

Программное обеспечение информационных технологий

Офисное программное обеспечение **СУБД**

1. Модели данных
2. Реляционные базы данных
3. Реляционные СУБД



Модели данных

Модели данных различаются по способу установления связей между данными.



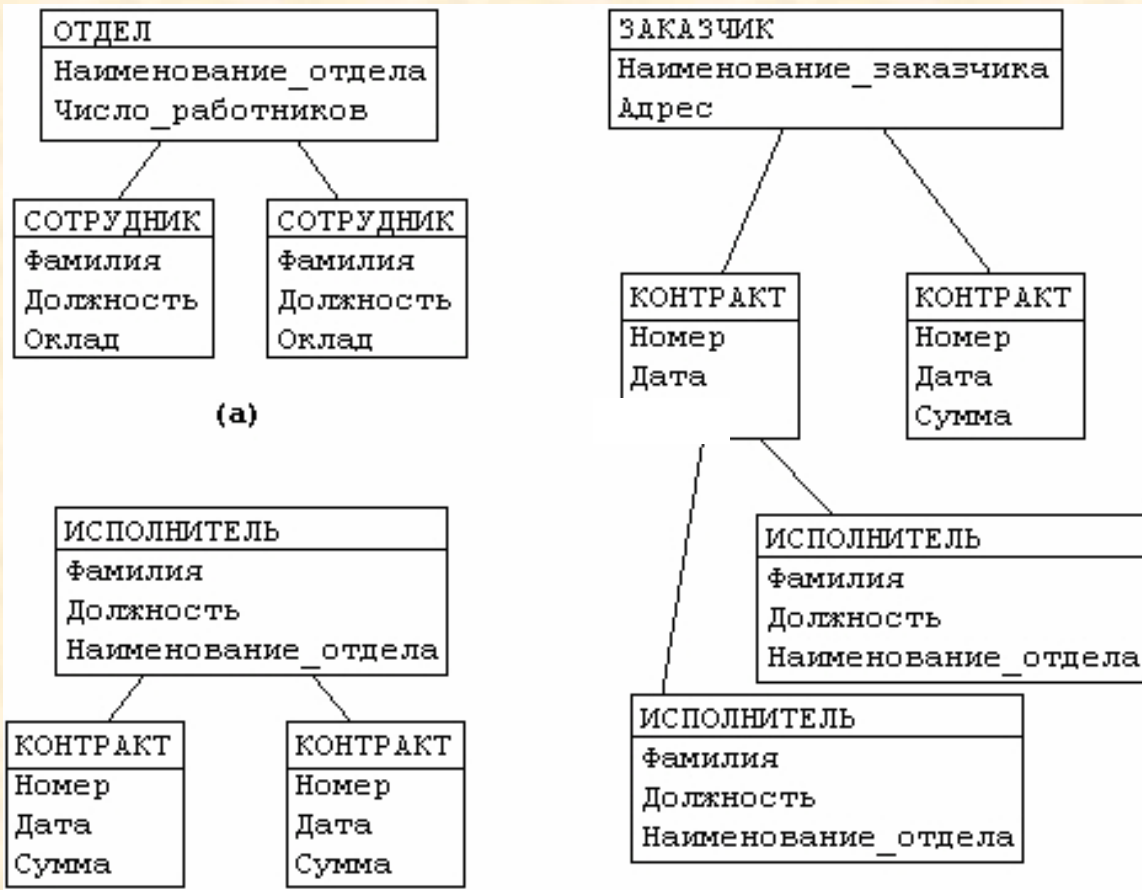
Реляционная модель является простейшей формой представления данных в виде **таблицы**. Скорость работы зависит от размера базы данных.

В **иерархической** модели связи между данными могут быть отображены в виде **дерева-графа**, где возможны только односторонние связи от старших вершин к младшим.

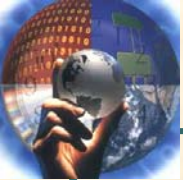
В **сетевой** модели теоретически возможны связи "всех со всеми".



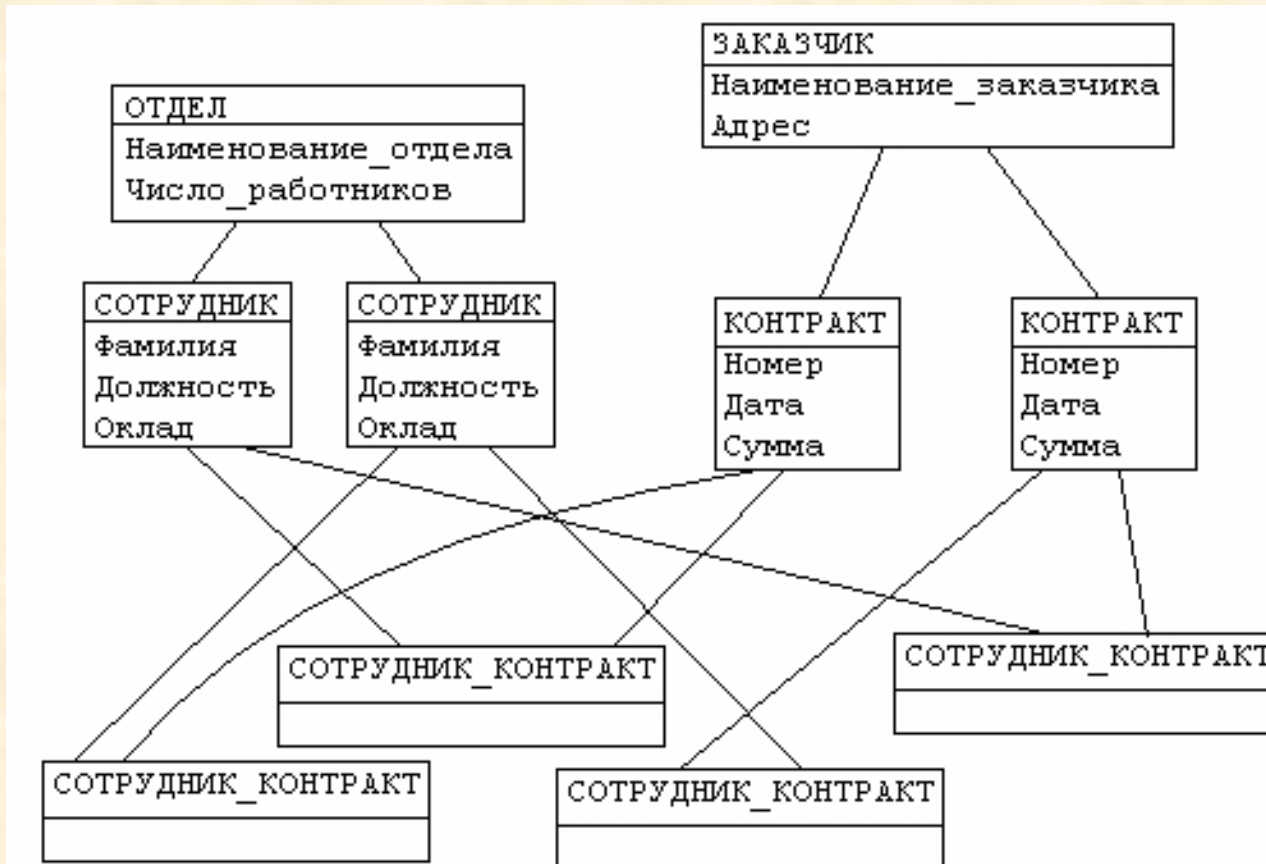
Пример иерархической модели данных



Иерархическая модель реализует отношение между исходной и дочерней записью по схеме 1:N, то есть одной родительской записи может соответствовать любое число дочерних.



Пример сетевой модели данных



Сетевая модель данных состоит из множества записей, которые могут быть **владельцами** или **членами** групповых отношений. Связь между записью-владельцем и записью-членом имеет вид 1:N.

В сетевой модели запись может быть членом более чем одного группового отношения.



1 Модели данных

Пример реляционной модели данных

Реляционная модель представляет базу данных в виде совокупности упорядоченных нормализованных отношений.

Отношение называется **нормализованным** или приведенным к первой нормальной форме, если все его атрибуты простые (неделимые).

а) Отношение ГРУППА			
Индекс ИГ	Название группы ИГ	Количество ответов КОЛ	Проходной балл ПБАЛЛ
1	A1	16	4,3
2	A2	28	4,0
3	A3	18	4,3
б) Отношение СТУДЕНТ			
Номер зачетной книжки НЗ	ИГ	Фамилия И.О. СФИО	Год рождения ГР

**Реляционная
модель**

**Реляционная
таблица**

№ личного дела	Фамилия	Имя	Отчество	Дата рождения	Группа
16493	Сергеев	Петр	Михайлович	01.01.76	ИСТ 11
16593	Петрова	Анна	Владимировна	15.03.75	СК 12
16693	Анохин	Андрей	Борисович	14.04.76	ИСТ 11



Реляционные базы данных

База данных — совокупность данных, организованных по определенным правилам, предусматривающим общие принципы описания, хранения и манипулирования данными, независимо от прикладных программ.

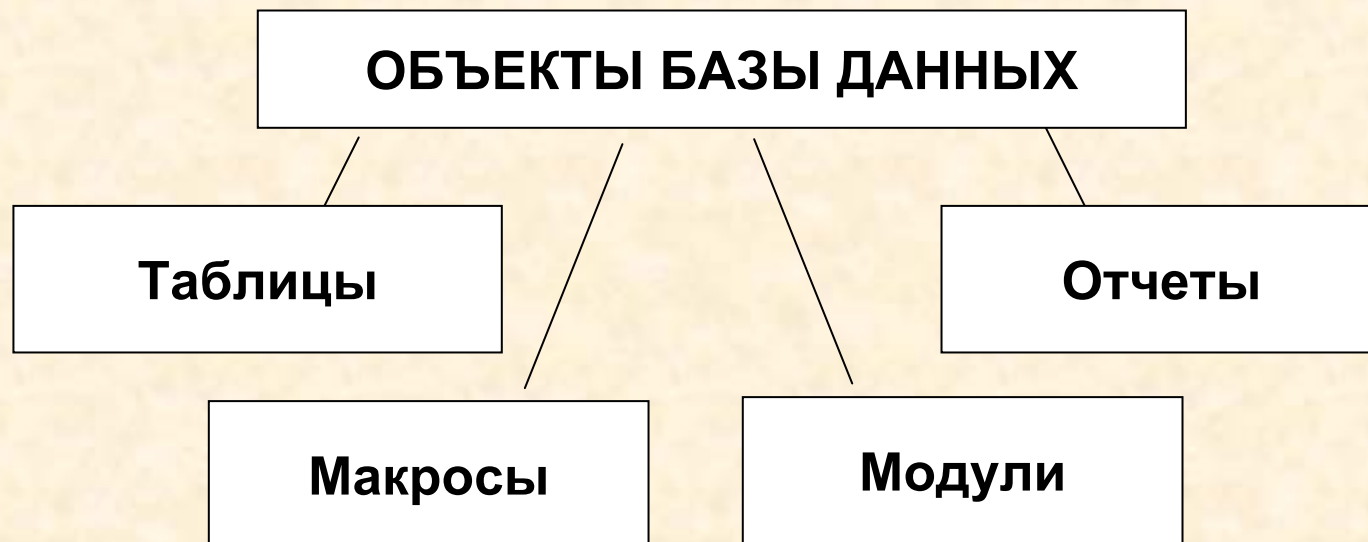
Реляционная модель ориентирована на организацию данных в виде **двумерных таблиц**.

Реляционные базы данных состоят из нескольких таблиц, связь между которыми устанавливается с помощью совпадающих полей — **ключей**.

Система управления базами данных (СУБД) представляет собой комплекс программ и языковых средств для создания, ведения и использования баз данных.



Объекты базы данных



Основными объектами базы данных являются **таблицы**.



Организация данных

Организация данных напоминает лист книги электронной таблицы. В каждом столбце таблицы находятся данные одинакового типа, а каждая строка содержит одинаковые по структуре записи.

АНКЕТЫ							
РЕГ	ФАМИЛИЯ	ИМЯ	ОТЧЕСТВО	ДАТА_РОЖ	ГОРОД	УЧ_ЗАВЕДЕ	Добавить поле
1	Васильева	Ольга	Николаевна	12.10.1985	Пермь	ПТУ №8	
2	Быков	Алексей	Ильич	24.04.1986	Астрахань	Школа №3	
3	Круглов	Борис	Макарович	18.09.1985	Самара	ПТУ №25	
4	Листьев	Сергей	Иванович	01.03.1986	Самара	Школа №48	
5	Елькин	Виктор	Алексеевич	23.05.1985	Пенза	Школа № 3	

При заполнении таблицы данными
сохранение их происходит автоматически.



Режимы работы с данными

Режим таблицы используется, если надо получить наиболее полный обзор данных или для эпизодического изменения данных.

При частом изменении данных лучше использовать **форму**.

Формы используются как средство защиты базы данных от действий неквалифицированных пользователей, повышают комфортность работы.

Любая форма строится на основе **таблицы** или **запроса**.

Поля в форме располагаются произвольно.

Логически взаимосвязанные поля объединяются в **группы**.

На основе одной таблицы могут быть построены **разные формы** для различных работников.

Содержимое формы всегда соответствует информации в таблицах и запросах, и **обновляется** при каждом открытии.



Свойства реляционных баз данных

СВОЙСТВА

каждый столбец имеет уникальное имя

все столбцы в таблице однородные

одинаковые строки в таблице отсутствуют

каждый элемент таблицы — один элемент данных

порядок следования строк и столбцов произвольный



Этапы проектирования реляционной базы данных

База данных — поименованная совокупность структурированных данных, относящихся к определенной предметной области.

- 1 Определение цели создания БД
- 2 Определение таблиц БД
- 3 Определение полей таблиц
- 4 Задание основного ключа для связи
- 5 Определение отношений между таблицами
- 6 Корректировка структуры БД
- 7 Ввод данных, создание форм, запросов, отчетов



Функции СУБД

Основные

Ввод данных

Запросы по данным

Составление отчетов
по данным и запросам

Дополнительные

Сортировка данных

Фильтрация данных

Вычисление итоговых значений

Изменение, удаление,
копирование данных

Отличия запросов от фильтров

Фильтры не позволяют в одной строке отображать данные из нескольких таблиц.

Фильтры всегда отображают все поля базовой таблицы.

Фильтры не могут быть сохранены как отдельный объект в окне базы данных.

Фильтры не позволяют вычислять суммы, средние значения, подсчитывать количество записей и находить другие итоговые значения.



Оценка производительности СУБД





Типы данных

Текстовый

Числовой

Дата/Время

Денежный

Счетчик

Имя поля	Тип данных
РЕГ_НОМ	Счетчик
ФАМИЛИЯ	Текстовый
ИМЯ	Текстовый
ОТЧЕСТВО	Текстовый
ДАТА_РОЖДЕНИЯ	Дата/время
ГОРОД	Текстовый
УЧ_ЗАВЕДЕНИЕ	Текстовый

СПИСОК ПОЛЕЙ СПИСОК ТИПОВ Свойства поля

Общие Подстановка

Размер поля Длинное целое

Новые значения Последовательные

Формат поля

Панель редактирования

Значения числовых полей по умолчанию до ввода данных равны нулю.

Для нормальной работы с БД необходима однозначная идентификация каждой записи. Это достигается автоматической нумерацией записей в поле с типом данных *Счетчик*. Возможна ручная нумерация записей в поле с типом данных *Числовой* и свойством *Индексированное поле* — ДА (совпадения не допускаются).



Свойства полей

Размер. Максимальная длина текстового или числового поля.

Формат. Форма отображения данных.

С разделителями разрядов (пробел между триадами, запятая между целой и дробной частью и 2 знака после запятой);

Основной (число отображается так, как его ввел пользователь).

Подпись. Выводится рядом с полем в форме или отчете, содержит пояснение.

Значение по умолчанию. Конкретное числовое значение, отображенное в поле.

Условие на значение. Устанавливает ограничения на вводимые значения (например, муж/жен)

Сообщение об ошибке. Появляется при вводе недопустимого значения.

Обязательное поле. Поле, подлежащее обязательному заполнению.



Организация данных

Запись — набор взаимосвязанных данных о конкретном предмете.

Поле — информационная единица записи.

Ключ (ключевое поле) — поле, которое содержит наиболее характерную для данной записи информацию.

Каждая **запись** в таблицах идентифицирует один объект.

Первичный ключ может состоять из единственного поля таблицы, значения которого уникальны для каждой записи.

Студент

Номер зач.кн.

Фамилия

Имя

Отчество

Группа

Сессия

Номер зач.кн.

Оценка 1

...

Оценка n

Если связать таблицы *Студент* и *Сессия* по совпадающему полю, то можно получить данные о результатах сдачи сессии конкретным студентом.



Ключевые поля

В реляционных базах данных отдельные таблицы связаны между собой по **ключевым полям**.

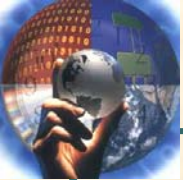
Обязательное **условие установления связи** между таблицами — наличие двух одинаковых полей.

Поля, являющиеся главными и внешними ключами, должны иметь одинаковые **имя, тип и размер**.

В родительской таблице значения главного ключа должны быть **уникальны**.

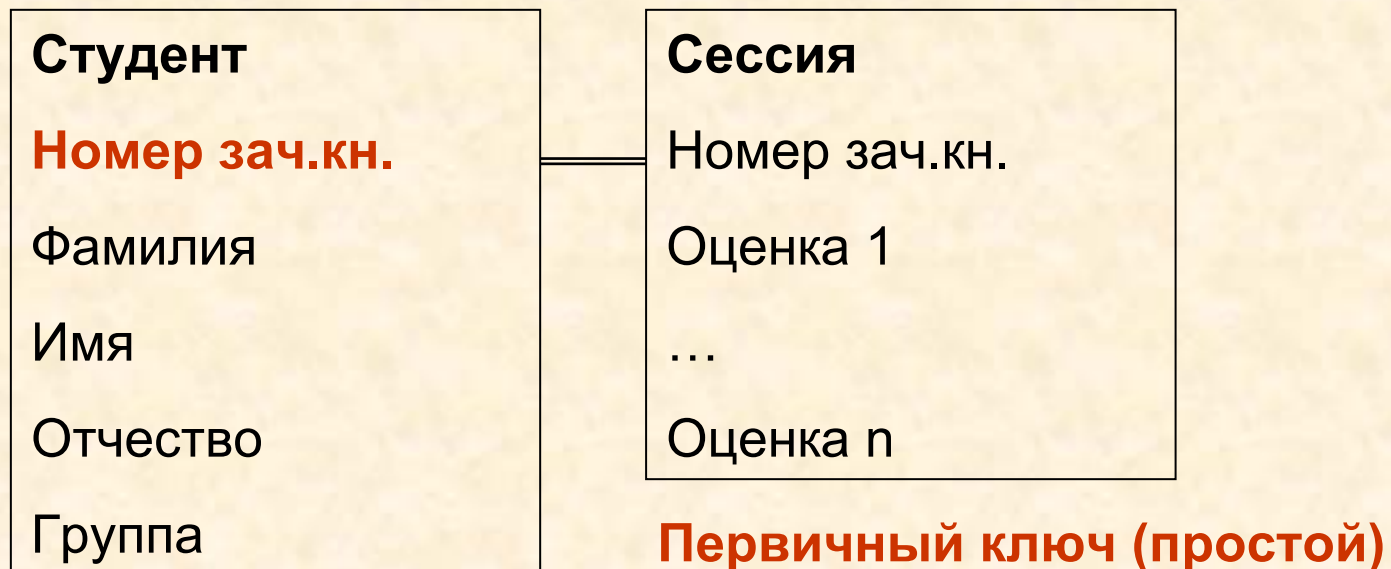
В дочерней таблице значения внешнего ключа могут **повторяться**.

Поле первичного ключа не позволяет вводить в таблицу **повторяющиеся** записи, так как содержит однозначный идентификатор для каждой записи.



Пример использования первичного ключа

Между таблицами *Студент* и *Сессия* нужно установить связь. В вузе не может быть двух студентов с одинаковыми номерами зачетных книжек. Поэтому в таблице, содержащей записи о студентах, номер зачетной книжки может быть первичным ключом. Такой первичный ключ называют **простым ключом**.





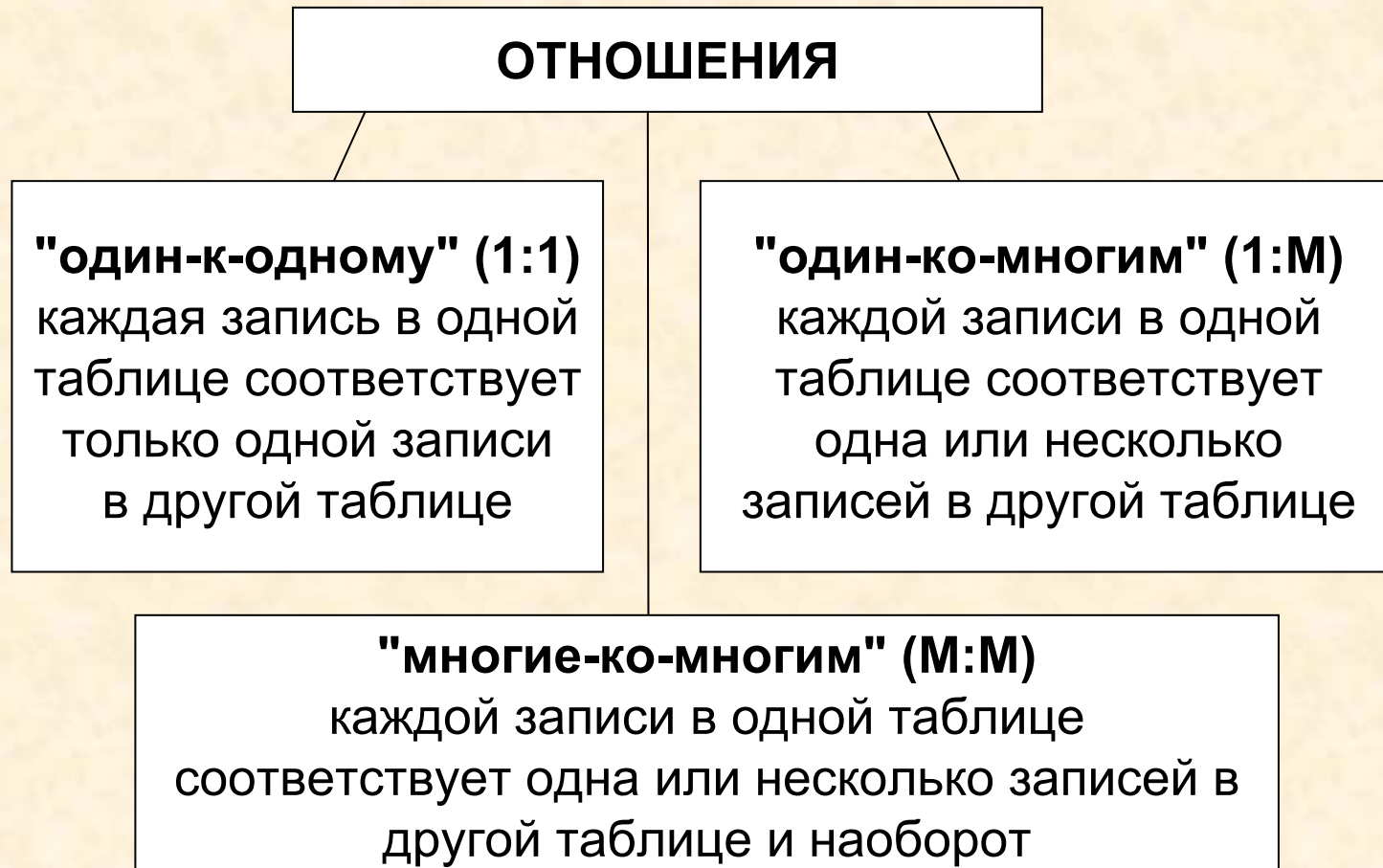
Пример использования внешнего ключа

Поле, с помощью которого осуществляется связывание таблиц, в главной (родительской) таблице называется **главным (первичным)**, а в подчиненной (дочерней) — **внешним**.





Типы отношений между таблицами





Пример отношений между таблицами





Пример нормализации отношений

Цель — создание такого проекта базы данных, где будет исключена избыточность информации.

Назначение — исключение возможной противоречивости данных и экономия памяти.

Студент группы

Номер зач.кн.

Фамилия

Имя

Отчество

Группа

Староста

=

Студент

Номер зач.кн.

Фамилия

Имя

Отчество

Группа

+

Группа

Группа

Староста



Особенности работы с таблицами

При удалении записей в поле первичного ключа нумерация не изменяется.

Пример

Было 5 записей с номерами от 1 до 5. После удаления третьей и четвертой записей, оставшиеся будут с прежними номерами 1, 2, 5.

Для компактного отображения полей таблицы часть из них может быть скрыта. Скрытые поля не участвуют в поиске и обработке запросов (Команда *Скрыть столбцы/Отобразить столбцы*).

Для повышения удобочитаемости таблицы используют фиксацию маркированных полей (Команда *Закрепить столбцы/Освободить все столбцы*). В результате фиксированное поле помещается в левой части таблицы и остается видимым при прокрутке полей вправо.

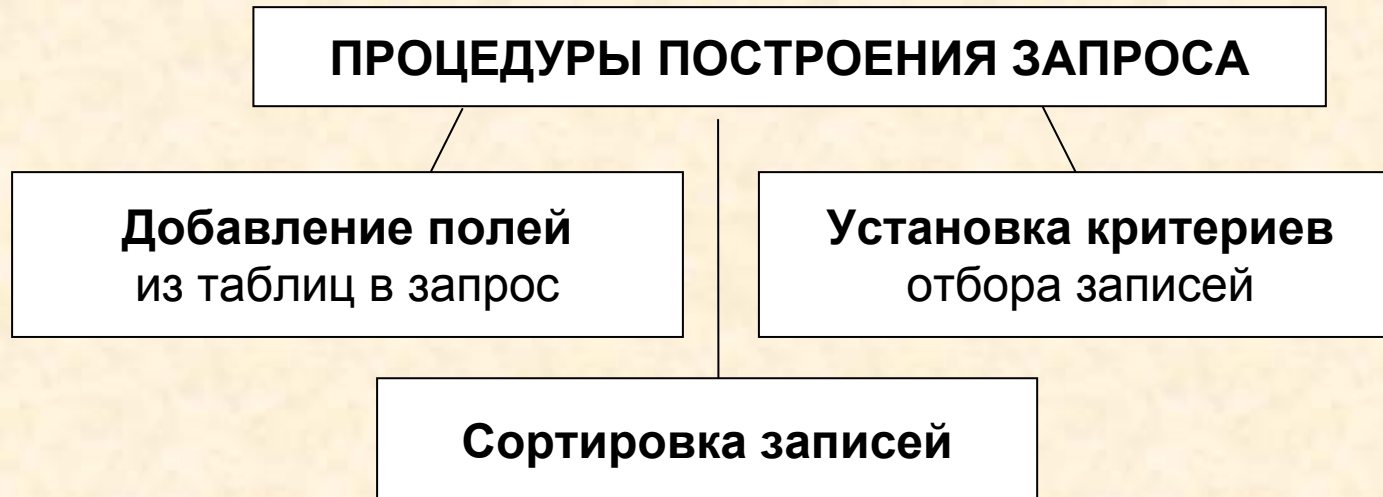
Уменьшение размера поля, в которое вводится данное, ведет к его обрезке по новой границе.



Запросы

Наиболее распространенный запрос — **запрос на выборку**, который выполняет отбор данных из одной или нескольких таблиц по некоторым критериям.

Конструктор позволяет сконструировать запрос на выборку самостоятельно.





Запросы-2

Простой запрос создается на основе выбранных полей.

Перекрестный запрос предоставляет данные в компактном формате как в электронных таблицах.

При составлении запроса на основе нескольких таблиц между ними должны быть установлены связи.

Критерии отбора, расположенные в разных строках, объединяются логическим оператором **ИЛИ**; расположенные в одной строке — логическим оператором **И**.

Условия **И** выполняются раньше условий **ИЛИ**.

Для исключения группы данных надо записать критерий **НЕ РАВНО** «значение» или <> «значение».

Для удаления поля из бланка запроса используется клавиша *Delete*. Для удаления таблицы из запроса надо ее выделить, щелкнув по полю в верхней части окна конструктора запроса и нажать *Delete*.



Вычисления с помощью функций

Группировка. Объединение в одну группу записей с одинаковыми значениями.

Count. Определение числа записей в группе.

Sum. Суммирование значений поля в отобранных запросом записях.

Min. Определение минимального значения в заданном поле группы записей.

Max. Определение максимального значения в заданном поле группы записей.

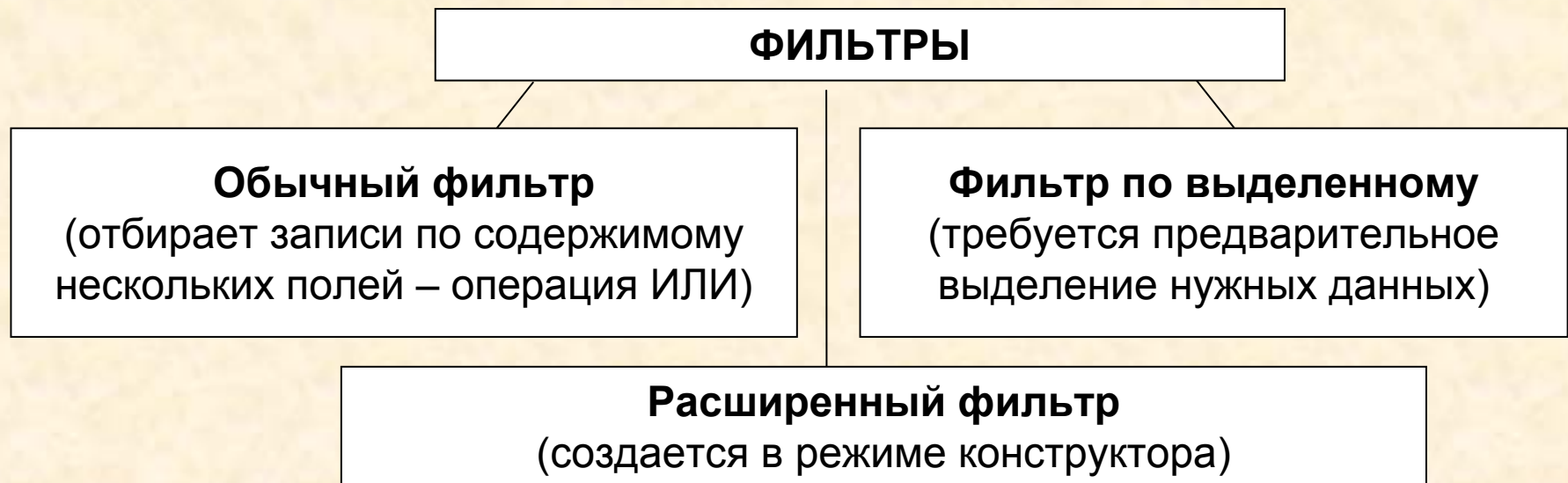
Avg. Определение среднего значения в заданном поле группы записей.

First. Определение первого значения в заданном поле группы записей.

Last. Определение последнего значения в заданном поле группы записей.



Фильтры



Фильтры используются в тех же целях, что и запросы на выборку данных.

При фильтрации нельзя подавить отображение отдельных полей и выполнить вычисления.

В фильтре по выделенному можно **исключить** выделенное.



Формы

Режим таблицы используется, если надо получить наиболее полный обзор данных или для эпизодического изменения данных. При частом изменении данных лучше использовать **форму**.

Формы используются как средство защиты базы данных от действий неквалифицированных пользователей, повышают комфортность работы. Любая форма строится на основе таблицы или запроса. Поля в форме располагаются произвольно. Логически взаимосвязанные поля объединяются в группы. На основе одной таблицы могут быть построены разные формы для различных работников.

Мастер форм в зависимости от назначения формы предлагает на выбор стандартные шаблоны и стили оформления.

Конструктор предоставляет инструменты для создания формы на пустом бланке.



Формы-2

Мастер сводных таблиц создает сводную электронную таблицу на основе таблиц и запросов СУБД.

Мастер диаграмм создает формы с диаграммами на основе выбранных полей таблицы.

По умолчанию значения столбца образуют на диаграмме **ряд данных**, а значения строки — **кате́горию**.

Для отображения числовых значений служат **маркеры данных** — линии, полосы, секторы и т.п.

Метки у маркеров данных представляют числовые или текстовые значения элементов ряда.

Оси используются для отображения шкалы — диапазона значений рядов данных.

Легенда — пояснительный текст и свод условных знаков.



Формы-3

Каждый элемент диаграммы имеет набор параметров, который можно изменить в соответствующем диалоговом окне, дважды щелкнув на нужном элементе.

Круговую диаграмму рекомендуется использовать для отображения соотношения частей целого.

Точечные диаграммы используют для представления взаимосвязи между двумя или несколькими величинами.

Гистограммы используют для сравнения значений, относящихся к определенному моменту времени.

**Содержимое формы всегда соответствует информации
в таблицах и запросах
и обновляется при каждом открытии.**

Удалить запись из формы можно клавишей *Delete*.



Отчеты

Преимуществом **отчетов** перед **формами** является возможность группировки данных. Это не только улучшает внешний вид, но и повышает читабельность отчета.

Данные в отчете группируются не более чем по трем полям. Есть возможность изменения уровня группировки.

В **нижнем колонтитуле** отчета автоматически устанавливается дата его создания.

Заголовок расположен на первой странице отчета перед верхним колонтитулом.

Верхний колонтитул по умолчанию помещается на каждую страницу отчета и обычно содержит заголовки колонок.

На страницах, содержащих шапку или резюме, отображение колонтитула можно заблокировать.



Отчеты-2

Заголовок группы состоит из имени группы и названия *Заголовок группы*.

Элементы **области данных** повторяются для каждой выбранной из таблицы записи.

Примечание группы появляется в конце группы записей. Применяется для указания числа записей или суммы значений в полях группы.

Нижний колонтитул присутствует на каждой странице отчета;. Используется для отображения номера страницы и текущей даты.

Примечание отчета вставляется в конце отчета и содержит резюме ко всему документу.

По умолчанию все области отчета имеют одинаковую ширину. Изменить высоту и ширину области одновременно можно перемещением слева маркировочного квадрата, либо поочередным перемещением ограничительной линии сверху и справа.

Нумерация страниц отчета выполняется в текстовом поле, добавленном в заголовки или колонтитулы.