

#system

#analyst



Основы баз данных

Люба Вайгель

Ментор ASAP Education

Занятие 14



**Для чего используется
сиквенс диаграмма?**

**Как на сиквенсе показать
асинхронное сообщение?**

План занятия

→ Базы данных

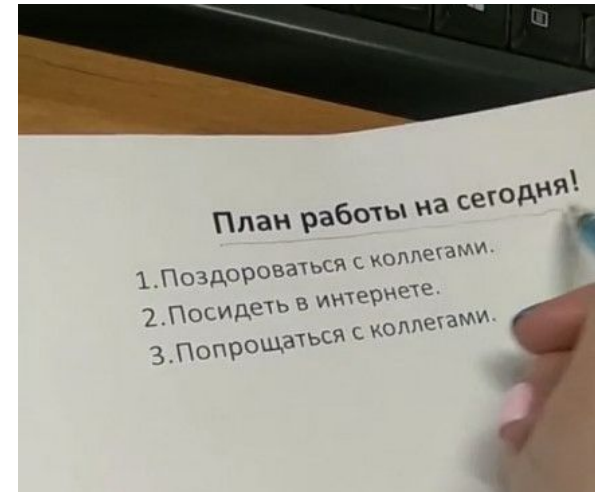
Зачем, почему и ~~за что~~ как

→ Виды баз данных

Про разные виды и про их преимущества

→ Моделирование предметной области

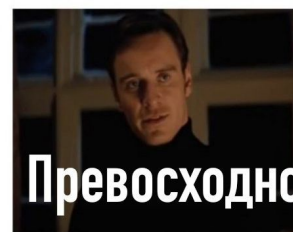
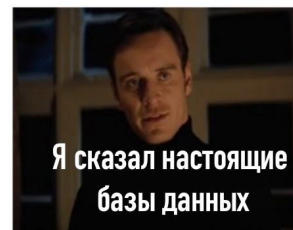
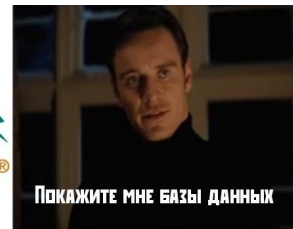
Логическая модель данных, сущности и связи



База данных (БД) — это имеющая название *совокупность данных*, которая отражает состояние объектов и их отношений в рассматриваемой предметной области

А почему бы не хранить все в файлах?

Почему не эксель?



Почему БД?

1. Быстродействие
2. Простота получения и обновления данных
3. Независимость структуры (изменения данных не влияют на структуру БД)
4. Стандартизация (при обновлении ПО база данных не меняет структуру)
5. Безопасность данных
6. Интегрированность (данные логически связаны)
7. Многопользовательский доступ

Операции над данными

- Добавление
- Просмотр элемента
- Изменение
- Удаление



CREATE

C



READ

R



UPDATE

U



DELETE

D

Виды баз данных

Нереляционные (NoSQL)

- Документные
- Ключ-значение
- Графовые
- Колоночные

Реляционные

- SQL
- NewSQL



Нереляционные БД

NoSQL

ПТИШЕМ РЕЗЮМЕ



Документные БД

- Данные хранятся в виде документов (JSON, XML)
- Внутри документа могут быть ссылки на другие документы
- Быстро и легко записываются любые типы данных, при этом эти данные не обязаны обладать четкой структурой
- Неудобно анализировать данные



Content
<pre>{ name: "Ann", surname: "Smith" age: "35", address: "24 Finsbury Street" }</pre>
<pre>{ name: "Boris", surname: "Smith" age: "51", address: "15 St Albans Street" }</pre>

БД ключ-значение

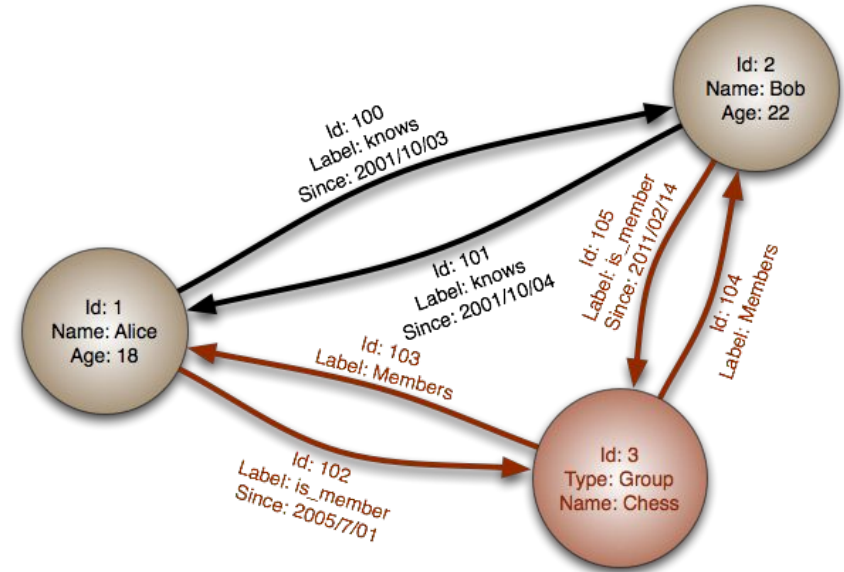
- Представляют собой словарь, хранящий данные в памяти
- Быстрее остальных БД
- Часто используются как вспомогательное звено (кеширование, сохранение статуса сессии и т.д.)

Key	Value
1	"apple"
2	1
3	{name: "Ann", surname: "Smith"}
4	null
5	"cherry"



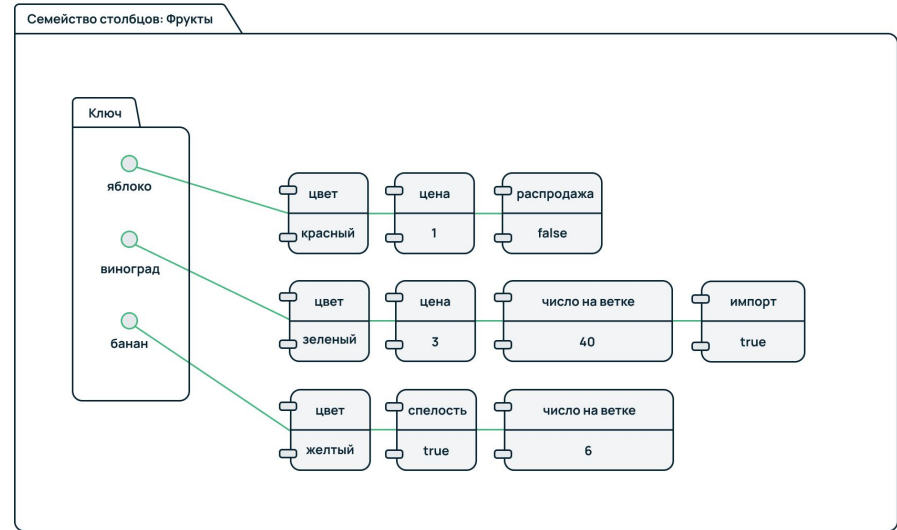
Графовые БД

- Данные хранятся в виде графа
- Вершины графа содержат данные
- Рёбра содержат тип связи, могут быть направленными
- Позволяет находить информацию не только об объекте, но и о связях этого объекта с другими



Колоночные БД

- Данные записаны в виде столбцов
- Поиск данных реализуется поиском необходимого значения в тех столбцах таблицы, которые нас интересуют



Нереляционные БД

ПРЕИМУЩЕСТВА

- Горизонтальное масштабирование
- Высокая устойчивость
- Производительность
- Гибкость
- Широкая применимость

НЕДОСТАТКИ

- Ограниченность языка
- Недостаточная надежность транзакций

Перерыв

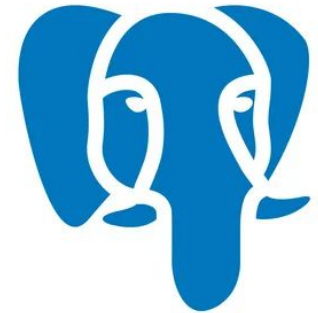


Реляционные БД



Реляционные базы данных

- Данные хранятся в структурированных таблицах
- Колонки связанных данных
- Каждая запись уникальна (кортеж)
- Связи между таблицами

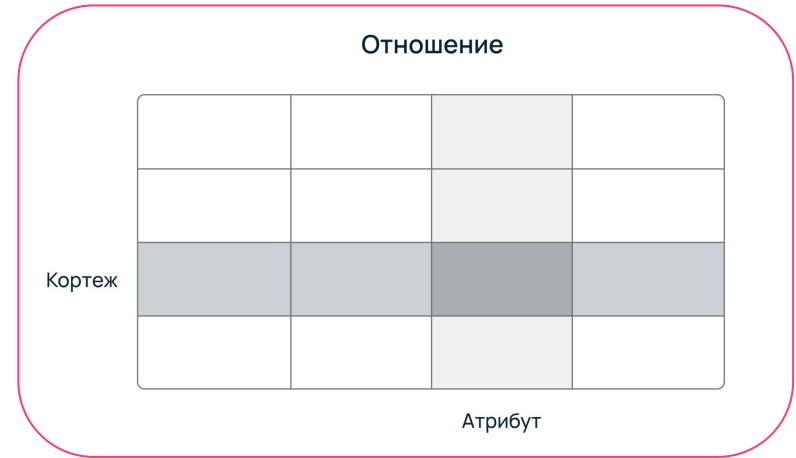


Структура РБД

- Таблицы (отношения)
- Строки (кортежи)
- Столбцы (атрибуты)
- Первичные ключи
- Внешние ключи

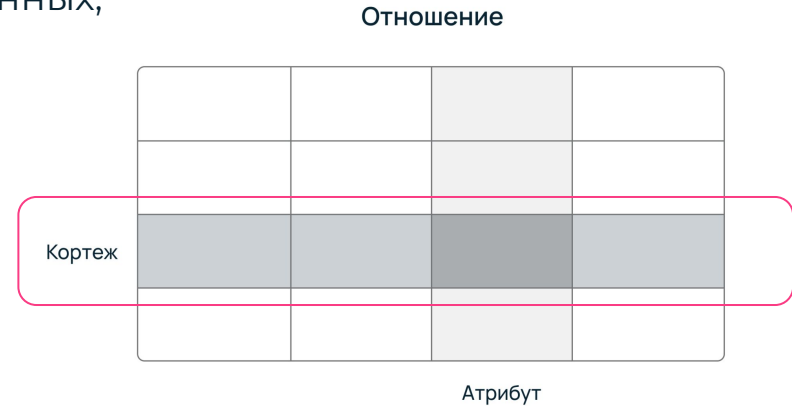
Структура РБД

- **Таблицы** (отношения): основной элемент БД, используется для хранения данных об определенном объекте
- Строки (кортежи)
- Столбцы (атрибуты)
- Первичные ключи
- Внешние ключи



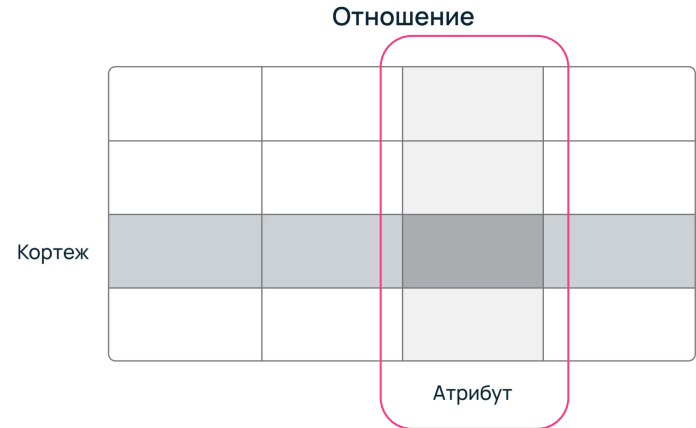
Структура РБД

- Таблицы (отношения)
- **Строки** (кортежи): отдельный экземпляр данных, который содержит значения для каждого атрибута
- Столбцы (атрибуты)
- Первичные ключи
- Внешние ключи



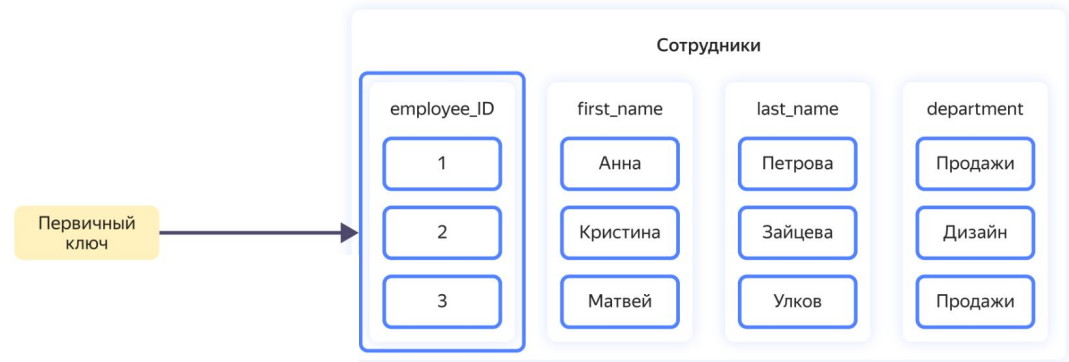
Структура РБД

- Таблицы (отношения)
- Строки (кортежи)
- **Столбцы** (атрибуты): выполняют функцию наименования столбцов и содержательно описывают смысл и назначение элементов данных в соответствующих ячейках
- Первичные ключи
- Внешние ключи



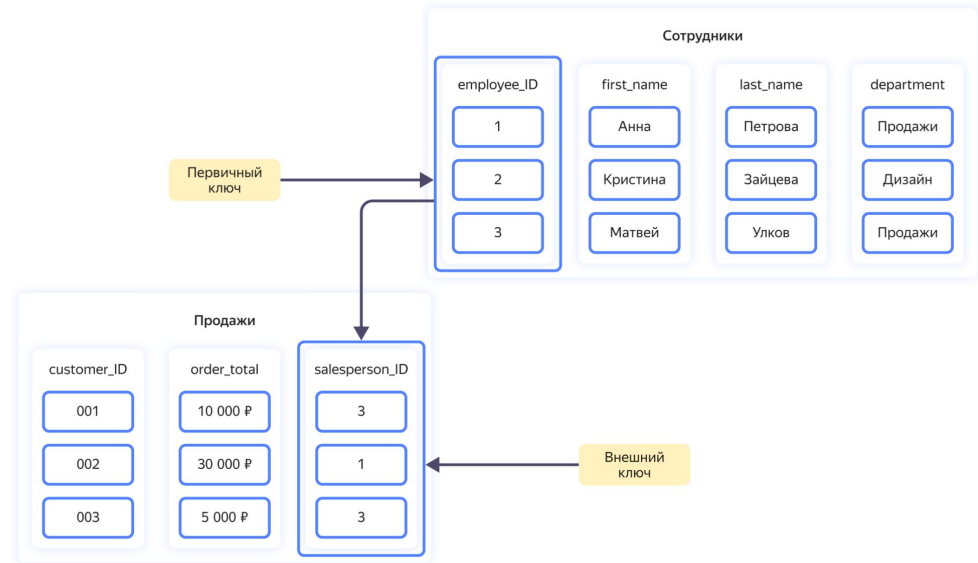
Структура РБД

- Таблицы (отношения)
- Строки (кортежи)
- Столбцы (атрибуты)
- **Первичные ключи:**
идентификаторы, которые
гарантирует уникальность
каждой строки
- Внешние ключи



Структура РБД

- Таблицы (отношения)
- Строки (кортежи)
- Столбцы (атрибуты)
- Первичные ключи
- **Внешние ключи:** атрибуты или их набор в одной таблице, которые ссылаются на первичный ключ или любой другой столбец с уникальными значениями второй таблицы



Принципы РБД

- Использование языка SQL для создания, изменения и удаления сущностей БД , а также выполнения запросов к данным, включая фильтрацию, сортировку и агрегирование
- Обеспечение целостности данных: поддерживается с помощью ограничений (constraints), которые гарантируют, что данные остаются корректными и согласованными
- Соблюдение требований ACID для каждой транзакции

Ограничения

Основные типы ограничений:

- первичный ключ (Primary Key) обеспечивает уникальность каждой записи в таблице
- внешний ключ (Foreign Key) устанавливает связь между значением в одном столбце и уникальным значением в другом (в другой таблице)
- уникальность (Unique) гарантирует, что все значения в столбце или комбинации столбцов будут уникальными
- NOT NULL запрещает пустые значения атрибута

Нужны для поддержания
целостности
и корректности данных

Транзакции и ACID

Транзакция — это последовательность операций с данными в БД

ACID — это набор требований, которые делают транзакции надежными

- **Атомарность (Atomicity)** — все операции выполняются полностью или не выполняются вообще
- **Согласованность (Consistency)** — транзакция переводит базу данных из одного согласованного состояния в другое
- **Изолированность (Isolation)** — результаты выполнения транзакции не видны другим транзакциям до ее завершения
- **Долговечность (Durability)** — после завершения транзакции все изменения сохраняются и остаются неизменными, даже в случае сбоя системы

Согласованное состояние (ACID)

Согласованность (Consistency) — транзакция переводит базу данных из одного *согласованного* состояния в другое

После транзакции БД должна оставаться в *допустимом* состоянии

То есть в каждом столбце значения относятся к нужным типам данных и необходимые значения будут предоставлены

[Согласованность данных: что это на самом деле такое и почему с ней все так сложно](#)
[Согласованность данных в высоконагруженных системах / Хабр](#)

Реляционные БД

ПРЕИМУЩЕСТВА

- Согласованность данных
- Хранимые процедуры
(заранее подготовленные SQL-запросы, которые хранятся на сервере БД и могут вызываться многократно)
- Блокировка и контроль параллелизма
(Когда одна транзакция блокирует данные, другие не могут их изменять)
- Простота использования

НЕДОСТАТКИ

- Работа только со структурированными данными
- Горизонтальное масштабирование и задержки

**В каких случаях стоит
применять ту или иную БД?**

В каких случаях стоит применять ту или иную БД?



Виды БД: преимущества,
недостатки, использование

Перерыв

джун

Блин, я очень боюсь, что не справлюсь и меня уволят. Работаю 24/7

сениор

вт
21

11:00
12:00
13:00

Поплакаться и выпить кофе

Моделирование данных и предметная область

Моделирование данных — это создание визуального представления о всей информационной системе либо ее части

Цель моделирования: проиллюстрировать типы данных, которые используются и хранятся в системе, отношения между этими типами данных, способы группировки и организации данных, их форматы и атрибуты

Основные термины

Сущность - это «вещи» в бизнес-среде, о которых мы хотим хранить данные, например, продукты, клиенты, заказы и т.д.

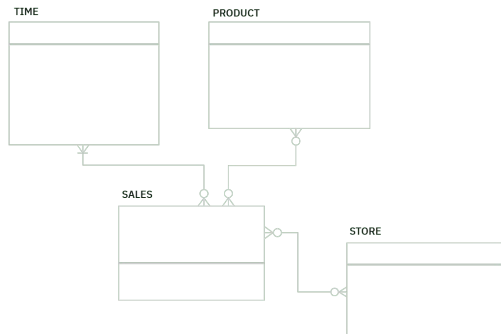
Атрибут - используются для организации и структурирования данных

Взаимосвязь - Взаимосвязь между сущностями описывает, как одна сущность связана с другой

Уровни моделирования

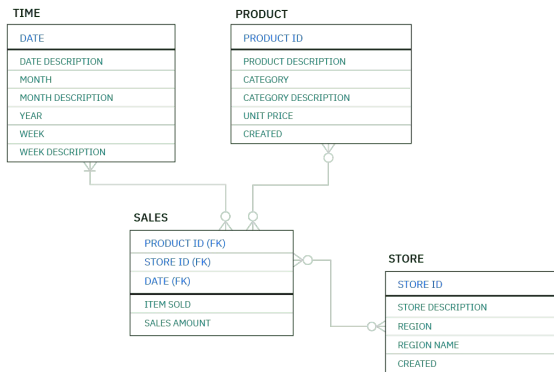
Концептуальный

- Сущности
- Взаимосвязи



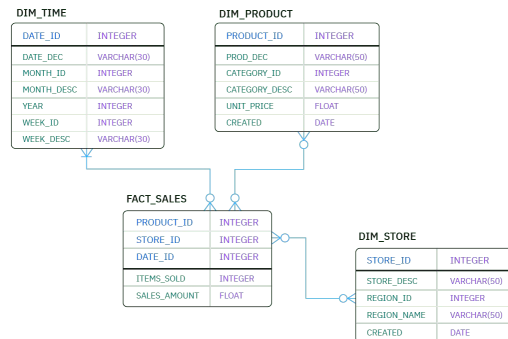
Логический

- Сущности
- Взаимосвязи
- Атрибуты



Физический

- Сущности
- Взаимосвязи
- Атрибуты
- Ключи



Процесс моделирования

1. **Определите сущности:** идентифицируйте объекты, события или концепции, представленные в наборе данных, который необходимо смоделировать
2. **Определите ключевые свойства каждой сущности**
3. **Определите связи между сущностями**
4. **Сопоставьте атрибуты с сущностями**
5. **Назначьте ключи и определите степень нормализации**
6. **Завершите и проверьте модель данных**

Процесс моделирования

1. Определите сущности: идентифицируйте объекты, события или концепции, представленные в наборе данных, который необходимо смоделировать
2. Определите ключевые свойства каждой сущности
3. Определите связи между сущностями
4. Сопоставьте атрибуты с сущностями
5. Назначьте ключи и определите степень нормализации
6. Завершите и проверьте модель данных

Концептуальный
уровень

Логический
уровень

Физический
уровень

Нормализация БД

В процессе нормализации данные преобразуют таким образом, чтобы они занимали меньше места, а поиск по элементам был быстрым и результативным

Для чего нужна нормализация

- Облегчает управление данными
- Снижает нагрузку для хранения однотипных данных
- Увеличивает сложность составления запросов и усложняет запись
- Снижает нагрузку для обновления типов данных

Вопросы для рефлексии

- Что такое база данных?
- Какие виды баз данных бывают?
- В чем особенность реляционных БД?
- Что такое отношение, атрибут, кортеж?
- Для чего используются первичные и внешние ключи?
- Что такое транзакция в БД?
- Какими свойствами должна обладать транзакция?
- Для чего нужно моделирование данных?
- Какие уровни моделирования данных существуют?
- Для чего нужна нормализация БД?

ДЗ

Пройти Модуль 0 и Модуль 1
интерактивного курса по SQL:
<https://sql-academy.org/ru/guide>

Материалы