

# Базы данных и системы управления базами данных(СУБД)

---

Хранение, поиск и сортировка  
информации

**База данных(БД) — организованная совокупность данных, предназначенная для длительного хранения во внешней памяти ЭВМ и постоянного применения.**

---

**БД** позволяет упорядоченно хранить данные о большом количестве однотипных объектов, обладающих одинаковым набором свойств

- База данных книжного фонда библиотеки;
- База данных кадрового состава учреждения;
- База данных законодательных актов в области уголовного права;
- База данных современных песен.

Для хранения БД может использоваться как **один компьютер**, так и **множество взаимосвязанных компьютеров**.

Если база данных хранится на одном компьютере, то такая БД называется **централизованной**. Если различные части одной базы данных хранятся на множестве компьютеров, объединенных между собой сетью, то такая БД называется **распределенной базой данных**.

---

# База данных

---

Информация в базах данных может быть организована по-разному. Чаще всего используется табличный способ.

**База данных** может состоять:

- из одной таблицы (однотабличная БД)
- из множества взаимосвязанных таблиц (многотабличная БД)

Множество взаимосвязанных таблиц, каждая из которых содержит информацию об объектах определенного типа называется **реляционной базой данных**.

---

# Табличная форма представления баз данных

---

## База данных «Записная книжка»

<b>№</b>	<b>Фамилия, имя</b>	<b>Дата рождения</b>	<b>Телефон</b>	<b>Домашний адрес</b>
<b>1</b>	<b>Лебедева Мария</b>	<b>16.06.93</b>	<b>98309</b>	<b>ул. Заозёрная д.1</b>
<b>2</b>	<b>Резвова Надежда</b>	<b>18.03.92</b>	<b>21279</b>	<b>ул. Садовая д.2</b>

**Поля** — это различные характеристики (иногда говорят — атрибуты) объекта. Значения полей в одной строчке относятся к одному объекту. Разные поля отличаются именами.

Одна **запись** содержит информацию об одном объекте той реальной системы, модель которой представлена в таблице.

---

Структурными составляющими таблицы являются **записи** и **поля**

The diagram illustrates the structure of a table. The word "поля" (fields) is positioned above the table, with five red arrows pointing to the column headers: "№", "Фамилия, имя", "Дата рождения", "Телефон", and "Домашний адрес". The word "записи" (records) is positioned to the left of the table, with two red arrows pointing to the first and second rows of the table.

№	Фамилия, имя	Дата рождения	Телефон	Домашний адрес
1	Лебедева Мария	16.06.93	98309	ул.Заозёрная д.1
2	Резвова Надежда	18.03.92	21279	ул.Садовая д.2

# Поле БД

---

Каждое поле имеет **ИМЯ** и может хранить данные определённого **типа**:

- текст,
  - число,
  - дата/время,
  - счётчик,
  - и т.д.
-

# Текст

<b>№</b>	<b>Фамилия, имя</b>	<b>Дата рождения</b>	<b>Телефон</b>	<b>Домашний адрес</b>
<b>1</b>	<b>Лебедева Мария</b>	<b>16.06.93</b>	<b>98309</b>	<b>ул.Заозёрная д.1</b>
<b>2</b>	<b>Резцова Надежда</b>	<b>18.03.92</b>	<b>21279</b>	<b>ул. Садовая д.2</b>



# Число

<b>№</b>	<b>Фамилия, имя</b>	<b>Дата рождени я</b>	<b>Телефон</b>	<b>Домашний адрес</b>
<b>1</b>	<b>Лебедева Мария</b>	<b>16.06.93</b>	<b>98309</b>	<b>ул.Заозёрная д.1</b>
<b>2</b>	<b>Резвова Надежда</b>	<b>18.03.92</b>	<b>21279</b>	<b>ул. Садовая д.2</b>





# Дата/Время

<b>№</b>	<b>Фамилия, имя</b>	<b>Дата рождения</b>	<b>Телефон</b>	<b>Домашний адрес</b>
<b>1</b>	<b>Лебедева Мария</b>	<b>16.06.93</b>	<b>98309</b>	<b>ул.Заозёрная д.1</b>
<b>2</b>	<b>Резцова Надежда</b>	<b>18.03.92</b>	<b>21279</b>	<b>ул. Садовая д.2</b>



# Счётчик

<b>№</b>	<b>Фамилия, имя</b>	<b>Дата рождения</b>	<b>Телефон</b>	<b>Домашний адрес</b>
<b>1</b>	<b>Лебедева Мария</b>	<b>16.06.93</b>	<b>98309</b>	<b>ул.Заозёрная д.1</b>
<b>2</b>	<b>Резвова Надежда</b>	<b>18.03.92</b>	<b>21279</b>	<b>ул. Садовая д.2</b>



# Строчное представление структуры таблицы

База данных «Записная книжка»

---

№	Фамилия, имя	Дата рождения	Телефон	Домашний адрес
1	Лебедева Мария	16.06.93	98309	ул. Заозёрная д.1
2	Резцова Надежда	18.03.92	21279	ул. Садовая д.2

**ИМЯ\_ТАБЛИЦЫ (ИМЯ\_ПОЛЯ1, ИМЯ\_ПОЛЯ2, ... , ИМЯ\_ПОЛЯ\_N)**

**ЗАПИСНАЯ КНИЖКА (НОМЕР, ФАМИЛИЯ\_ИМЯ, ДАТА\_РОЖДЕНИЯ,  
ТЕЛЕФОН, ДОМАШНИЙ\_АДРЕС)**

---

# Ключи

---

**ЗАПИСНАЯ КНИЖКА (НОМЕР, ФАМИЛИЯ\_ИМЯ,  
ДАТА\_РОЖДЕНИЯ, ТЕЛЕФОН, ДОМАШНИЙ\_АДРЕС)**

Поле НОМЕР подчеркнуто, т.к. оно является **ГЛАВНЫМ КЛЮЧОМ**.

**Ключевое поле** – поле или совокупность полей (составной ключ), однозначно определяющих запись.

---

# Простой ключ

---

## Простой ключ




Номер	Автор	Название	Год	Полка
001	Беляев А.Р.	Звезда КЭЦ	1990	3
002	Олеша Ю.К.	Избранное	1987	5
003	Беляев А.Р.	Избранное	1994	1

В БД «Домашняя библиотека» у разных книг могут совпадать значения полей,  
но инвентарный номер у каждой книги свой

---

# Составной ключ

## Составной ключ



Город	№ школы	Директор	Адрес	Телефон
Крюков	1	Иванов А.П.	Пушкина, 5	12-35
Шадринск	1	Строев С.С.	Лесная, 14	4-33-11
Шадринск	2	Иванов А.П.	Мира, 34	4-23-24

В БД «Школы области» у разных записей  
одновременно не могут совпасть  
только сочетание двух полей:  
город и номер школы (это составной ключ)

# Связи между таблицами

---

**Реляционная БД** представляет собой множество взаимосвязанных таблиц, каждая из которых содержит информацию об объектах определенного типа. То есть между таблицами, хранилищами информации, установлены какие-то ассоциации, связи или отношения. В зависимости от типа информации, содержащейся в таблицах, между таблицами БД могут существовать три основных типа связей:.

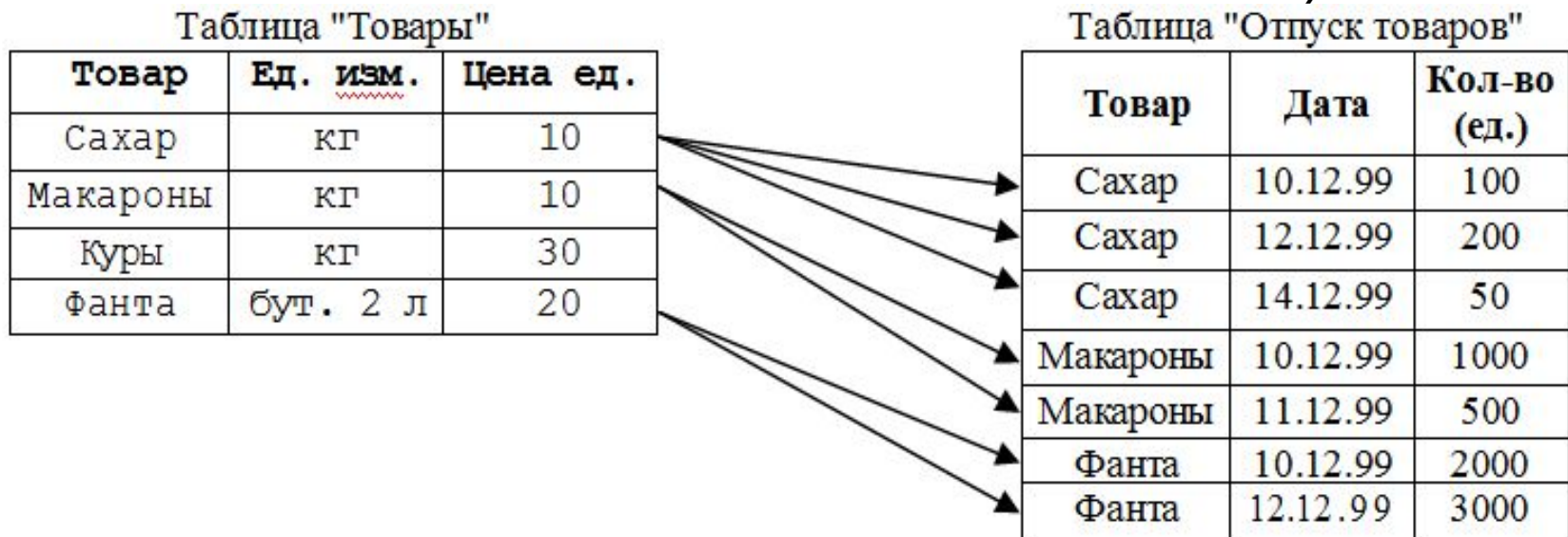
**ОДИН-КО-МНОГИМ,  
ОДИН-К-ОДНОМУ,  
МНОГИЕ-КО-МНОГИМ**

---

# Отношение один-ко-многим

Отношение *один-ко-многим* устанавливается между таблицами в том случае, когда одной записи в одной таблице может соответствовать любое количество записей в другой таблице.

Как видно из рисунка, одной записи из таблицы *Товары* может соответствовать несколько записей в таблице *Отпуск*



Связь *один-ко-многим* позволяет моделировать иерархические структуры данных.



# Отношение один-к-одному

**Отношение *один-к-одному*** имеет место, когда одной записи в родительской таблице соответствует не более одной записи в дочерней таблице. Для примера рассмотрим отношение между таблицами *Сотрудники* и *Информация о сотрудниках*.

Таблица «Сотрудники»

№	ФИО	Должность	Отдел
1	Иванов И.И.	инженер	10
2	Петров П.П.	бухгалтер	20
3	Васин В.В.	прораб	10
...	...	...	...

Таблица «Информация о сотрудниках»

№	Год рожд.	Кол-во детей	...
1	1950	3	...
2	1952	1	...
3	1960	2	...
...	...	...	...

**Отношение *один-к-одному*** используют, если не хотят, чтобы таблица БД "распухла" от второстепенной информации.

# Отношение многие-ко-многим

В случае **отношения многие-ко-многим** одной записи в родительской таблице может соответствовать любое количество записей в дочерней таблице и наоборот. Рассмотрим таблицы *Преподаватели* и *Учебные группы и дисциплины*, состоящие в отношении **многие-ко-многим**. Каждой учебной группе соответствует несколько преподавателей. Каждый преподаватель может вести, во-первых, несколько разных предметов и, во-вторых, преподавать в

Таблица "Учебные группы и дисциплины"

Группа	Предмет	№ преподавателя
ПС-1	Программирование	10
ТИ-1	Программирование	10
ПС-1	Теория систем	12
РТ-2	Философия	62
ПС-1	Социология	62
...	...	...

Таблица "Преподаватели"

№ преподавателя	ФИО	Кафедра
10	Краснов Ю.Б.	ТИ
12	Володин В.Н.	ТИ
62	Булгаков В.М.	РИО
78	Горин Л.С.	ТИ
85	<u>Подушкин М.А.</u>	ЭИ
...	...	...



БД всегда можно перестроить так, чтобы связь многие-ко-многим была заменена двумя связями типа один-ко-многим путем создания промежуточной таблицы.

Система управления базами данных (СУБД) - приложение, позволяющее создавать базы данных и осуществлять в них сортировку и поиск данных

---

Любая СУБД позволяет выполнять четыре простейшие операции с данными:

- добавлять в таблицу одну или несколько записей;
- удалять из таблицы одну или несколько записей;
- обновлять значения некоторых полей в одной или нескольких записях;
- находить одну или несколько записей, удовлетворяющих заданному условию.

Мы будем учиться создавать БД с помощью  
СУБД **MICROSOFT ACCESS**

---