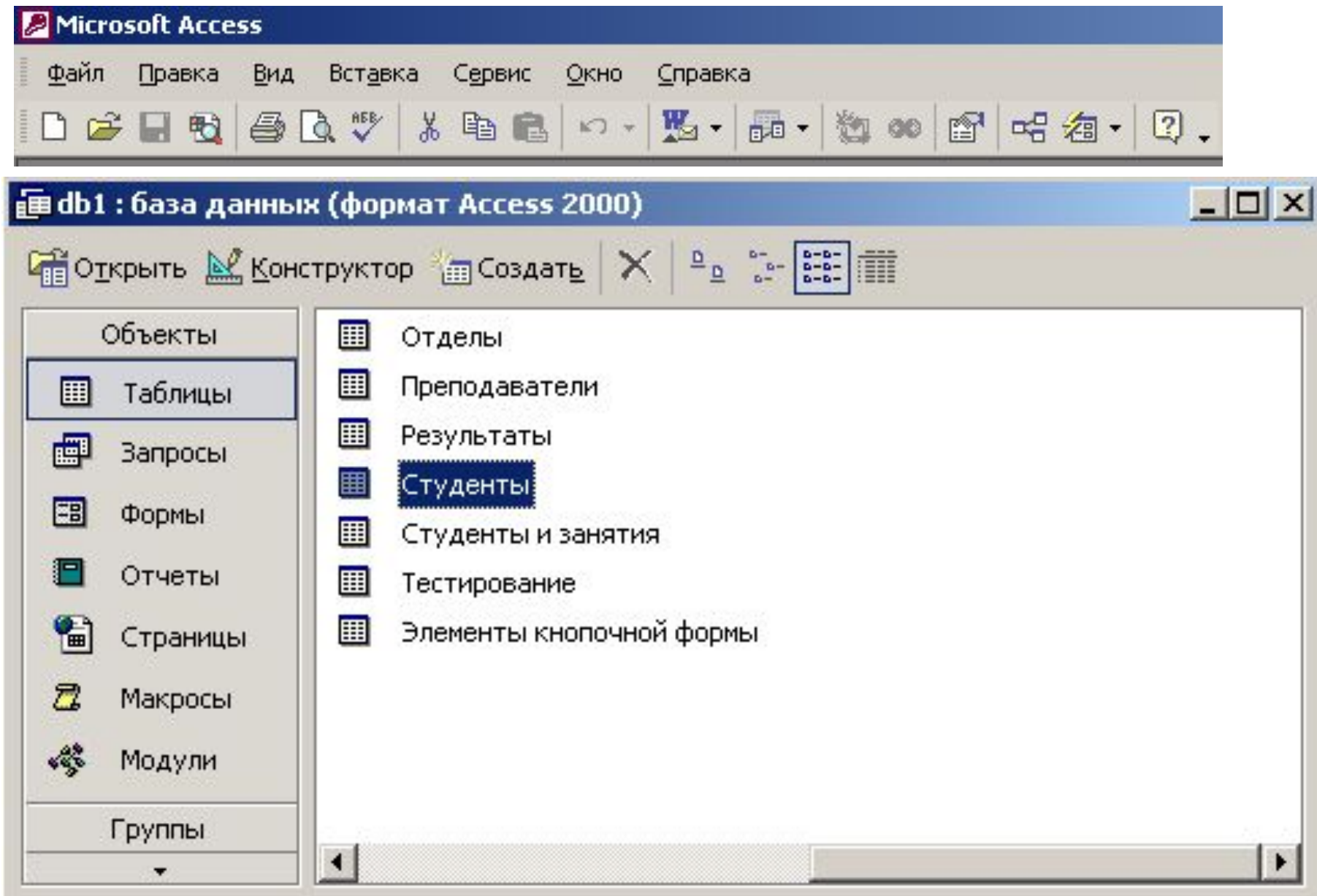
db1



temp

- СУБД – это программа, позволяющая управлять базами данных
 - Microsoft Access
 - Paradox
 - Oracle

Microsoft Access



Модели данных

- В любой БД информация представляется в виде определенной структуры:
 - иерархическая
 - табличная
 - сетевая

Иерархическая модель данных

Группа (Код группы, название, дата создания, страна)

7 The Beatles 1961 Англия

Альбом (Код альбома, название, год выпуска, фирма)

2
Love Songs
1977
Capitol

12
Help!
1965
Parlaphone

42
With The Beatles
1963
Parlaphone

Песня (Код песни, Название, № песни в альбоме, время звучания)

12
Yesterday
2,07

6
I Need You
2,31

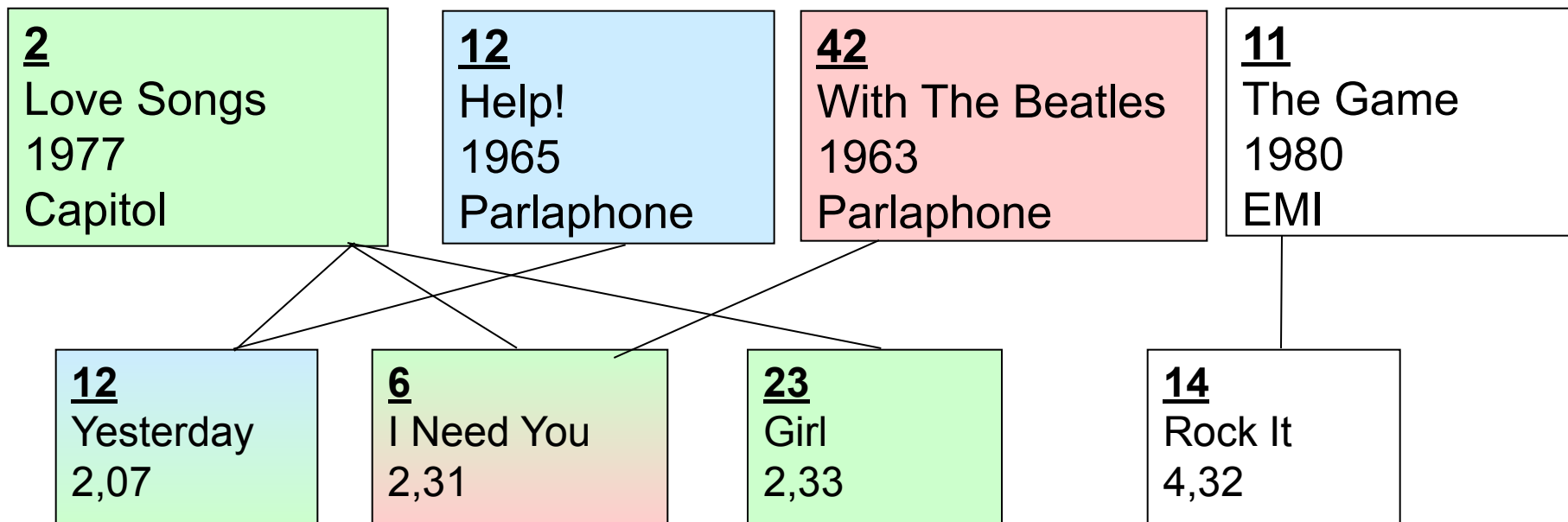
23
Girl
2,33

Иерархическая модель данных

- Иерархическая модель представляет собой совокупность элементов, расположенных в порядке их подчинения от общего к частному и образующих ориентированный граф (перевернутое дерево).
- К каждой записи существует только один иерархический путь от корневой записи

Сетевая модель данных

Альбом (Код альбома, название, год выпуска, фирма)



Песня (Код песни, название, время звучания)

Реляционная модель данных

Группа

| <u>Код группы</u> | Название | Дата создания | Страна |
|-------------------|-------------|---------------|--------|
| 7 | The Beatles | 1961 | Англия |
| 3 | Queen | 1973 | Англия |
| 5 | U2 | 1981 | |

Альбом

| <u>Код альбома</u> | Название | Год выпуска | Фирма | <i>Код Группы</i> |
|--------------------|------------------|-------------|------------|-------------------|
| 2 | Love Songs | 1977 | Capitol | 7 |
| 12 | Help! | 1965 | Parlaphone | 7 |
| 42 | With The Beatles | 1963 | Parlaphone | 7 |
| | | | | |
| 11 | The Game | 1980 | EMI | 3 |

Эта модель характеризуется простотой структуры данных, удобным для пользователя **табличным представлением**

Реляционная модель данных

- Реляционная модель построена на основе таблицы или связанных между собой таблиц.
- Каждая таблица содержит однородную информацию об объектах, процессах или явлениях некоторой предметной области.

Объект – Группа

Свойства - Код группы, Название, Дата создания, Страна

Атрибут – некий показатель, характеризующий объект

| <u>Код группы</u> | Название | Дата создания | Страна |
|-------------------|-------------|---------------|--------|
| 7 | The Beatles | 1961 | Англия |
| 3 | Queen | 1973 | Англия |
| 5 | U2 | 1981 | |

Каждая строка таблицы есть совокупность значений атрибутов, относящихся к конкретному объекту.

В терминах реляционных баз данных **строку** называют **записью** (кортеж), а **столбец** **полем**

Реляционные таблицы обладают определенными свойствами

1. каждый элемент таблицы – это один элемент данных

На пересечении строки и столбца находится **атомарное** (неделимое, не имеющее внутренней структуры) значение.

Нарушено свойство атомарности!

| <u>Код группы</u> | Название | Дата создания | Альбом | Страна |
|-------------------|-------------|---------------|-------------------|--------|
| 7 | The Beatles | 1961 | Help!, Love Songs | Англия |
| 3 | Queen | 1973 | The Game, Jazz | Англия |

Свойства реляционных таблиц

2. все столбцы однородные
3. каждое поле таблицы имеет уникальное ИМЯ
4. порядок строк и столбцов может быть произвольным

| <u>Код группы</u> | Название | Дата создания | Страна |
|-------------------|-------------|---------------|--------|
| 7 | The Beatles | 1961 | Англия |
| 3 | Queen | 1973 | Англия |
| 5 | U2 | 1981 | |

Свойства реляционных таблиц

5. Отсутствуют одинаковые записи:

То есть в таблице не должно быть одинаковых строк данных.

- Следствием этого является положение, что в любой таблице должно быть **поле** или **набор полей**, **значения** данных, в которых **отличаются для каждой записи** этой таблицы
- **Потенциальный ключ** – это поле или набор полей, которые **однозначно определяют** соответствующую запись

Потенциальные ключи

| № лично го дела | Фамилия | Имя | Отчество | № зачетки | Дата рождения | Ул. | Дом | Кв. |
|--------------------|---------|-------|-----------|-----------|---------------|-------|-----|-----|
| 128 | Петров | Иван | Иванович | 970482 | 13.07.81 | Мира | 12а | 10 |
| 176 | Бойко | Петр | Андреевич | 970467 | 01.01.81 | Щорса | 14 | 4 |
| 231 | Ким | Ольга | Петровна | 980134 | 12.01.81 | Лазо | 12 | 11 |
| 234 | Петров | Петр | Иванович | 980176 | 12.01.82 | Мира | 12а | 10 |

Потенциальные ключи:

№ личного дела,

№ зачетки

Фамилия + Имя + Отчество + Дата рождения +

Улица + Дом + Квартира

Потенциальные ключи

- Если потенциальных ключей несколько, то один из них выбирается за основной (**первичный ключ**).

| № лично го дела | Фамилия | Имя | Отчество | <u>№ зачетки</u> | Дата рождения | Ул. | Дом | Кв. |
|--------------------|---------|-------|-----------|------------------|---------------|-------|-----|-----|
| 128 | Петров | Иван | Иванович | <u>970482</u> | 13.07.81 | Мира | 12а | 10 |
| 176 | Бойко | Петр | Андреевич | <u>970467</u> | 01.01.81 | Щорса | 14 | 4 |
| 231 | Ким | Ольга | Петровна | <u>980134</u> | 12.01.81 | Лазо | 12 | 11 |
| 234 | Петров | Петр | Иванович | <u>980176</u> | 12.01.82 | Мира | 12а | 10 |

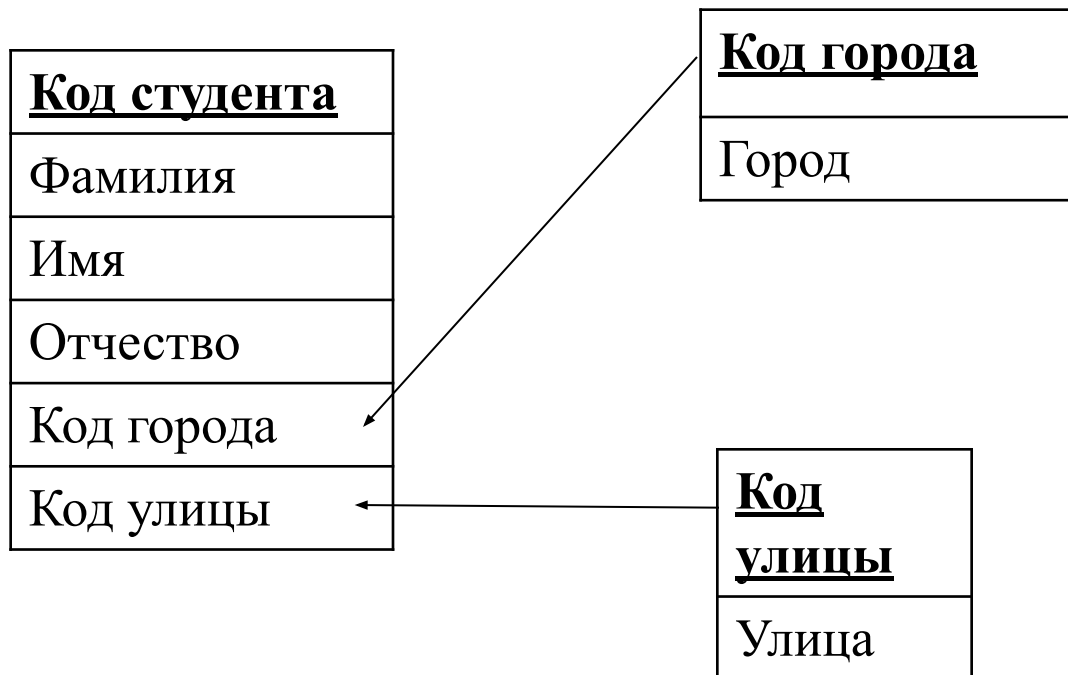
Потенциальный ключ

- По первичному ключу выполняется автоматическое упорядочивание записи, что ускоряет поиск информации.
- Первичные ключи в СУБД обеспечивают основной механизм адресации на уровне записи.
- В MS Access – ключевое поле.

| | Имя поля | Тип данных |
|---|-----------------------|------------|
|  | КодСтудента | Счетчик |
| | Имя | Текстовый |
| | Фамилия | Текстовый |
| | Адрес | Текстовый |
| | Город | Текстовый |
| | ОбластьКрайРеспублика | Текстовый |
| | ПочтовыйИндекс | Текстовый |

Многотабличные БД

- Чаще всего структура БД представлена в виде нескольких таблиц, связанных между собой через общие атрибуты.



Необходимость использования в БД нескольких таблиц

| Название группы | Страна | Дата создания группы | Название альбома | Год издания | Фирма |
|-----------------|--------|----------------------|-------------------|-------------|------------|
| The Beatles | Англия | 1963 | With The Beatles | 1963 | Parlophone |
| The Beatles | Англия | 1963 | Please, please me | 1963 | Parlophone |
| The Beatles | Англия | 1963 | Rubber Soul | 1965 | Parlophone |

При внесении в нее данных об альбомах определенной группы, каждый раз приходится **дублировать информацию** первых трех полей таблицы

Представленную выше таблицу разобьем на две: Альбомы и Группы

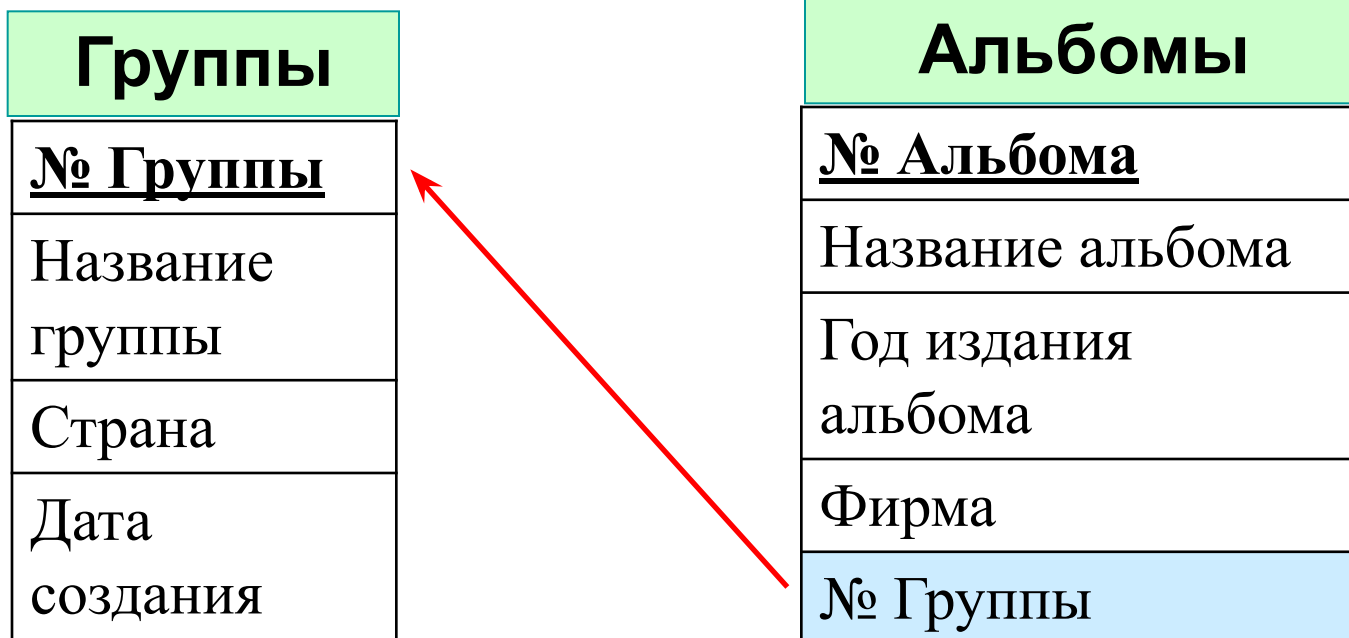
Группы

| <u>Номер группы</u> | Название группы | Страна | Дата создания группы |
|----------------------------|------------------------|---------------|-----------------------------|
| 1 | The Beatles | Англия | 1963 |
| 2 | Led Zeppelin | Англия | 1970 |
| 3 | U2 | Ирландия | 1980 |

Альбомы

| <u>Номер альбома</u> | Название альбома | Год издания альбома | Фирма |
|-----------------------------|-------------------------|----------------------------|--------------|
| 25 | With The Beatles | 1963 | Parlophone |
| 26 | Pleas, pleas me | 1963 | Parlophone |
| 27 | Rubber Soul | 1965 | Parlophone |

Как связать таблицы между собой?



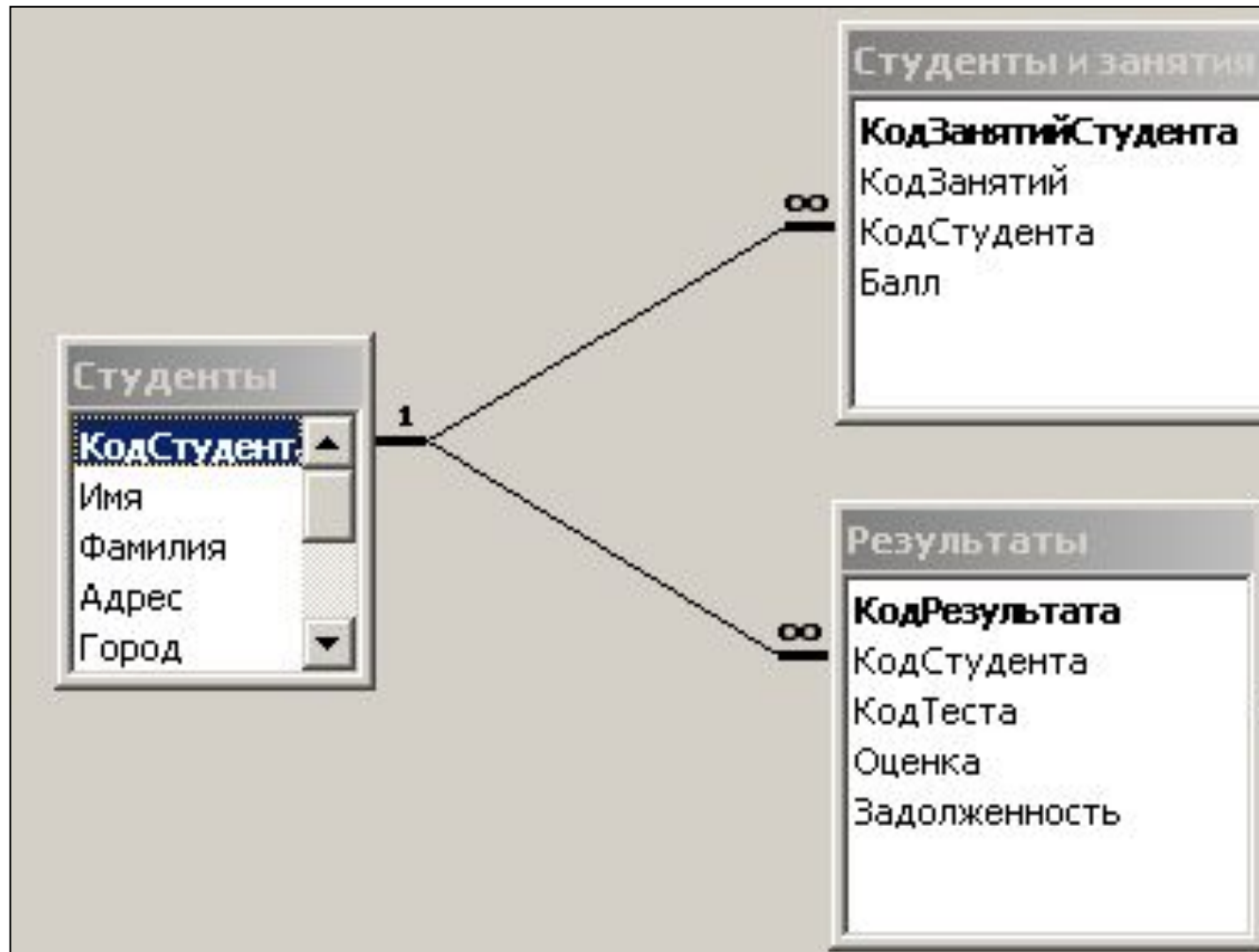
Внешний ключ

- Эти таблицы связаны между собой через общие атрибуты (№ группы).
- Для обеспечения связи между таблицами используются *внешние ключи*.
- Значения внешнего ключа формируются на основе значений соответствующего ему первичного ключа.

Типы связей между таблицами

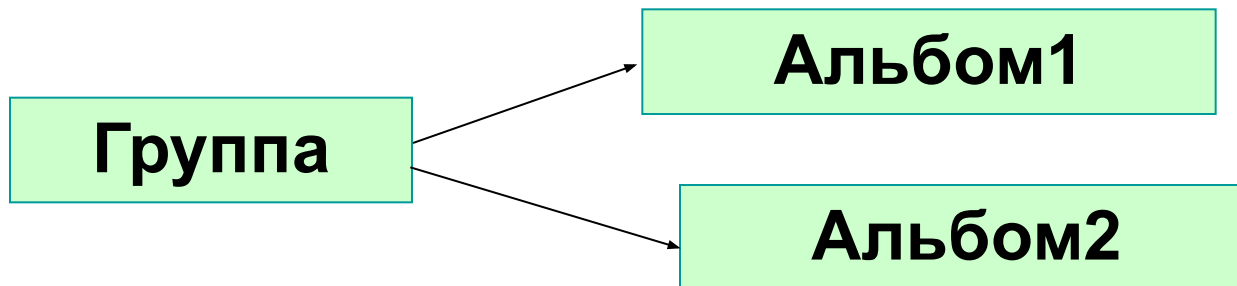
- Связи между таблицами очень важны, поскольку они указывают, как находить, размещать и использовать информацию из полей двух или более таблиц.
- Кроме того, связи отражают правила отношения между объектами, представленными в различных таблицах.
- Существует три типа связей:
 - один-к-одному, 1-1
 - один-ко-многим, 1-∞
 - многие-ко-многим. ∞-∞

Связи в MS Access



Связь *один-ко-многим* (1 – M)

- Группы 1 - M Альбомы
- Этот тип связи соответствует отношению между таблицами Группа и Альбомы.
- У каждой группы может быть несколько альбомов, но любой альбом может быть выпущен одной определенной группой.



- Таблица со стороны отношения 1 называется *главной*, таблица же со стороны многие – *подчиненной*.

СВЯЗЬ *один-ко-многим* (1 – M)

Группы

| <u>Номер группы</u> | Название группы | Страна | Дата создания |
|---------------------|-----------------|----------|---------------|
| <u>2</u> | Led Zeppelin | Англия | 1970 |
| <u>3</u> | U2 | Ирландия | 1980 |
| <u>6</u> | The Beatles | Англия | 1963 |

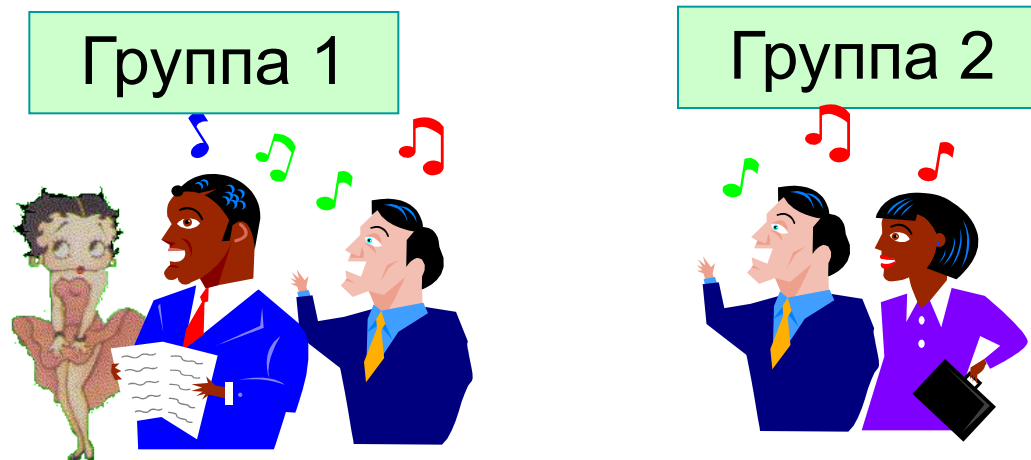
Альбомы

| <u>Номер альбома</u> | Номер группы | Название альбома | Год издания | Фирма |
|----------------------|--------------|------------------|-------------|------------|
| <u>25</u> | 6 | With The Beatles | 1963 | Parlophone |
| <u>4</u> | 6 | Pleas, pleas me | 1963 | Parlophone |
| <u>27</u> | 6 | Rubber Soul | 1965 | Parlophone |

Значения внешнего ключа формируются на основе значений соответствующего ему первичного ключа.

Связь *многие-ко-многим* (M – M)

- группа M – M музыкант
- Например, в группе может играть несколько музыкантов, а любой музыкант может играть в нескольких музыкальных группах.



СВЯЗЬ *многие-ко-многим* (M – M)

Таблица 3

| Номер группы | Номер музыканта |
|--------------|-----------------|
| 6 | 25 |
| 2 | 25 |
| 3 | 26 |
| 3 | 27 |
| 6 | 27 |

| <u>Номер группы</u> | Название группы | Страна | Дата создания |
|---------------------|-----------------|----------|---------------|
| <u>2</u> | Led Zeppelin | Англия | 1970 |
| <u>3</u> | U2 | Ирландия | 1980 |
| <u>6</u> | The Beatles | Англия | 1963 |

Группы

| <u>Номер музыканта</u> | Фамилия | Имя |
|------------------------|---------|-----|
| <u>25</u> | Леннон | А |
| <u>26</u> | Петров | Б |
| <u>27</u> | Смит | С |

Музыканты

Целостность данных

- Целостность данных - наличие некоторых правил призванных обеспечить непротиворечивость информации хранимой в БД.
- Общие правила целостности, связаны с понятиями **первичных** и **внешних** ключей.

Целостность данных

Правило 1. *Правило целостности объекта.*

- Ни один элемент первичного ключа не может содержать пустого значения.

Сотрудники

| № банковского счета | Фамилия | Имя | Должность |
|---------------------|----------|--------|-------------|
| 241234 | Проворов | Илья | директор |
| | Калугин | Степан | программист |
| 234322 | Гордеева | Анна | бухгалтер |
| | Павленко | Ольга | секретарь |

Целостность данных

Правило 2. *Правило ссылочной целостности* .

- Текущее значение внешнего ключа должно совпадать со значением соответствующего ему первичного ключа или являться пустым значением

Правило ссылочной целостности

| № отдела | Название | Бюджет |
|----------|---------------------------|-----------|
| 11 | Бухгалтерия | 230000 р. |
| 2 | Технического обслуживания | 221000 р. |
| 4 | Информационный | 332000 р. |



| Код сотрудника | Фамилия | Имя | Должность | № отдела |
|----------------|----------|--------|-------------|----------|
| 21 | Калугин | Степан | Программист | 4 |
| 124 | Гордеева | Анна | Бухгалтер | 11 |
| 121 | Павленко | Ольга | Секретарь | 11 |
| 12 | Иваненко | Олег | Консультант | |

Реляционные операторы и язык SQL

- SQL не является языком программирования в традиционном представлении.
- На нем пишутся не программы, а **запросы** к базе данных.
- Поэтому SQL - декларативный язык. Это означает, что с его помощью можно сформулировать, что необходимо получить, а не то, как это следует сделать.

Запрос на языке SQL состоит из одного или нескольких операторов, следующих один за другим и разделенных точкой с запятой.

INSERT

INTO Группы (КодГруппы,
НазваниеГруппы, ДатаСоздания,
КодСтраны, ФотоГруппы, Описание)

VALUES (9, "Мумий троль", "12.01.98", 16,
Null, Null);

Операторы языка SQL

- **Язык определения данных (ЯОД)**
 - создание таблиц,
 - определение типов полей,
 - удаление таблиц
- **Язык манипулирования данными (ЯМД).**
 - выбор данных,
 - вставка и удаление данных,
 - обновление значений в таблицах

Язык определения данных

- **CREATE TABLE** Группы (Номер_группы integer, Название_группы text (20), Страна text (50), Дата_создания date, primary KEY (Номер_группы));

| Номер_группы | Название_группы | Страна | Дата_создания |
|--------------|-----------------|--------|---------------|
| | | | |

В результате выполнения этого оператора будет создана таблица **Группы** с первичным ключом Номер_группы, в которой значения поля:

Номер_группы могут быть целыми числами;

Название_группы - строка размером не более 20 символов;

Название_группы - строка размером не более 50 символов;

Дата_создания данные типа дата.

Язык определения данных

- DROP TABLE Группы;
- Результатом выполнения этого оператора будет удаление таблицы Группы вместе со всеми данными, если они есть.

Язык манипулирования данными

- INSERT
- SELECT
- DELETE
- UPDATE

Справка