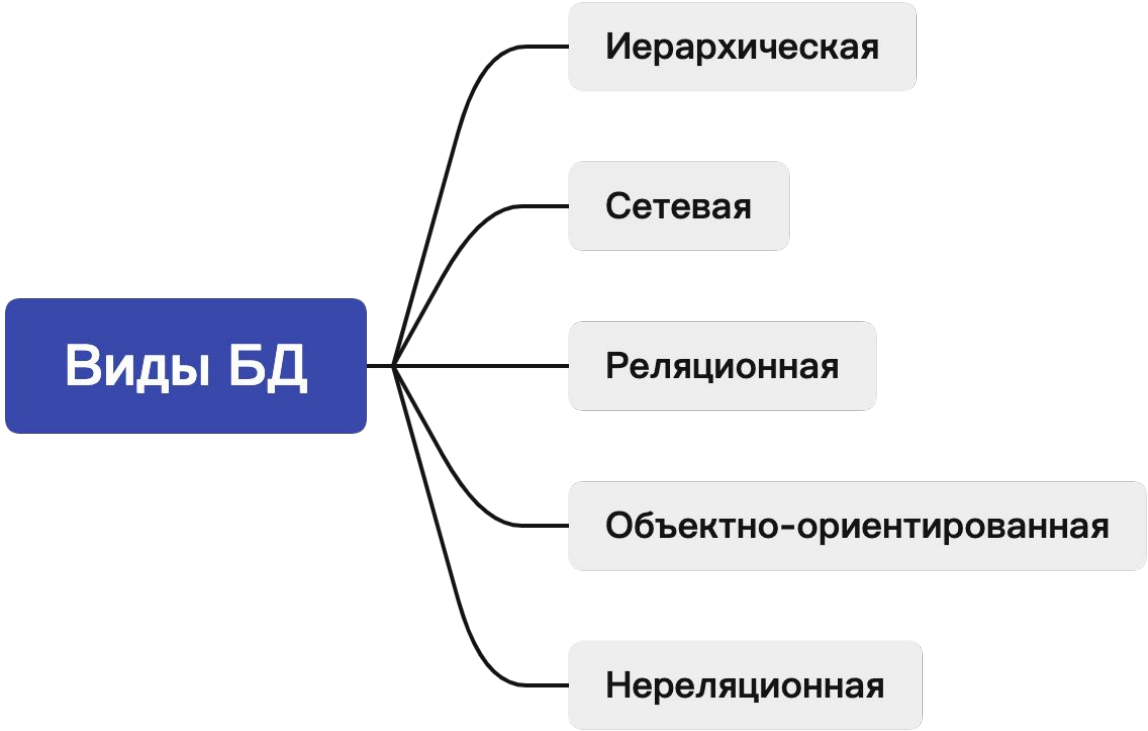
ТЕОРИЯ БД

База данных (database, DB) - упорядоченный набор структурированной информации или данных, которые обычно хранятся в электронном виде в компьютерной системе.

https://www.oracle.com/cis/database/what-is-database/

Система управления базами данных (СУБД, database management system, DBMS) - интерфейс (ПО) между базой данных и пользователями или программами.

https://www.oracle.com/cis/database/what-is-database/



https://www.oracle.com/cis/database/what-is-database/

Реляционные базы данных (SQL)

Формат: таблицы, которые состоят из строк (записей) и столбцов (полей)

SQL (Structured Query Language) — это язык программирования, используемый в большинстве реляционных баз данных для запросов, обработки и определения данных, а также контроля доступа.

Примеры СУБД: MySQL, Microsoft SQL Server, PostgreSQL, Oracle DB

https://www.oracle.com/cis/database/what-is-database/

Структура таблицы

Первичный ключ (Primary Key) - уникальный идентификатор записи в таблице

Primary Key

USERS

Записи

ID	fname	lname	age
1	Alex	Petrov	20
2	Nina	Ivanova	25
3	Viktor	Drozdov	40

Поля

Примечание: первичный ключ это не всегда число. Например, он может выглядеть так 3422b448-2460-4fd2-9183-8000de6f8343 (uuid).

Структура таблицы

Внешний ключ (Foreign Key) - обеспечивает связь между таблицами внутри одной базы данных.

USERS

Foreign Key

ID	fname	lname	age	cities_ID
1	Alex	Petrov	20	c1
2	Nina	Ivanova	25	c2
3	Viktor	Drozdov	40	c3

Primary Key

ID	city
c1	Milan
c2	New York
c3	London

CITIES

Связи (relations) между таблицами

Один к одному (one to one): один человек -> один номер налогоплательщика

USERS

ID	fname	lname	age	INN_ID
1	Alex	Petrov	20	c1
2	Nina	Ivanova	25	c2
3	Viktor	Drozdov	40	c3

INN

ID	INN	users_ID
c1	567895	1
c2	345678	2
c3	123777	3

Связи (relations) между таблицами

Один ко многим (one to many): один город -> много людей

USERS

ID	fname	lname	age	cities_ID
1	Alex	Petrov	20	c1
2	Nina	Ivanova	25	c2
3	Viktor	Drozdov	40	c3
4	Anya	Zhukova	34	c3

CITIES

ID	city
c1	Milan
c2	New York
c3	London

Связи (relations) между таблицами

Многие ко многим (many to many): много стран <-> много валют

CURRENCIES

ID	currency
v1	euro
v2	dollars

ID	currency_id	country_id
1	v1	c1
2	v2	c2
3	v1	c3
4	v2	c3

CURRENCIES_COUNTRIES

COUNTRIES

ID	country
c1	USA
c2	Poland
c3	Neverland

Нормализация баз данных

Нормализация - процесс удаления избыточных данных

Избыточность данных - использование одних и тех же данных в нескольких местах

Нормальная форма - набор правил и критериев, которым должна отвечать база данных

Нормализация нужна для:

- Устранения аномалий
- Повышения производительности
- Повышения удобства управления данными

https://info-comp.ru/database-normalization

Первая нормальная форма (1NF)

- В таблице не должно быть дублирующих строк
- В каждой ячейке таблицы хранится атомарное значение (одно не составное значение)
- В столбце хранятся данные одного типа
- Отсутствуют массивы и списки в любом виде

https://info-comp.ru/first-normal-form

Таблица в нулевой форме

Person	Product	Price	Currency
Alex, male	Macbook, grey	300	$
Irina, female	Samsung S22, white	15	euro
Irina, female	Samsung S22, white	15	euro
Nina, female	Honor 20	thirteen	dollar

Таблица в первой форме

Person	Gender	Product	Model	Color	Price	Currency
Alex	male	Macbook	Pro	grey	300	$
Irina	female	Samsung	S22	white	15	euro
Nina	female	Honor	20	silver	13	dollar

Вторая нормальная форма (2NF)

- Таблица должна находиться в первой нормальной форме
- Таблица должна иметь первичный ключ
- *Все неключевые столбцы таблицы должны зависеть от полного ключа (в случае если он составной)*

https://info-comp.ru/database-normalization

Таблица во второй форме

Order_ ID	Person	Gender	Product	Model	Color	Price	Currency
1	Alex	male	Macbook	Pro	grey	300	$
2	Irina	female	Samsung	S22	white	15	euro
3	Nina	female	Honor	20	silver	13	dollar

Третья нормальная форма (3NF)

- Таблица должна находиться во второй нормальной форме
- В таблицах должна отсутствовать транзитивная зависимость
- **Транзитивная зависимость** – это когда неключевые столбцы зависят от значений других неключевых столбцов.

База данных считается нормализованной, если она находится как минимум в третьей нормальной форме (3NF).

https://info-comp.ru/third-normal-form

Все данные находятся в одной таблице, но их можно разделить и связать через внешний ключ

Order_ ID	Person	Gender	Product	Model	Color	Price	Currency
1	Alex	male	Macbook	Pro	grey	300	$
2	Irina	female	Samsung	S22	white	15	euro
3	Nina	female	Honor	20	silver	13	dollar

Таблица в третье нормальной форме

ID	Person	Gender
1	Alex	male
2	Irina	female
3	Nina	female

ID	Product	Model	Color	Price	Currency
1	Macbook	Pro	grey	300	$
2	Samsung	S22	white	15	euro
3	Honor	20	silver	13	dollar

Денормализация

Денормализация — намеренное приведение структуры базы данных в состояние, не удовлетворяющее требованиям нормализации.

https://info-comp.ru/denormalizing-database

Нереляционные базы данных (NoSQL)

Нереляционная база данных — это база данных, в которой не используется табличная схема строк и столбцов.

Данные могут храниться как простые пары "ключ — значение", документы JSON или граф, состоящий из ребер и вершин. Чаще всего вы будете работать с БД на основе документов.

Примеры СУБД: MongoDB, DynamoDB

https://learn.microsoft.com/ru-ru/azure/architecture/data-guide/big-data/non-relational-data

Реальная жизнь и советы

1. Не всегда на проекте тестировщикам выдают доступ к БД. С вашей стороны вы должны настаивать на выдачу таких доступов, так как мы должны тестировать правильность сохранения данных, а также без БД гораздо сложнее тестировать разрешения и права, другие специфические кейсы
2. Обычно права тестировщиков ограничены. Чаще всего на Prod DB не может работать никто, кроме самых доверенных людей
3. *Миграция* - перемещение данных из одной базы в другой, изменение схемы текущей базы
4. У базы данных должен быть backup (резервная копия)
5. Все операции у DELETE делайте сначала на SELECT (подробнее в следующих уроках)