



## БАЗОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ: ТЕХНОЛОГИЯ АВТОМАТИЗИРОВАННОГО ОФИСА, ТЕХНОЛОГИИ БАЗ ДАННЫХ

- ❖ Технология автоматизированного офиса.
- ❖ Основные компоненты автоматизации офиса.
- ❖ Технологии баз данных.
- ❖ Классификация БД по виду модели.

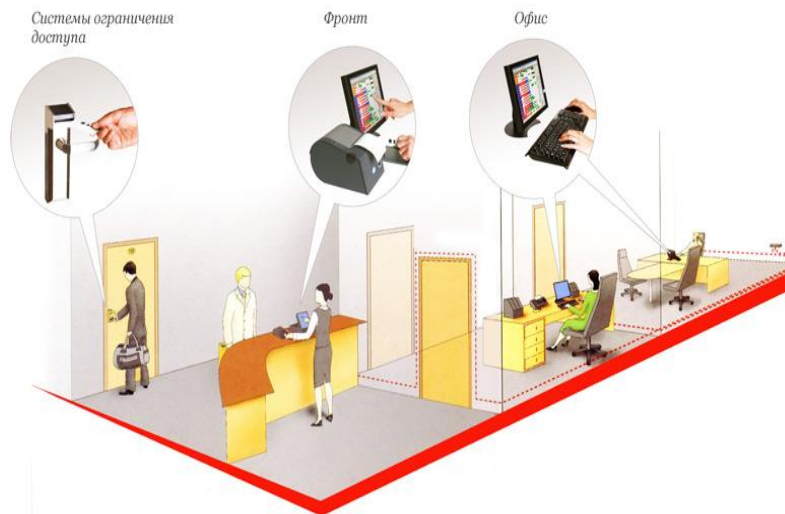
# ИТ автоматизированного предприятия

- **Информационная технология автоматизированного предприятия (офиса)** – организация и поддержка коммуникационных процессов как внутри организации, так и с внешней средой на базе компьютерных сетей и других современных средств передачи и обработки информации.



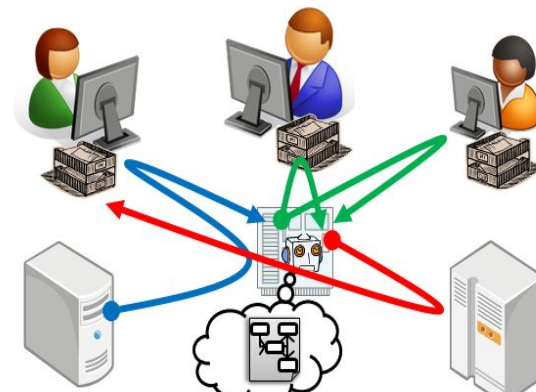
# Зачем нужно автоматизировать офис ???

- ❖ поддерживает внутреннюю связь персонала;
- ❖ дает возможность группового решения задач;
- ❖ повышает производительность труда;
- ❖ предоставляет новые средства коммуникации с внешним окружением.



# Основные компоненты автоматизации офиса

- Текстовый процессор
- Электронная почта
- Аудиопочта
- Табличный процессор
- Электронный календарь
- Компьютерные конференции
- Телеконференция
- Видеотекст
- Хранение изображений
- Базы данных



# Основные компоненты автоматизации офиса

- **Текстовый процессор** – это вид прикладного программного обеспечения, предназначенный для создания и обработки текстовых документов.
- **Электронная почта (e-mail)**, основываясь на сетевом использовании компьютеров, дает возможность пользователю получать, хранить и отправлять сообщения по сети.
- **Аудиопочта** – это почта для передачи сообщений голосом. Она напоминает электронную почту, но сообщение вместо набора на клавиатуре передается через телефон.



# Основные компоненты автоматизации офиса

- **Табличный процессор**, так же как и текстовый процессор, является базовой составляющей автоматизированной офисной технологии. Функции современных программных сред табличных процессоров позволяют выполнять многочисленные операции над данными, представленными в табличной форме.



- **Электронный календарь** предоставляет еще одну возможность использовать сетевой вариант компьютера для хранения и манипулирования рабочим расписанием.

# Основные компоненты автоматизации офиса

**Компьютерные конференции** используют компьютерные сети для обмена информацией между участниками группы, решающей определенную проблему.



Круг лиц, имеющих доступ к этой технологии, ограничен. Количество участников компьютерной конференции может быть во много раз больше, чем участников аудио- и видеоконференций.

# Основные компоненты автоматизации офиса

**Телеконференция** включает в себя три типа конференций:

- ❖ **аудио**
- ❖ **видео**
- ❖ **компьютерную**





# Видеотекст

**Видеотекст** основан на использовании компьютера для получения отображения текстовых и графических данных на экране монитора. Для лиц, принимающих решение, имеются три возможности получения информации в форме видеотекста:

- ❖ создать файлы видеотекста на своих собственных компьютерах;
- ❖ заключить договор со специализированной компанией на получение доступа к разработанным ею файлам видеотекста;
- ❖ заключить договоры с другими компаниями на получение доступа к их файлам видеотекста.



# Хранение изображений

**Хранение изображений** (imaging) – перспективная офисная технология, которая основывается на использовании специального устройства – оптического распознавателя образов, позволяющего преобразовывать изображение документа или фильма в цифровой вид для дальнейшего хранения во внешней памяти компьютера.

- Для хранения изображений используются оптические диски, обладающие огромными емкостями. Созданию данной технологии способствовало появление нового технического решения – оптического диска в комбинации с цифровой записью изображения.



# Базы данных

- Обязательным компонентом любой технологии является база данных.
- В автоматизированном офисе она концентрирует в себе данные о производственной системе, так же как в технологии обработки данных на операционном уровне.
- Информация в базу данных может также поступать из внешнего окружения фирмы.



# Технологии баз данных

**База данных (БД)** – это структурированный организованный набор данных, описывающий характеристики какой-либо физической или виртуальной системы.

## Организация структуры БД формируется исходя из следующих соображений:

- • Адекватность описываемому объекту/системе – на уровне концептуальной и логической модели.
- • Удобство использования для ведения учёта и анализа данных – на уровне так называемой физической модели.



# СУБД

Для создания и манипулирования базой данных используется специализированная программа, называемая системой управления базой данных (СУБД).

## Основные функции СУБД:

- • управление данными во внешней памяти (на дисках);
- • управление данными в оперативной памяти;
- • журнализация изменений и восстановление базы данных после сбоев;
- • поддержка языков БД (язык определения данных, язык манипулирования данными).



# Компоненты СУБД

- ❖ **ядро**, которое отвечает за управление данными во внешней и оперативной памяти и журнализацию;
- ❖ **процессор языка базы данных**, обеспечивающий оптимизацию запросов на извлечение и изменение данных и создание, как правило, машинно-независимого исполняемого внутреннего кода;
- ❖ **подсистема поддержки времени исполнения**, которая интерпретирует программы манипуляции данными, создающие пользовательский интерфейс с СУБД;
- ❖ **сервисные программы** (внешние утилиты), обеспечивающие ряд дополнительных возможностей по обслуживанию информационной системы.



# Классификация СУБД

По архитектуре организации хранения данных СУБД подразделяются на:

- ❖ **локальные** (все части локальной СУБД размещаются на одном компьютере)
- ❖ **распределенные** (части СУБД могут размещаться на двух и более компьютерах).



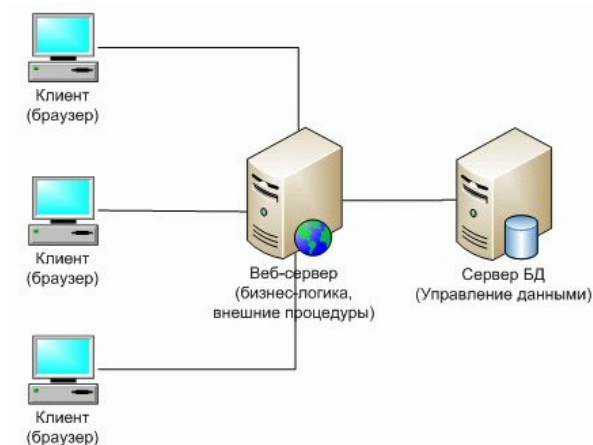
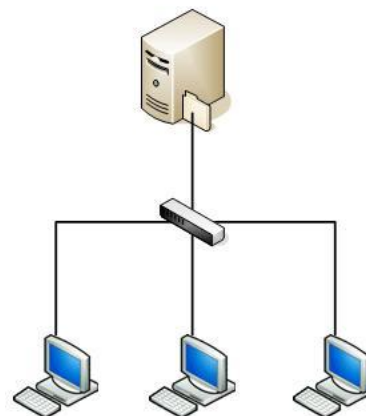
# Классификация СУБД

По способу доступа к БД СУБД разделяются на:

- файл-серверные
- клиент-серверные

**Архитектура «файл-сервер»** не имеет сетевого разделения компонентов диалога и использует компьютер для функции отображения, что облегчает построение графического интерфейса.

**Клиент-серверная СУБД** позволяет обмениваться клиенту и серверу минимально необходимыми объёмами информации. При этом основная вычислительная нагрузка ложится на сервер.

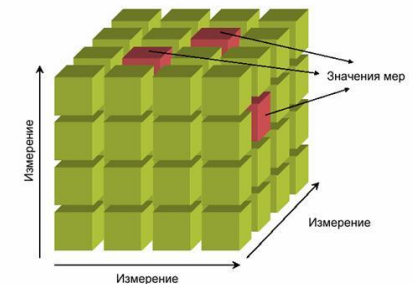
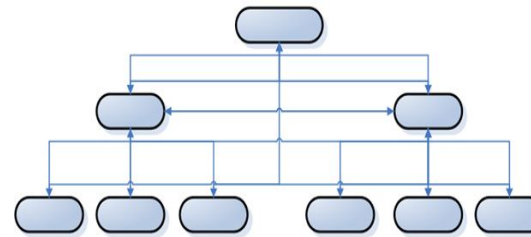
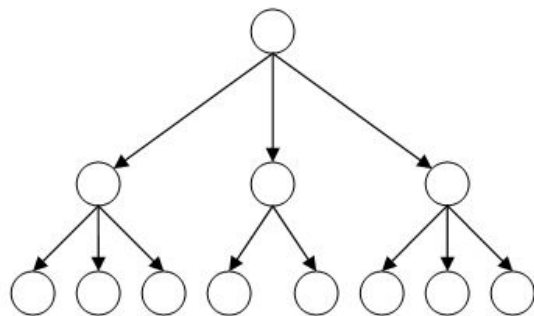
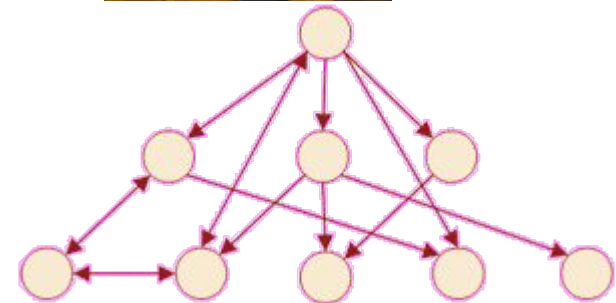




# КЛАССИФИКАЦИЯ БД ПО ВИДУ МОДЕЛИ

## Существующие виды концептуальных и логических моделей БД:

- картотека
- сетевая модель
- иерархическая модель
- реляционная модель
- многомерная модель
- объектная модель.



# Картотека

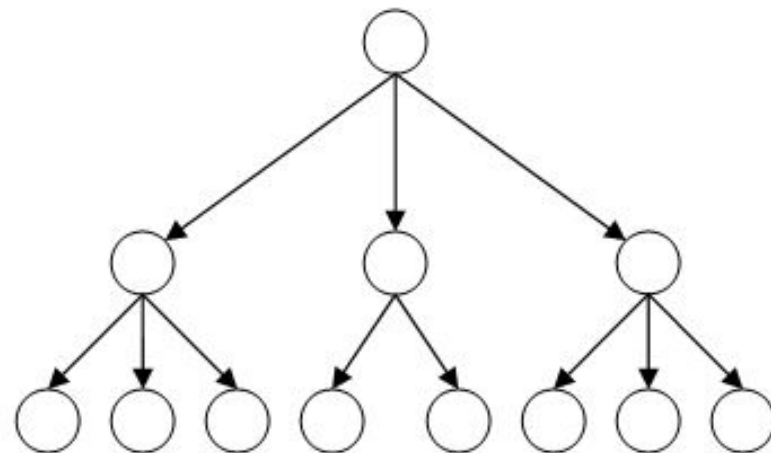
**Картотекой называется систематизированное хранилище информации, как правило, в форме карточек с некоторыми данными.**

- Встретиться с картотекой до сих пор можно, к примеру, в библиотеке: в виде картотеки зачастую представляется библиотечный каталог.
- Картотеками повсеместно пользовались до появления электронных баз данных: в настоящее время картотеки почти полностью вытеснены последними.



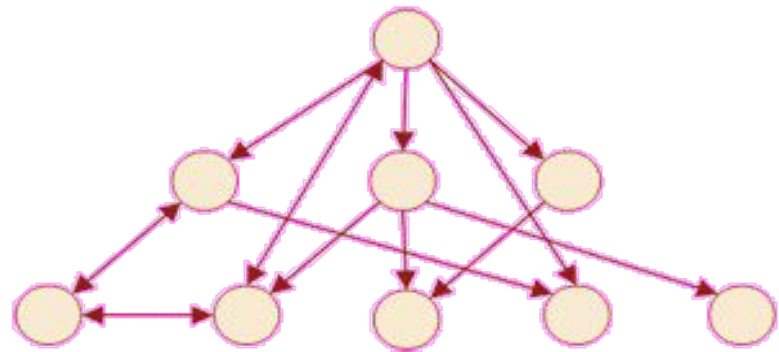
# Иерархическая модель

- **Иерархическая модель**  
базы данных состоит из объектов с указателями от родительских объектов к потомкам, соединяя вместе связанную информацию. Например, если иерархическая база данных содержит информацию о покупателях и заказах, то будет существовать родительский объект «покупатель» и дочерний объект «заказ».



# Сетевая модель

- **Сетевые базы данных** подобны иерархическим, за исключением того, что в них имеются указатели в обоих направлениях, которые соединяют родственную информацию.
- К основным понятиям сетевой модели базы данных относятся уровень, элемент (узел), связь.
- **Узел** – это совокупность атрибутов данных, описывающих некоторый объект. В сетевой структуре каждый элемент может быть связан с любым другим элементом.

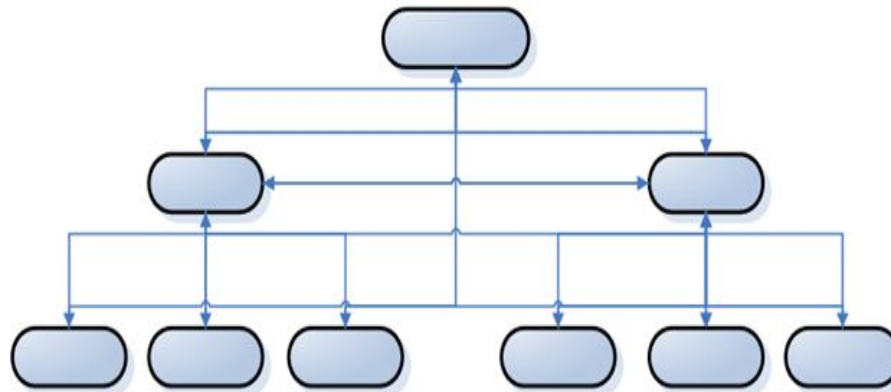


# Реляционная модель

Реляционная модель ориентирована на организацию данных в виде двумерных таблиц.

**Каждая из этих таблиц обладает следующими свойствами:**

- Каждый элемент таблицы – один элемент данных.
- Все столбцы в таблице однородны, т.е. все элементы в столбце имеют одинаковый тип (числовой, символьный и т.д.).
- Каждый столбец имеет уникальное имя.
- Одинаковые строки в таблице отсутствуют.
- Порядок следования строк и столбцов может быть произвольным.

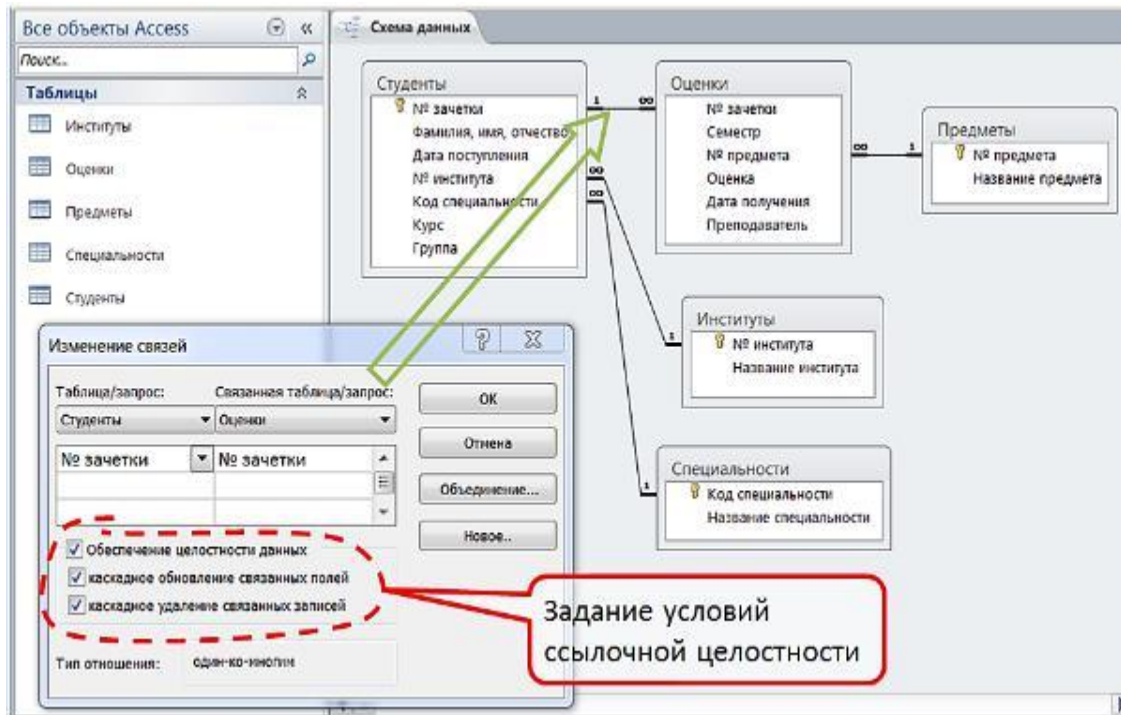


## Виды связей между таблицами:

- **Связь «один к одному».** На каждое значение первичного ключа первой таблицы ссылается не более одной записи второй таблицы.
- **Связь «один ко многим».** На каждое значение первичного ключа первой таблицы может ссылаться множество записей второй таблицы.
- **Связь «многие ко многим».** Для организации этой разновидности связи создается отдельная таблица, называемая таблицей связи или таблицей ассоциации, каждая запись которой содержит значения первичных ключей двух связываемых записей в разных таблицах.

# Ссылочная целостность

- ❖ Необходимым качеством реляционной базы данных является т. н. ссылочная целостность, заключающаяся в отсутствии в любой таблицы базы данных внешних ключей, ссылающихся на несуществующие записи в этой или других таблицах.



База данных, в которой ссылочная целостность нарушена, не может нормально эксплуатироваться, т.к. в ней разорваны связи между зависимыми объектами или даже между частями одного и того же объекта.

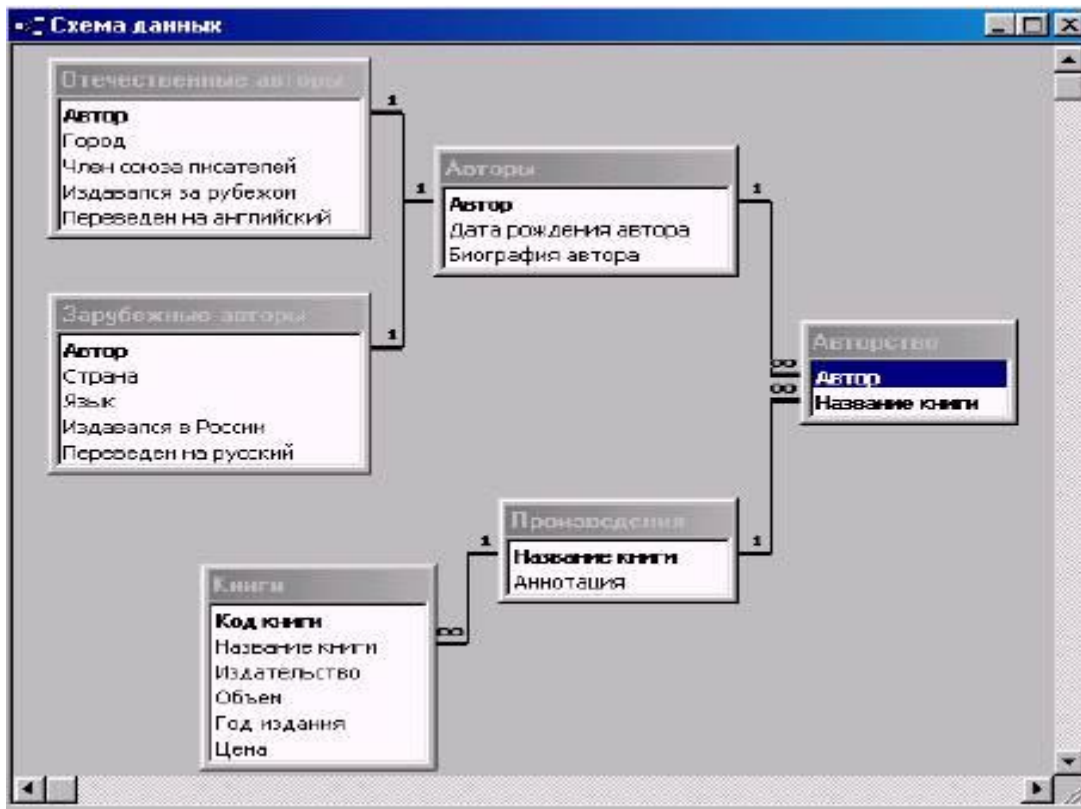
## Автоматическая проверка ссылочной целостности

- При операции добавления или редактирования записи автоматически проверяется, ссылаются ли внешние ключи в этой записи на существующие записи в заявленных при описании связанных таблицах. Если выясняется, что операция приведет к появлению некорректных ссылок, она отменяется.
- При операции редактирования записи проверяется, не изменяется ли ее первичный ключ, и нет ли на нее ссылок. Если первичный ключ изменяется, и при этом на данную запись имеются ссылки, то операция редактирования отменяется или же происходит каскадное обновление внешних ключей в связанных таблицах.
- При операции удаления записи проверяется, нет ли на нее ссылок. Если ссылки имеются, то удаление отменяется, либо происходит каскадное удаление связанных записей.



# Нормализация

- Для устранения из БД избыточных функциональных зависимостей между полями таблиц используется **нормализация** – процесс преобразования БД к виду, соответствующему одной из т.н. нормальных форм.



Если таблица не соответствует нормальной форме, она может быть приведена к ней (нормализована) за счет декомпозиции, т. е. разбиения на несколько таблиц, связанных между собой.

# Виды нормальных форм

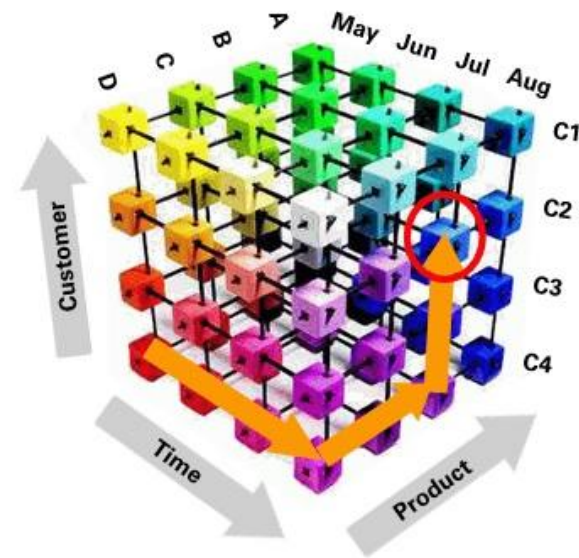
- **Первая нормальная форма (1NF).** Таблица находится в первой нормальной форме, если каждое из ее полей содержит только одно значение, и все строки различны.
- **Вторая нормальная форма (2NF).** Таблица находится во второй нормальной форме, если она находится в первой нормальной форме, и при этом любое ее поле, не входящее в состав первичного ключа, зависит от первичного ключа, но при этом не находится в зависимости от какой-либо его части.
- **Третья нормальная форма (3NF).** Таблица находится в третьей нормальной форме, если она находится во второй нормальной форме, и при этом любое ее неключевое поле функционально зависит *только* от первичного ключа.

# Многомерная модель

**Многомерная модель** рассматривает данные либо как факты с соответствующими численными параметрами, либо как текстовые измерения, которые характеризуют эти факты. К примеру, в розничной торговле покупка – это факт, объем покупки и стоимость – параметры, а тип приобретенного продукта, время и место покупки – измерения.

## преимуществами использования:

- Возможность анализа больших объемов данных с приемлемой скоростью.
- Возможность осуществления любых «срезов» и «углублений» в структуре БД.
- Быстрая локализация трендов и проблемных областей.



# Объектная модель

- В объектно-ориентированной БД данные оформлены в виде моделей объектов, включающих прикладные программы, которые управляются внешними событиями.
- Объектно-ориентированный подход представляет более совершенные средства для отображения реального мира, чем реляционная модель

## Недостатки

отсутствуют мощные непроцедурные средства извлечения объектов из базы, а вместо декларативных средств ограничений целостности приходится писать процедурный код.

**Примеры объектных СУБД: IBM Lotus Notes/Domino, Jasmine, ObjectStore.**



Спасибо за внимание