

Термины которые могут пригодиться в этом модуле

1. **База данных (Database):** Сборник упорядоченных данных. *Пример:* Как библиотека книг, где каждая книга - это набор данных.
2. **Система управления базами данных (СУБД, Database Management System - DBMS):** Программное обеспечение для создания, управления и манипулирования данными в базе данных. *Пример:* Как библиотечный каталог, который помогает найти, организовать и управлять книгами.
3. **Таблица (Table):** Основная единица хранения в реляционной базе данных, состоящая из строк и столбцов. *Пример:* Таблица с контактной информацией, где каждый столбец - это тип информации (имя, телефон), а каждая строка - отдельный контакт.
4. **Строка (Row):** Отдельная запись или набор данных в таблице. *Пример:* Конкретная контактная информация одного человека в таблице контактов.
5. **Столбец (Column):** Вертикальный элемент в таблице, представляющий определенный тип данных. *Пример:* Столбец "Имя" в таблице контактов, содержащий имена людей.
6. **Первичный ключ (Primary Key):** Уникальный идентификатор для каждой строки в таблице. *Пример:* Номер студенческого билета в таблице студентов, где каждый номер уникален для каждого студента.
7. **Внешний ключ (Foreign Key):** Столбец в одной таблице, который ссылается на первичный ключ в другой таблице, создавая связь между ними. *Пример:* Номер отдела в таблице сотрудников, который ссылается на таблицу отделов.
8. **Запрос (Query):** Способ извлечения или обновления данных в базе данных. *Пример:* Запрос на получение списка всех книг автора "Иван Иванов" из библиотечной базы данных.
9. **Нормализация (Normalization):** Процесс организации данных в базе данных для минимизации дублирования и зависимости. *Пример:* Разделение таблицы, содержащей информацию о студентах и их курсах, на две отдельные таблицы: студенты и курсы.
10. **Схема базы данных (Database Schema):** Структурное описание всей базы данных, включая таблицы, поля, отношения между ними. *Пример:* Архитектурный план библиотеки, показывающий расположение всех книг и каталогов.

11. **SQL базы данных** (реляционные базы данных): используют структурированный язык запросов (SQL) для управления и манипулирования реляционными данными. Они организуют данные в таблицы, состоящие из строк и столбцов, и обеспечивают строгую структуру, целостность данных и мощные возможности запросов. *Пример:* Представьте магазин, который ведет базу данных своих клиентов. В этой базе данных есть таблица "Клиенты", где каждая строка представляет отдельного клиента. В таблице есть столбцы, такие как "ID клиента", "Имя", "Фамилия", "Электронная почта", "Номер телефона".
12. **NoSQL (Not Only SQL)**: Тип баз данных, предназначенный для гибкого хранения, обработки и извлечения неструктурированных и полуструктурированных данных. *Пример:* Как склад с разными типами коробок, каждая из которых может иметь уникальную форму и содержимое, в отличие от унифицированных полок в обычном складе (как в SQL).
13. **OLTP** - это способ организации данных, который ориентирован на быструю и эффективную обработку большого количества транзакций. Данные в OLTP организованы таким образом, чтобы каждая транзакция была обработана максимально быстро и надежно. Это достигается за счет упрощения каждой транзакции и оптимизации базы данных для быстрого доступа. *Пример:* Каждый раз, когда клиент делает заказ, в системе фиксируется транзакция. Система регистрирует заказ, обновляет данные о доступности блюд и обрабатывает платеж. Все эти операции нужно выполнить быстро, чтобы обеспечить эффективное обслуживание клиентов.
14. **OLAP** - это подход к организации данных, который ориентирован на поддержку сложного анализа и многомерных запросов. В OLAP данные организованы таким образом, чтобы облегчить агрегацию, сравнение и анализ больших объемов информации из разных источников. Это подходит для сложных аналитических задач и поддержки принятия решений. *Пример:* В конце года руководство кафе может использовать OLAP для анализа общих продаж, определения наиболее популярных блюд, сравнения продаж по месяцам или выявления трендов потребления. Здесь важна не скорость каждой отдельной транзакции, а глубина и многосторонность анализа.
15. **Data Warehouse (DWH)**: Система для хранения и анализа больших объемов данных из различных источников, предназначенная для поддержки процессов принятия решений. *Пример:* Как центральный архив, где собирается, систематизируется и хранится информация из разных отделов компании для анализа и составления отчетов.
16. **ETL (Extract, Transform, Load)**: Процесс извлечения данных из различных источников, их преобразования и загрузки в хранилище данных. *Пример:* Сбор данных из разных систем (CRM, ERP) и их интеграция в DWH.