БД & MySQL

Какие задачи решают БД

- Хранение данных
- Обработка данных (Выборки, обновление, удаление)
- Отказоустойчивость (репликация, шардирование)
- Масштабируемость (несколько инстансов СУБД)

СУБД

- Программа запущенная на компьютере, которая умеет сохранять в БД и предоставлять данные из БД
- Содержит в себе файлы с данными
- Предоставляет возможность подключиться к себе через TCP-сокет.
- Во многих языках программирования в стандартной библиотеке есть методы для подключения к СУБД

Какие бывают базы данных?

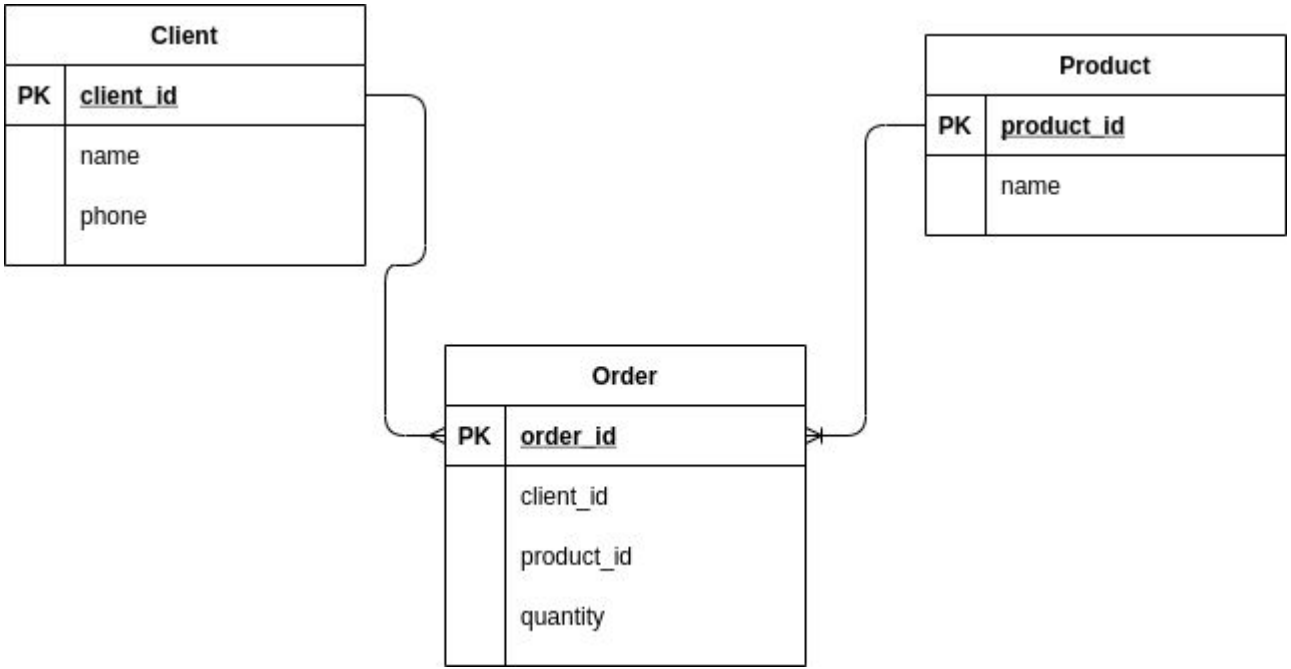
- Реляционные
- Графовые
- Объектные

Реляционные базы данных

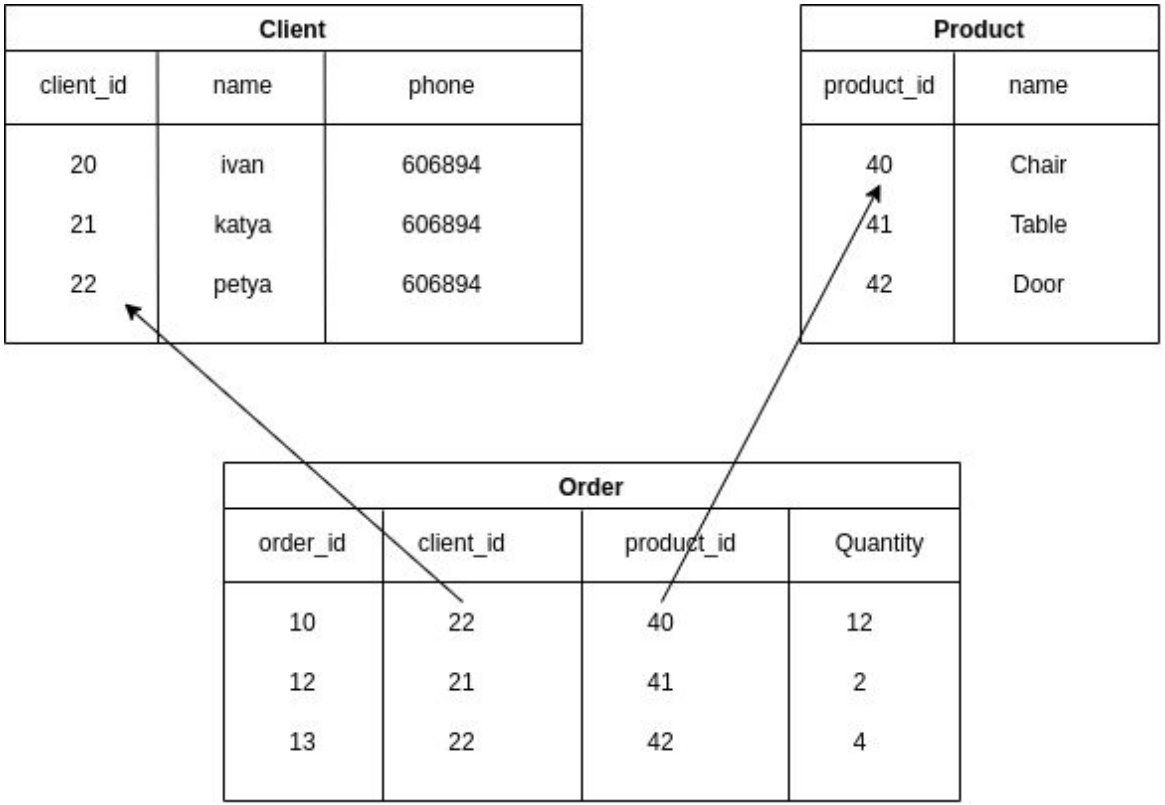
- Данные хранятся в виде таблиц и колонок
- Таблица представляет собой какую-то сущность - например, пост
- Колонка представляет собой информацию о конкретной сущности, например, о конкретном посте

Реляционные базы данных

клиенты-заказы-продукты



Реляционные базы данных - связи между записями



MySQL - база данных

- Реляционная база данных
- Высокопроизводительная БД
- Популярное решение для хранения большого количества данных

Взаимодействие с БД

```
mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.00 sec)

mysql> CREATE DATABASE blog;
Query OK, 1 row affected (0.02 sec)

mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| blog               |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)
```

Добавление таблицы

```
mysql> use blog;
Database changed
mysql> CREATE TABLE post
    -> (
    ->     `post_id`      INT NOT NULL AUTO_INCREMENT,
    ->     `title`        VARCHAR(255) NOT NULL,
    ->     `subtitle`     VARCHAR(255) NOT NULL,
    ->     `publish_date` VARCHAR(255) NOT NULL,
    ->     `featured` TINYINT(1),
    ->     PRIMARY KEY (`post_id`)
    -> ) ENGINE = InnoDB
    -> CHARACTER SET = utf8mb4
    -> COLLATE utf8mb4_unicode_ci
    -> ;
Query OK, 0 rows affected, 1 warning (0.06 sec)
```

Удаление таблицы

```
mysql> SHOW TABLES;
+----------------+
| Tables_in_blog |
+----------------+
| post           |
+----------------+
1 row in set (0.02 sec)

mysql> DROP TABLE post;
Query OK, 0 rows affected (0.05 sec)

mysql> SHOW TABLES;
Empty set (0.00 sec)
```

Добавление записей в таблицу

```
mysql> INSERT INTO post (title, subtitle, publish_date, featured) VALUES ('Post about traveling', 'My best post', '9/25/2015', 1);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO post (title, subtitle, publish_date, featured) VALUES ('Post number 2 about traveling', 'My best number 2 post', '9/25/2015', 0);
Query OK, 1 row affected (0.03 sec)
```

Чтение записей - выборка

```
mysql> SELECT * FROM post;
+---------+-------------------------------+-----------------------+--------------+----------+
| post_id | title                         | subtitle              | publish_date | featured |
+---------+-------------------------------+-----------------------+--------------+----------+
|       1 | Post about traveling          | My best post          | 9/25/2015    |        1 |
|       2 | Post number 2 about traveling | My best number 2 post | 9/25/2015    |        0 |
+---------+-------------------------------+-----------------------+--------------+----------+
2 rows in set (0.00 sec)

mysql> SELECT * FROM post WHERE featured = 1;
+---------+----------------------+--------------+--------------+----------+
| post_id | title                | subtitle     | publish_date | featured |
+---------+----------------------+--------------+--------------+----------+
|       1 | Post about traveling | My best post | 9/25/2015    |        1 |
+---------+----------------------+--------------+--------------+----------+
1 row in set (0.01 sec)
```

Подключение в GO

```
module blog

go 1.20

// Дописываем в go.mod подключение библиотек для работы с MySQL
require (
   github.com/go-sql-driver/mysql v1.7.0
   github.com/jmoiron/sqlx v1.3.5
)

// После чего выполняем в консоли - для обновления зависимостей
go mod tidy
```

Подключение в GO

```
package main

import (
   _ "github.com/go-sql-driver/mysql" // Импортируем для возможности подключения к MySQL
   "github.com/jmoiron/sqlx"
)



func main() {
     // Получаем клиента к БД и ошибку в случае, если не удалось подключиться
     db, err := sql.Open(“mysql”, “root:1234@tcp(localhost:3306)/blog”)
     //...
}
```