

УРОК 20.2 REVISION

ПОВТОРЕНИЕ MONGODB	2
Базовое равенство	2
Диапазонные запросы	2
Логические операторы	3
Текстовый поиск	3
Агрегация, пайплайны	5
Группировка	7
Проекция (Project)	7
Сортировка	7
Разделение на группы	7
Фасетирование	8



ПОВТОРЕНИЕ MONGODB

Работаем с коллекцией: ich.Spotify_Youtube.

Базовое равенство

Найти все документы, где упоминается конкретный артист (Gorillaz):

```
Unset
{
  filter: {
    Artist: 'Gorillaz'
  }
}
```

Диапазонные запросы

Найти треки с показателем танцевальности больше 0.8.

```
Unset
{
  filter: {
    Danceability: {
      $gt: 0.8
    }
  }
}
```

Найти все треки с темпом выше 130.

```
Unset
{
  filter: { "Tempo": { $gt: 130 } }
```



```
}
```

Логические операторы

Найти треки с высоким уровнем энергии (Energy выше 0.7) и положительной эмоциональной окраской (Valence выше 0.7).

```
Unset
{
  filter: {
    $and: [
      {
        Energy: {
          $gt: 0.7
        }
      },
      {
        Valence: {
          $gt: 0.7
        }
      }
    ]
  }
}
```

Найти треки определенного артиста (Elton John) с более чем 5000 комментариями.

```
Unset
{ $and: [ { "Artist": "Elton John" }, { "Comments": { $gt: 5000 } } ] }
```

Текстовый поиск

Найти треки с словом "Good" в названии.



```
Unset
{
  filter:
  { "Title": { $regex: /Good/ } } }
```

Найти все треки со словом hot и Hot и HOT в поле Description.

Поиск по точному совпадению (регистрозависимый):

```
Unset
{ "Description": { $regex: /hot/ } }
```

Поиск по частичному совпадению (регистронезависимый):

```
Unset
{ "Description": { $regex: /hot/i } }
```

Поиск по точному совпадению (регистрозависимый) с использованием оператора \$in:

```
Unset
{ "Description": { $in: [/hot/, /Hot/, /HOT/] } }
```

Поиск по нескольким словам (регистронезависимый):

Найти все треки, в описании которых есть и "hot" и "Good" (регистронезависимый).

```
Unset
{ "Description": { $all: [/hot/i, /Good/i] } }
```



Агрегация, пайплайны

Основные стадии агрегации:

\$match:

- Фильтрация документов на основе заданных критериев. Позволяет отобрать только те документы, которые соответствуют заданным условиям.

\$group:

- Группировка документов по определенному полю и выполнение агрегаций внутри каждой группы. Может включать операции, такие как суммирование, подсчет, нахождение среднего и другие.

\$project:

- Проекция или изменение формата вывода документов. Позволяет выбирать определенные поля, переименовывать их, добавлять новые или удалять существующие.

\$sort:

- Сортировка документов в результате агрегации. Позволяет упорядочивать результаты в заданном порядке (возрастающем или убывающем) на основе одного или нескольких полей.

\$limit:

- Ограничение числа документов, передаваемых в следующую стадию агрегации. Указывает максимальное количество результатов.

\$unwind:

- Разбивка массивов в документе, создавая для каждого элемента массива отдельный документ. Полезно при работе с массивами и необходимостью проведения агрегаций на каждом элементе массива.

\$lookup:



- Объединение данных из нескольких коллекций. Позволяет связывать документы в одной коллекции с документами в другой на основе заданных условий.

\$out:

- Записывает результаты агрегации в новую коллекцию. Полезно, когда требуется сохранить результаты для долгосрочного использования.

\$geoNear:

- Выполняет геолокационный запрос, возвращая документы в порядке увеличения расстояния от указанной точки. Полезно при работе с географическими данными.

\$bucket:

- Разделяет документы на группы, называемые "граблями" (buckets), на основе заданных границ. Удобно для создания гистограмм данных.

\$merge:

- Используется для объединения или сохранения результатов агрегации в указанное местоположение, обычно в новую коллекцию

\$sample:

- Для случайного выбора определенного количества документов

\$limit:

- Используется для ограничения количества документов, передаваемых в следующую стадию

\$facet:

- Выполнение нескольких независимых агрегаций параллельно и предоставление результатов в виде набора данных. Удобно для одновременного выполнения нескольких запросов и структурирования результатов.



Группировка

Рассчитать среднее количество просмотров и лайков для каждого исполнителя.

Unset

```
[
  { $group: { _id: "$Artist", avgViews: { $avg: "$Views" }, avgLikes: { $avg:
"$Likes" } } }
]
```

Проекция (Project)

Показать только основную информацию для каждого документа.

Unset

```
[
  { $project: { _id: 0, "Artist": 1, "Track": 1, "Album": 1, "Views": 1, "Likes": 1 } }
]
```

Сортировка

Отсортировать документы по количеству просмотров по убыванию.

Unset

```
[
  { $sort: { "Views": -1 } }
]
```

Разделение на группы

Сгруппировать треки по уровню танцевальности.

```
Unset
[
  {
    $bucket: {
      groupBy: "$Danceability",
      boundaries: [0, 0.5, 0.75, 1],
      default: "Other",
      output: { "tracks": { $push: "$Track" } }
    }
  }
]
```

Фасетирование

Фасетирование - позволяет разделить результаты запроса на несколько категорий, обработать их независимо и предоставить результаты в структурированном формате.

Разделить по типу альбома и подсчитать количество треков в каждом типе.

```
Unset
[
  { $facet: { "AlbumTypes": [ { $sortByCount: "$Album_type" } ] } }
]
```

Найти количество треков по типам альбомов и их среднюю энергию.

```
Unset
[
  {
    $facet: {
      "AlbumTypes": [
        { $sortByCount: "$Album_type" },
        { $project: { _id: 0, type: "$_id", count: 1 } }
      ],
    }
  }
]
```



```
"avgEnergy": [
  { $group: { _id: "$Album_type", avgEnergy: { $avg: "$Energy" } } },
  { $project: { _id: 0, type: "$_id", avgEnergy: 1 } }
]
}
}
]
```

- \$sortByCount используется для подсчета количества треков для каждого типа альбома.
- \$group используется для расчета средней энергии для каждого типа альбома.
- \$project используется для форматирования вывода и удаления полей _id, чтобы результат был более читаемым.

Результат будет содержать два набора данных - один с количеством треков по типам альбомов (AlbumTypes), и второй с средней энергией для каждого типа альбома (avgEnergy).