

# Introduction

## Задание 1

**Может ли быть в таблице реляционной базы данных несколько строк с повторяющимся РК? Почему?**

Нет, в таблице реляционной базы данных не может быть несколько строк с повторяющимся РК (Primary Key). РК является уникальным идентификатором строки в таблице, и его повторение нарушает целостность данных и возможность однозначного определения строки. Если две или более строки имеют одинаковое значение РК, это может привести к ошибкам при выполнении операций с базой данных.

## Задание 2

**2.1. К какому типу относится СУБД Redis?**

Redis относится к типу NoSQL (неструктурированных) баз данных.

**2.2. В чем особенность данной СУБД при работе с памятью?**

Особенность Redis заключается в том, что он является in-memory базой данных, то есть все данные хранятся в оперативной памяти (RAM). Это позволяет достичь очень высокой скорости чтения и записи данных.

## Задание 3

**3.1. Что является аналогом таблицы (в реляционной СУБД) в MongoDB?**

В MongoDB аналогом таблицы в реляционной СУБД является коллекция (collection). Коллекция представляет собой группу документов, которые могут быть различной структуры. Каждый документ в коллекции представляет отдельную запись с данными. В отличие от таблиц в реляционных СУБД, коллекции в MongoDB не требуют жесткой схемы, то есть каждый документ может иметь различные поля и структуру данных.

### **3.2. Что является аналогом строки/записи (в реляционной СУБД) в MongoDB?**

В реляционных СУБД строка или запись представляет отдельную запись данных с определенными столбцами и значениями. В MongoDB, аналогом строки или записи является документ (document). Документ в MongoDB представляет собой JSON-подобный объект, который состоит из пар ключ-значение, где каждое значение может быть любого типа данных, включая вложенные документы или массивы.

### **3.3. Приведите небольшой пример того, как может выглядеть данный аналог.**

```
{
  "_id": ObjectId("615cb242c2b14d58dbedf849"),
  "name": "John",
  "age": 30,
  "email": "john@example.com",
  "address": {
    "street": "123 Main St",
    "city": "New York",
    "state": "NY",
    "zip": "10001"
  },
  "hobbies": ["hiking", "reading", "cooking"]
}
```

В этом примере документ содержит различные поля, такие как "name", "age", "email", "address" и "hobbies". Поле "\_id" является уникальным идентификатором для этого документа.

Каждый документ в коллекции может иметь различные поля и структуру.

## **Задание 4**

**Выберите наиболее подходящие СУБД для следующих задач:**

### **4.1. Анализ поведения знаменитостей в социальной сети.**

При анализе поведения знаменитостей в социальной сети, где могут возникать большие объемы данных и требуется сложная аналитика, нужно использовать СУБД с поддержкой аналитических запросов и обработкой больших данных. Хорошим выбором может быть Apache Cassandra или Apache HBase. Они предоставляют горизонтальное масштабирование, надежность и мощные возможности аналитики для обработки больших объемов данных.

### **4.2. Онлайн-аналитика действий пользователей на сайте онлайн магазина. Включая информацию о клиентах, товарах, заказах.**

Для онлайн-аналитики действий пользователей и хранения информации о клиентах, товарах и заказах, можно рассмотреть использование NoSQL СУБД, таких как MongoDB или Apache CouchDB. Они предоставляют гибкую структуру данных и хорошо масштабируются. Это особенно важно для онлайн-магазинов, где могут быть частые изменения структуры данных и большие объемы данных.

### **4.3. Анализ документации по фильмам и сериалам, включая: название, год выпуска, список актеров, жанр и т.д.**

Для анализа документации по фильмам и сериалам, где данные обладают структурированной природой, рекомендуется использовать реляционные СУБД, такие как PostgreSQL или MySQL. Они обеспечивают хорошую поддержку для структурированных данных, возможность выполнения сложных запросов и поддержку связей между данными (например, актеры и фильмы).

#### **4.4. Проектирование наиболее оптимальных маршрутов поставок для маркетплейса.**

Для проектирования оптимальных маршрутов поставок, где важна эффективность обработки данных и выполнения сложных запросов, рекомендуется использовать графовые СУБД, такие как Neo4j. Графовые СУБД специализируются на обработке и анализе связей между данными и могут быть полезны в моделировании и поиске оптимальных маршрутов.

#### **4.5. Хранение кэшированных HTML страниц пользователей, посещающих сайт.**

Для кэширования HTML страниц пользователей можно использовать системы кэширования, такие как Memcached или Redis. Эти СУБД предоставляют быстрый доступ к данным и простые механизмы кэширования для хранения готовых HTML страниц. Они часто используются для ускорения доступа к повторно используемым данным и снижения нагрузки на более ресурсоемкие операции.

# Data Manipulation Language (DML)

## Задание 1

Укажите, в каком порядке производится написание команд при выборе данных из таблицы:

``SELECT` `FROM` `WHERE` `GROUP BY` `HAVING` `ORDER BY` `LIMIT``

Также укажите, в каком порядке команды выполняются при запуске отработки запроса.

**1.1. Порядок написания:** ``SELECT``, ``FROM``, ``WHERE``, ``GROUP BY``, ``HAVING``, ``ORDER BY``, ``LIMIT``

### 2.1. Порядок выполнения при запуске:

2.1.1. ``FROM``: указывает, из какой таблицы выбираются данные.

Сначала выполняется операция выборки данных из указанной таблицы.

2.1.2. ``WHERE``: определяет условие, которому должны удовлетворять строки данных, чтобы быть включенными в результат. Здесь происходит фильтрация данных, и только удовлетворяющие условию строки будут выбраны для дальнейшей обработки.

2.1.3. ``GROUP BY``: группирует данные по указанному столбцу или столбцам, затем разделяет данные на группы на основе значения в указанных столбцах.

2.1.4. ``HAVING``: фильтрует группы данных на основе заданного условия.

2.1.5. ``SELECT``: определяет, какие столбцы или выражения должны быть выбраны в результате запроса, затем возвращает результирующий датасет.

2.1.6. ``ORDER BY``: устанавливает порядок сортировки строк данных в результате запроса.

2.1.7. ``LIMIT``: ограничивает количество возвращаемых строк данных в результате запроса.

## Задание 2

Какая команда позволит вам удалить строки таблицы по условию? Можно ли удалить таблицу с помощью данной команды?

2.1. Команда, позволяющая удалить **строки таблицы** по условию - DELETE.

Однако, команда DELETE предназначена только для удаления строк в таблице, а не для удаления самой таблицы.

2.2. Для **удаления таблицы** необходимо использовать команду DROP TABLE.

Для заданий 3 – 7 создан отдельный файл .sql