

2 Querying Data

- There are 3242 artists data in this sample;
- For each id, there are 6 overview attributes, which are artist, genre, genres, link, popularity, and songs. All 6 attributes are unique and existed for each artist;
- There is a nested document for each song in attribute "songs" . 5 fields named as SName (song name), SLink (song link), lyric and idiom can be found in each document.

/*Question 1 */

```
db.artistsnestedsongs.find({}).limit(1)
```


















Result:

| _id | Artist | Popularity | Link | Genre |
|--|---|---|---|--|
|  6183f336408... |  10000 Maniacs |  0.3 |  /10000-mania... |  Rock |

/*Question 2 */

```
db.artistsnestedsongs.find({Link:{$exists:true}},{Artist:1, Genre:1})
```





Result:

| _id | Artist | Genre |
|--|--|--|
|  6183f336408... |  10000 Maniacs |  Rock |
|  6183f336408... |  A Cruz Está V... |  Rock |
|  6183f336408... |  ACEIA |  Rock |
|  6183f336408... |  Acid Tree |  Rock |
|  6183f336408... |  Adam Lambert |  Pop |
|  6183f336408... |  Aborto Elétrico |  Rock |

/*Question 3 */

```
db.artistsnestedsongs.find({Artist:{$regex:"Stones"}},{Artist:1})
```

Result:

| _id | Artist |
|--|--|
|  6183f336408... |  12 Stones |
|  6183f336408... |  Rolling Stones |

/*Question 4 */

```
db.artistsnestedsongs.find({Artist:{$regex:"Stones"}},{Artist:1}).count()
```

Result: 2

/*Question 5 */

```
db.artistsnestedsongs.find({Popularity:{$gt:4}},{Artist:1})
```

Result:

| _id | Artist |
|----------------------------------|-------------------------------------|
| [id] 6183f336408... | [id] Aerosmith |
| [id] 6183f336408... | [id] AC/DC |
| [id] 6183f336408... | [id] 4 Non Blondes |
| [id] 6183f336408... | [id] Bring Me The... |
| [id] 6183f336408... | [id] Bruce Springsteen |

/*Question 6 */

```
db.artistsnestedsongs.find({}, {Popularity: 1}).sort({"Popularity": 1}).limit(1)
```

```
db.artistsnestedsongs.find({Popularity: 0}, {Artist: 1})
```

Result:

| _id | Artist |
|----------------------------------|------------------------------------|
| [id] 6183f336408... | [id] A Cruz Está V... |
| [id] 6183f336408... | [id] ACEIA |
| [id] 6183f336408... | [id] Acid Tree |
| [id] 6183f336408... | [id] Adrian Suirady |
| [id] 6183f336408... | [id] Abuse |
| [id] 6183f336408... | [id] Andrew W.K. |
| [id] 6183f336408... | [id] Andi Deris |

/*Question 7 */

```
db.artistsnestedsongs.find({$where: "this.Songs.length >= 10" }, {Artist: 1})
```

Result:

| _id | Artist |
|----------------------------------|------------------------------------|
| [id] 6183f336408... | [id] 10000 Maniacs |
| [id] 6183f336408... | [id] A Cruz Está V... |
| [id] 6183f336408... | [id] Adam Lambert |
| [id] 6183f336408... | [id] Aborto Elétrico |
| [id] 6183f336408... | [id] 311 |

/*Question 8 */

```
db.artistsnestedsongs.find({Songs: {$size: 0}}, {Artist: 1})
```

Result:

| _id | Artist |
|--|---|
|  6183f336408... |  ACEIA |
|  6183f336408... |  Los Cunhados |
|  6183f336408... |  Connexão 017 |
|  6183f336408... |  Duzzão |
|  6183f336408... |  Família 4 Vidas |

/*Question 9 */

Result:

/*Question 10 */

```
db.artistsnestedsongs.aggregate([
  { $limit: 40 },
  { $group: { _id: null, distinctValues: { $addToSet: "$Genres" } } },
  { $project: { _id: 0 } }
])
```

Result:

| _id | distinctValues |
|-----|---|
| |  [33 elements] |

/*Question 11 */

We use aggregation operation to return the distinct values of genres among 40 documents, with the following 3 stages specified as follows.

With the \$limit stage, we specify the maximum number of documents to pass along, and here we only process the set of genres of the first 40 documents in the collection.

In \$group stage, we use \$addToSet accumulator to compute the list of unique items in Genres among the 40 documents we process.

Using the \$project: { _id: 0 }, we suppress the _id field.

/*Question 12 */

```
db.artistsnestedsongs.aggregate([
  { $match: { Genres: { $regex: "Folk" } } },
  { $group: { _id: null, myCount: { $sum: 1 } } },
  { $project: { _id: 0 } }
])
```

Result: 322

/*Question 13 */

Number of songs by main genre of the artist

```
db.artistsnestedsongs.aggregate([
  { $group: { _id: "$Genre", myCount: { $sum: { "$size": "$Songs" } } } }
])
```

Result:

| _id | myCount |
|----------------|---------|
| "Rock" | 76624 |
| "Samba" | 15322 |
| "Hip Hop" | 30593 |
| "Sertanejo" | 35231 |
| "Funk Carioca" | 8292 |
| "Pop" | 61451 |

Number of songs by language:

```
db.artistsnestedsongs.aggregate([
  { $unwind: "$Songs" },
  { $group: { _id: "$Songs.Idiom", myCount: { $sum: 1 } } }
])
```

Result:

| _id | myCount |
|-------------|----------|
| "SUNDANESE" | 8.0 |
| "JAPANESE" | 2.0 |
| "ARABIC" | 2.0 |
| "BOSNIAN" | 1.0 |
| "ENGLISH" | 124230.0 |
| "SWAHILI" | 7.0 |