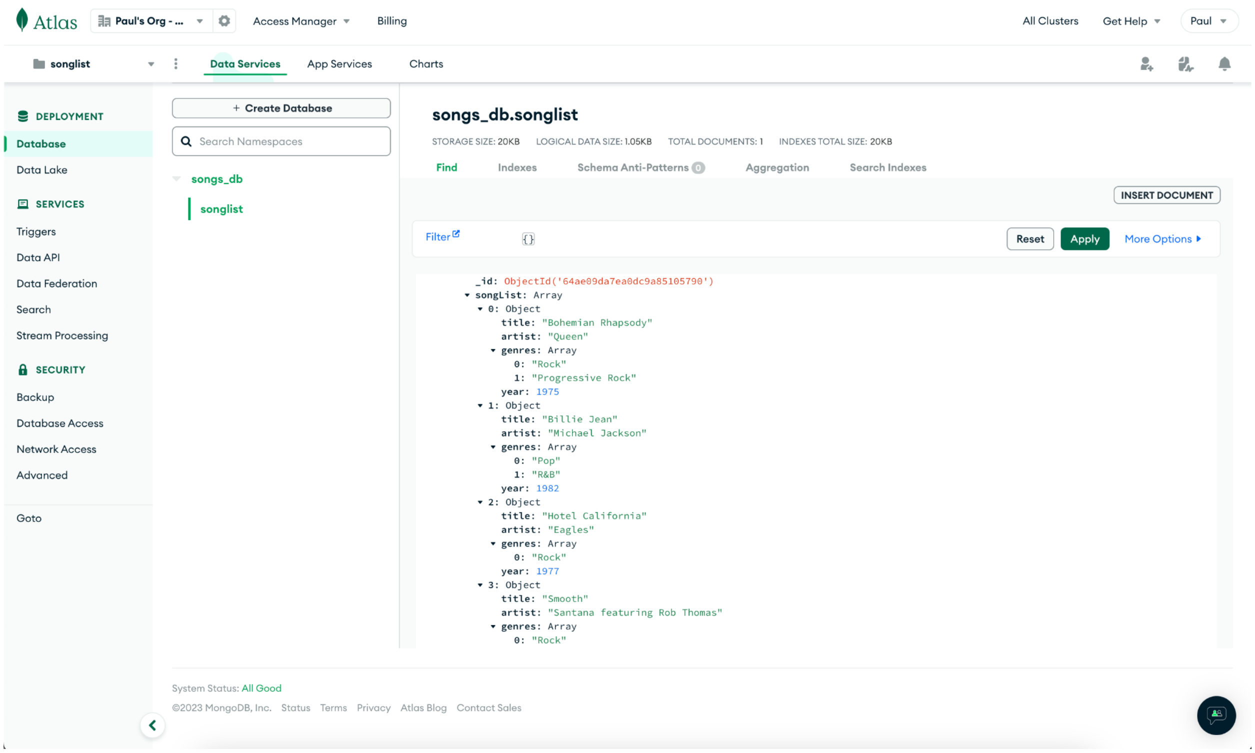
Unit3 PS2 - NoSQL
Paul Roche
Solutions

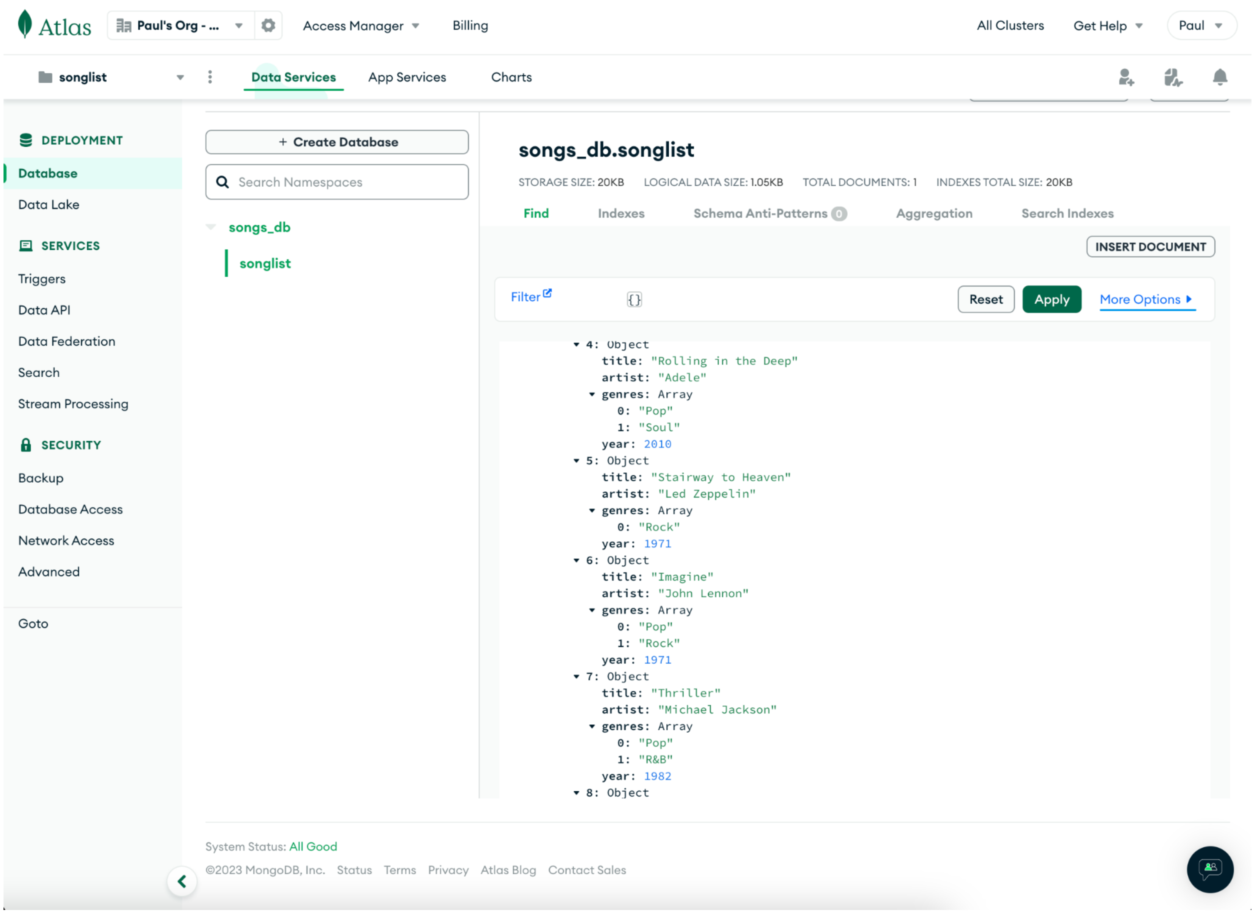
1. In MongoDb Atlas, create a MongoDB database/collection to store your song list (from the JSON and SQL assignments).
    a. Insert all data
    b. Show all data in the collection in MongoDB Atlas - take a screenshot (you may not be able to see the full list - show what you can) - screenshot fits most of it



songs_db.songlist

STORAGE SIZE: 20KB LOGICAL DATA SIZE: 1.05KB TOTAL DOCUMENTS: 1 INDEXES TOTAL SIZE: 20KB

```
  _id: ObjectId('64ae09da7ea0dc9a85105790')
▾ songList: Array
  ▾ 0: Object
      title: "Bohemian Rhapsody"
      artist: "Queen"
    ▾ genres: Array
        0: "Rock"
        1: "Progressive Rock"
      year: 1975
  ▾ 1: Object
      title: "Billie Jean"
      artist: "Michael Jackson"
    ▾ genres: Array
        0: "Pop"
        1: "R&B"
      year: 1982
  ▾ 2: Object
      title: "Hotel California"
      artist: "Eagles"
    ▾ genres: Array
        0: "Rock"
      year: 1977
  ▾ 3: Object
      title: "Smooth"
      artist: "Santana featuring Rob Thomas"
    ▾ genres: Array
        0: "Rock"
```



songs_db.songlist

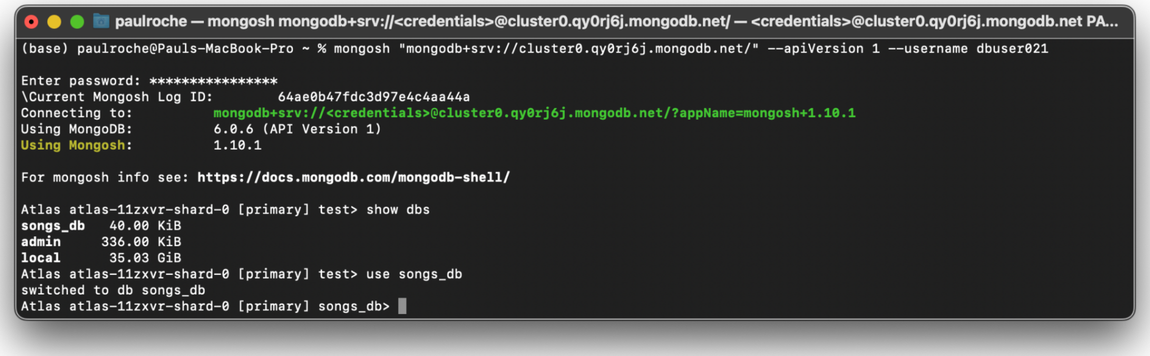
STORAGE SIZE: 20KB LOGICAL DATA SIZE: 1.05KB TOTAL DOCUMENTS: 1 INDEXES TOTAL SIZE: 20KB

```
▾ 4: Object
    title: "Rolling in the Deep"
    artist: "Adele"
  ▾ genres: Array
      0: "Pop"
      1: "Soul"
    year: 2010
▾ 5: Object
    title: "Stairway to Heaven"
    artist: "Led Zeppelin"
  ▾ genres: Array
      0: "Rock"
    year: 1971
▾ 6: Object
    title: "Imagine"
    artist: "John Lennon"
  ▾ genres: Array
      0: "Pop"
      1: "Rock"
    year: 1971
▾ 7: Object
    title: "Thriller"
    artist: "Michael Jackson"
  ▾ genres: Array
      0: "Pop"
      1: "R&B"
    year: 1982
▾ 8: Object
```

System Status: All Good

©2023 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

2. In the Mongo Shell, connect to your database and do the following (show a screen shot for each)
    a. select the database


paulroche — mongosh mongodb+srv://<credentials>@cluster0.qy0rj6j.mongodb.net/ — <credentials>@cluster0.qy0rj6j.mongodb.net PA...

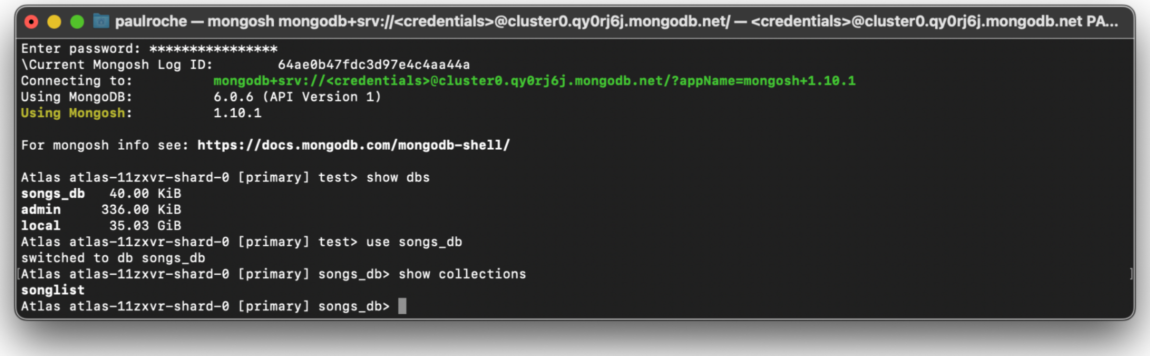
```
(base) paulroche@Pauls-MacBook-Pro ~ % mongosh "mongodb+srv://cluster0.qy0rj6j.mongodb.net/" --apiVersion 1 --username dbuser021

Enter password: ****************
\Current Mongosh Log ID:        64ae0b47fdc3d97e4c4aa44a
Connecting to:          mongodb+srv://<credentials>@cluster0.qy0rj6j.mongodb.net/?appName=mongosh+1.10.1
Using MongoDB:          6.0.6 (API Version 1)
Using Mongosh:          1.10.1

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

Atlas atlas-11zxvr-shard-0 [primary] test> show dbs
songs_db   40.00 KiB
admin     336.00 KiB
local      35.03 GiB
Atlas atlas-11zxvr-shard-0 [primary] test> use songs_db
switched to db songs_db
Atlas atlas-11zxvr-shard-0 [primary] songs_db> 
```

    b. show all collections in the database

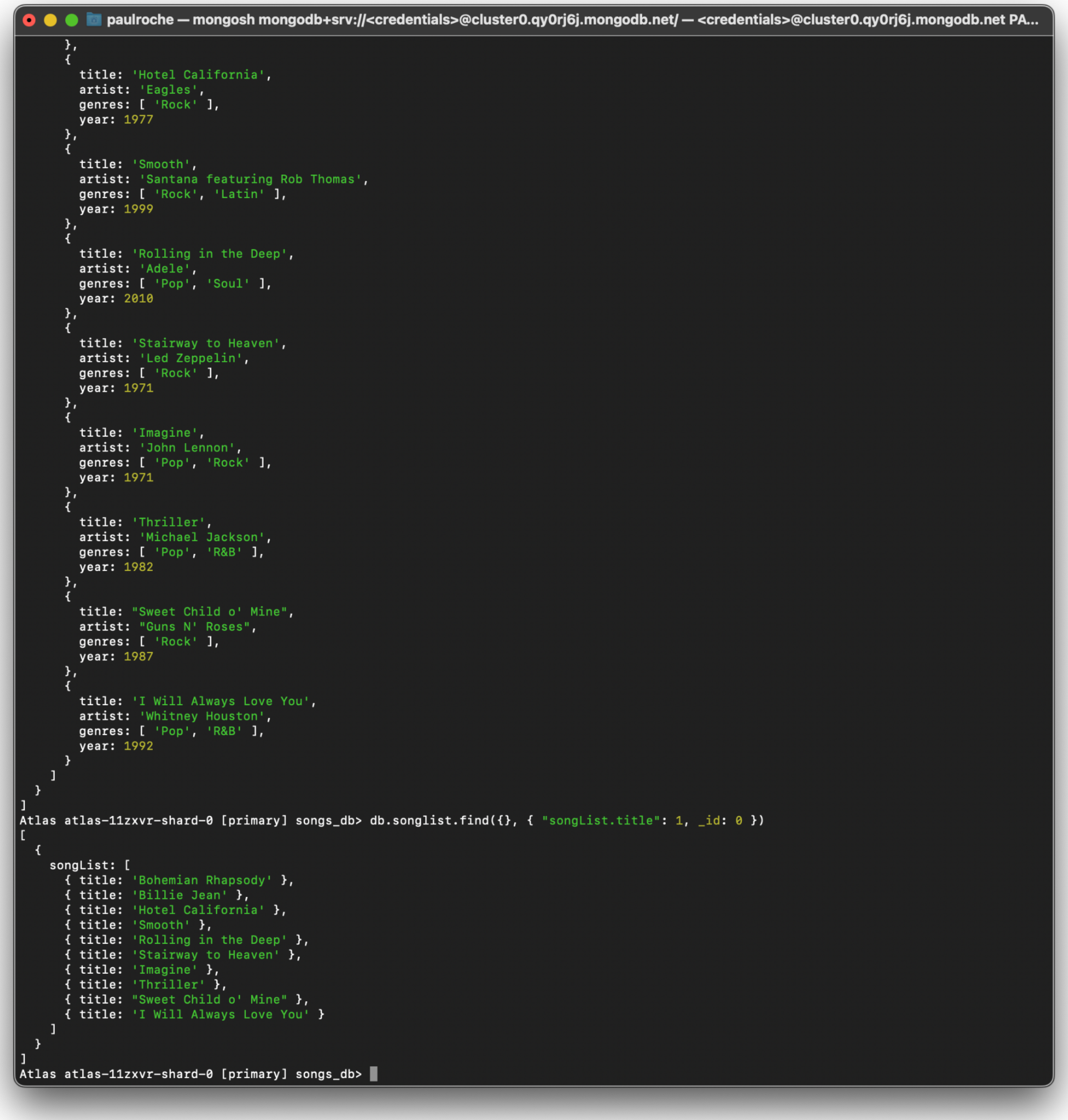

paulroche — mongosh mongodb+srv://<credentials>@cluster0.qy0rj6j.mongodb.net/ — <credentials>@cluster0.qy0rj6j.mongodb.net PA...

```
Enter password: ****************
\Current Mongosh Log ID:        64ae0b47fdc3d97e4c4aa44a
Connecting to:          mongodb+srv://<credentials>@cluster0.qy0rj6j.mongodb.net/?appName=mongosh+1.10.1
Using MongoDB:          6.0.6 (API Version 1)
Using Mongosh:          1.10.1

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

Atlas atlas-11zxvr-shard-0 [primary] test> show dbs
songs_db   40.00 KiB
admin     336.00 KiB
local      35.03 GiB
Atlas atlas-11zxvr-shard-0 [primary] test> use songs_db
switched to db songs_db
Atlas atlas-11zxvr-shard-0 [primary] songs_db> show collections
songlist
Atlas atlas-11zxvr-shard-0 [primary] songs_db> 
```

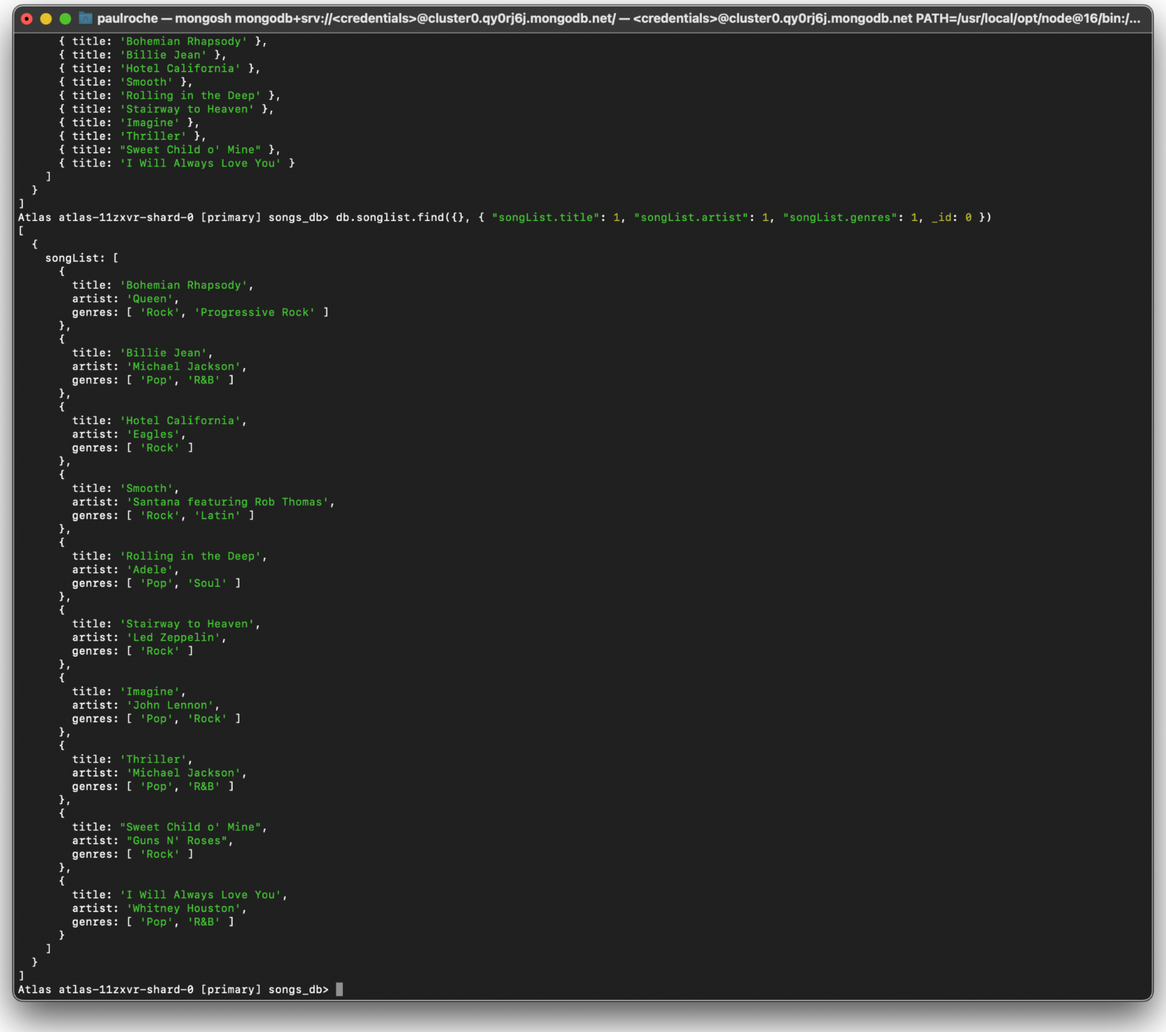
c. show all song titles (titles only)



paulroche — mongosh mongodb+srv://<credentials>@cluster0.qy0rj6j.mongodb.net/ — <credentials>@cluster0.qy0rj6j.mongodb.net PA...

```
      },
      {
        title: 'Hotel California',
        artist: 'Eagles',
        genres: [ 'Rock' ],
        year: 1977
      },
      {
        title: 'Smooth',
        artist: 'Santana featuring Rob Thomas',
        genres: [ 'Rock', 'Latin' ],
        year: 1999
      },
      {
        title: 'Rolling in the Deep',
        artist: 'Adele',
        genres: [ 'Pop', 'Soul' ],
        year: 2010
      },
      {
        title: 'Stairway to Heaven',
        artist: 'Led Zeppelin',
        genres: [ 'Rock' ],
        year: 1971
      },
      {
        title: 'Imagine',
        artist: 'John Lennon',
        genres: [ 'Pop', 'Rock' ],
        year: 1971
      },
      {
        title: 'Thriller',
        artist: 'Michael Jackson',
        genres: [ 'Pop', 'R&B' ],
        year: 1982
      },
      {
        title: "Sweet Child o' Mine",
        artist: "Guns N' Roses",
        genres: [ 'Rock' ],
        year: 1987
      },
      {
        title: 'I Will Always Love You',
        artist: 'Whitney Houston',
        genres: [ 'Pop', 'R&B' ],
        year: 1992
      }
    ]
  }
]
Atlas atlas-11zxvr-shard-0 [primary] songs_db> db.songlist.find({}, { "songList.title": 1, _id: 0 })
[
  {
    songList: [
      { title: 'Bohemian Rhapsody' },
      { title: 'Billie Jean' },
      { title: 'Hotel California' },
      { title: 'Smooth' },
      { title: 'Rolling in the Deep' },
      { title: 'Stairway to Heaven' },
      { title: 'Imagine' },
      { title: 'Thriller' },
      { title: "Sweet Child o' Mine" },
      { title: 'I Will Always Love You' }
    ]
  }
]
Atlas atlas-11zxvr-shard-0 [primary] songs_db> 
```

d. show all titles, artist and all genres that correspond to each



paulroche — mongosh mongodb+srv://<credentials>@cluster0.qy0rj6j.mongodb.net/ — <credentials>@cluster0.qy0rj6j.mongodb.net PATH=/usr/local/opt/node@16/bin:/...

```
      { title: 'Bohemian Rhapsody' },
      { title: 'Billie Jean' },
      { title: 'Hotel California' },
      { title: 'Smooth' },
      { title: 'Rolling in the Deep' },
      { title: 'Stairway to Heaven' },
      { title: 'Imagine' },
      { title: 'Thriller' },
      { title: "Sweet Child o' Mine" },
      { title: 'I Will Always Love You' }
    ]
  }
]
Atlas atlas-11zxvr-shard-0 [primary] songs_db> db.songlist.find({}, { "songList.title": 1, "songList.artist": 1, "songList.genres": 1, _id: 0 })
[
  {
    songList: [
      {
        title: 'Bohemian Rhapsody',
        artist: 'Queen',
        genres: [ 'Rock', 'Progressive Rock' ]
      },
      {
        title: 'Billie Jean',
        artist: 'Michael Jackson',
        genres: [ 'Pop', 'R&B' ]
      },
      {
        title: 'Hotel California',
        artist: 'Eagles',
        genres: [ 'Rock' ]
      },
      {
        title: 'Smooth',
        artist: 'Santana featuring Rob Thomas',
        genres: [ 'Rock', 'Latin' ]
      },
      {
        title: 'Rolling in the Deep',
        artist: 'Adele',
        genres: [ 'Pop', 'Soul' ]
      },
      {
        title: 'Stairway to Heaven',
        artist: 'Led Zeppelin',
        genres: [ 'Rock' ]
      },
      {
        title: 'Imagine',
        artist: 'John Lennon',
        genres: [ 'Pop', 'Rock' ]
      },
      {
        title: 'Thriller',
        artist: 'Michael Jackson',
        genres: [ 'Pop', 'R&B' ]
      },
      {
        title: "Sweet Child o' Mine",
        artist: "Guns N' Roses",
        genres: [ 'Rock' ]
      },
      {
        title: 'I Will Always Love You',
        artist: 'Whitney Houston',
        genres: [ 'Pop', 'R&B' ]
      }
    ]
  }
]
Atlas atlas-11zxvr-shard-0 [primary] songs_db> 
```

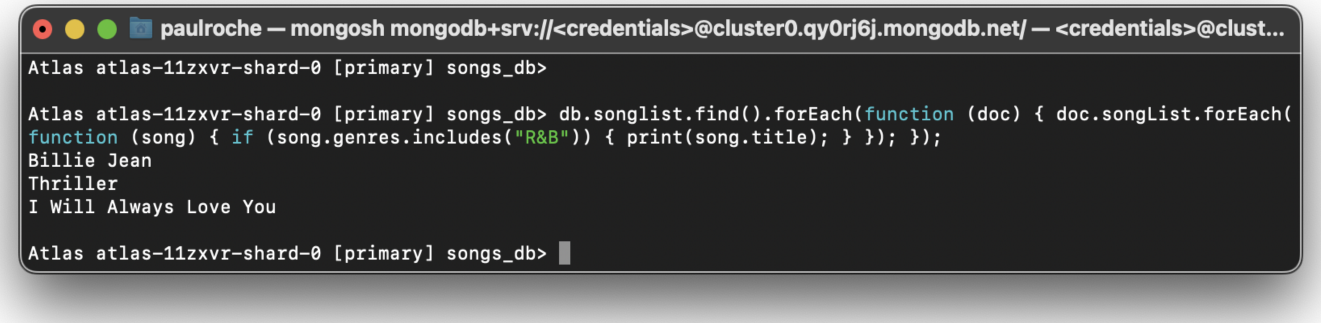
e. show all titles for a particular genre

NOTE: This question had multiple parts. More specifically, it required that I use the find() on two nested loops. For each song in the songlist, I had to first loop through every song's genres array to see if there was the "R&B" genre. If it was there, I'd print the song title.

Ref: https://www.mongodb.com/docs/manual/reference/method/cursor.forEach/

Query:

```
db.songlist.find().forEach(function (doc) {
  doc.songList.forEach(function(song) {
    if (song.genres.includes("R&B")) {
      print(song.title);
    }
  });
});
```



paulroche — mongosh mongodb+srv://<credentials>@cluster0.qy0rj6j.mongodb.net/ — <credentials>@clust...

```
Atlas atlas-11zxvr-shard-0 [primary] songs_db>

Atlas atlas-11zxvr-shard-0 [primary] songs_db> db.songlist.find().forEach(function (doc) { doc.songList.forEach(
function (song) { if (song.genres.includes("R&B")) { print(song.title); } }); });
Billie Jean
Thriller
I Will Always Love You

Atlas atlas-11zxvr-shard-0 [primary] songs_db>
```