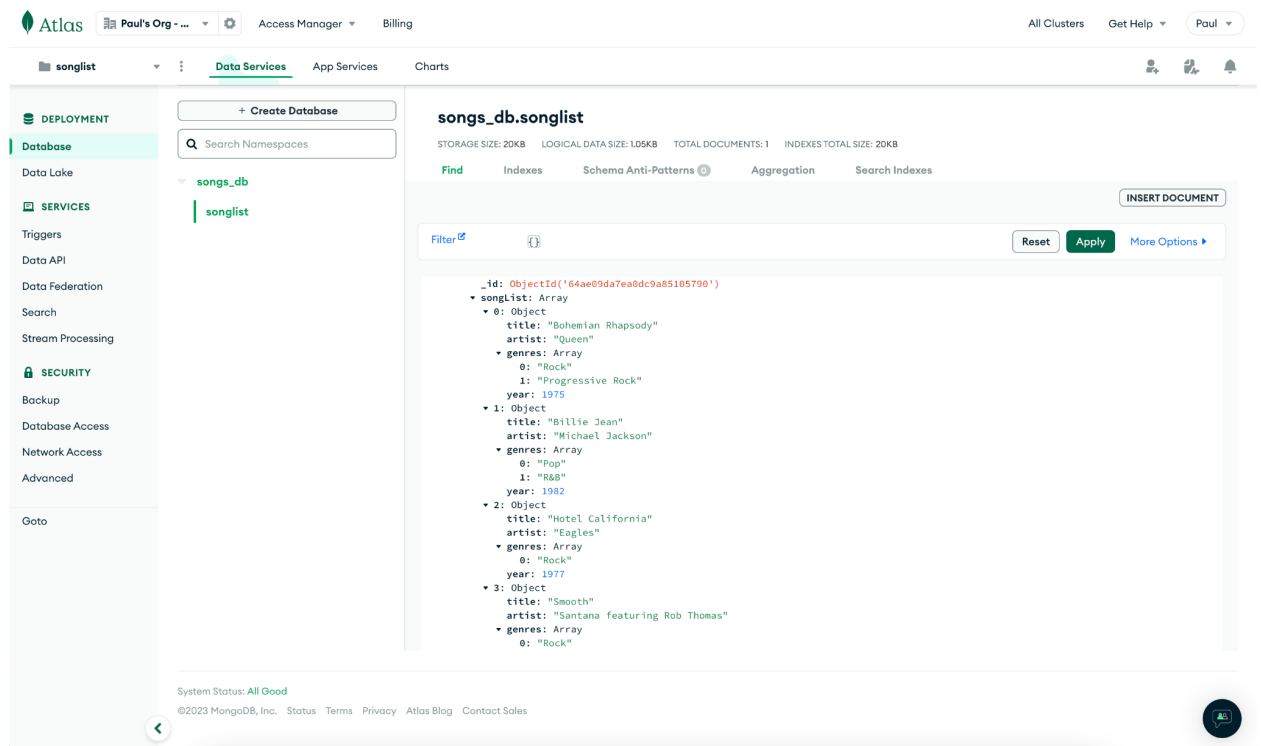


Unit3 PS2 - NoSQL

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



The screenshot shows the MongoDB Atlas web interface. The top navigation bar includes the Atlas logo, a dropdown for 'Paul's Org', and links for 'Access Manager' and 'Billing'. On the right, there are links for 'All Clusters', 'Get Help', and a user profile 'Paul'. The main interface is divided into a left sidebar, a central navigation pane, and a main content area.

Left Sidebar: Contains sections for 'DEPLOYMENT' (Database, Data Lake, SERVICES, Triggers, Data API, Data Federation, Search, Stream Processing) and 'SECURITY' (Backup, Database Access, Network Access, Advanced, Goto).

Central Navigation Pane: Shows a tree view with 'songs_db' expanded, revealing the 'songlist' collection. Above this is a search bar for namespaces and a '+ Create Database' button.

Main Content Area: Displays the 'songs_db.songlist' collection. It shows metadata: 'STORAGE SIZE: 20KB', 'LOGICAL DATA SIZE: 1.05KB', 'TOTAL DOCUMENTS: 1', and 'INDEXES TOTAL SIZE: 20KB'. Below this are tabs for 'Find', 'Indexes', 'Schema Anti-Patterns', 'Aggregation', and 'Search Indexes'. The 'Find' tab is active, showing a list of documents. Each document is an object with fields: 'title', 'artist', 'genres' (an array), and 'year'. The documents are:

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

 At the top right of the document list is an 'INSERT DOCUMENT' button. Below the list are 'Filter', 'Reset', 'Apply', and 'More Options' buttons.

Footer: Shows 'System Status: All Good', copyright information '©2023 MongoDB, Inc.', and links for 'Status', 'Terms', 'Privacy', 'Atlas Blog', and '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

```
pauroche — 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> █
```