## Requirements

For this hands on, you will be working through several real-life scenarios within new collections. This Hands-On is structured into two parts, and each part will ask you to run multiple queries. After each query, please take a screenshot and add it to a text document (or an equivalent) and name this file Lesson5handson. This way, you will be able to submit your answers to each part all at once.

## Part 1

You have just been hired at a startup company. They currently only have ten employees, but they need to be included in the database. So far, they have only been tracked within an excel sheet. Your boss would like you to create a new collection in Atlas named employees. Look at the following data and the notes listed below before inserting any data:



Notes:

* The Birthday field should have a data type of Date.
* The Position Name, Remote, and Full Time fields should be within an embedded document called position.
* Remote and Full Time fields should have boolean values.

Query:

db.employees.insertMany([

{

Name: "Alison Davidson",

Birthday: "04/05/1975",

Address: "874 W. Oak Place",

City: "Gary",

State: "Indiana",

Position: {PositionName: "Customer Support", Remote: true, FullTime: true}

},

{

Name: "Henry Chapelton",

Birthday: "09/29/1980",

Address: "9324 E. Vista Way",

City: "Tempe",

State: "Arizona",

Position: {PositionName: "Customer Support", Remote: false, FullTime: true}

},

{

Name: "Alex Miller",

Birthday: "11/22/1983",

Address: "244 Price Road",

City: "Mesa",

State: "Arizona",

Position: {PositionName: "Customer Support", Remote: false, FullTime: false}

},

{

Name: "Carly Nielson",

Birthday: "08/04/1987",

Address: "678 W. Westward Road",

City: "Phoenix",

State: "Arizona",

Position: {PositionName: "Office Manager", Remote: false, FullTime: true}

},

{

Name: "Tom Talbot",

Birthday: "12/30/1989",

Address: "12 Oakland Way",

City: "Chandler",

State: "Arizona",

Position: {PositionName: "Inventory Manager", Remote: false, FullTime: true}

},

{

Name: "Mary Crawley",

Birthday: "07/06/1980",

Address: "1010 Granite Way",

City: "Charlotte",

State: "North Carolina",

Position: {PositionName: "Human Resources", Remote: true, FullTime: true}

},

{

Name: "Daisy Baxter",

Birthday: "09/09/1987",

Address: "990 E. 84th St.",

City: "Tempe",

State: "Arizona",

Position: {PositionName: "CEO", Remote: false, FullTime: true}

},

{

Name: "William Coyle",

Birthday: "10/11/1991",

Address: "944 W. 16th St.",

City: "Phoenix",

State: "Arizona",

Position: {PositionName: "Intern", Remote: false, FullTime: false}

},

{

Name: "Edith Bates",

Birthday: "07/28/1990",

Address: "7 E. 20th Pl.",

City: "Chandler",

State: "Arizona",

Position: {PositionName: "Customer Support", Remote: false, FullTime: true}

},

{

Name: "Gwen Harding",

Birthday: "10/11/1986",

Address: "234 W. 48th. St.",

City: "Phoenix",

State: "Arizona",

Position: {PositionName: "Office Assistant", Remote: false, FullTime: true}

}

])

Results:



It's been about a month since you have inserted all employees into the database. There have been a couple of changes to the company. The CEO decided that he no longer wants remote employees, so they have transferred the remote employees and they are now living in Arizona. Alison Davidson now lives at 777 E. 1st St. # 120 Tempe, AZ and Mary Crawley now lives at 8322 W. Vista Pl. Scottsdale, AZ. Since all employees now all live in Arizona, there is no need to have a field named "state" within this collection, so please remove it. Lastly, they would like very efficient searching using the "position" field (remember that field includes a document with three other fields).

Queries:

Query remote employees:

db.employees.find({'Position.Remote': true}).pretty()

Update Employee Addresses:

db.employees.updateOne(

{Name: "Alison Davidson"},

{$set: {Address: "777 E. 1st St. # 120", City: "Tempe", State: "Arizona"}}

)

db.employees.updateOne(

{Name: "Mary Crawley"},

{$set: {Address: "8322 W. Vista Pl.", City: " Scottsdale", State: "Arizona"}}

)

Remove the ‘State’ field:

db.employees.updateMany(

{},

{$set: {'Position.Remote': false}}, {$set: {State: ""}}

)

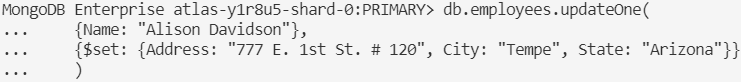
Index:

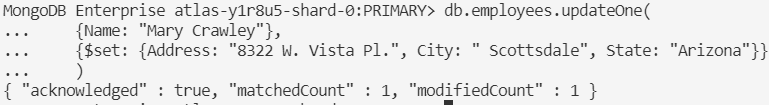
db.employees.createIndex({Position: 1})

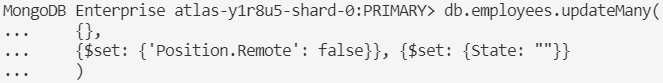
Results:

\* Final MongoDB Atlas output in appendix.











## Part 2

You are currently working for a company who wants to build an app similar to Spotify. Below is a list of data for different songs. Please insert this data into a new collection named songs.

Notes:

* The artist, album, and releaseYear fields should be an embedded document named details.
* Be sure that the songId and releaseYear fields have a type of number.

Query:

db.songs.insertMany([

{

songId: 1,

title: "Girls Just Want To Have Fun",

details: {artist: "Cyndi Lauper", album: "She's So Unusual", releaseYear: 1983}

},

{

songId: 2,

title: "Hips Don't Lie",

details: {artist: "Shakira feat. Wyclef Jean", album: "Oral Fixation Vol. 2", releaseYear: 2006}

},

{

songId: 3,

title: "Poker Face",

details: {artist: "Lady Gaga", album: "The Fame", releaseYear: 2008}

},

{

songId: 4,

title: "Wannabe",

details: {artist: "Spice Girls", album: "Spice", releaseYear: 1996}

},

{

songId: 5,

title: "California Gurls",

details: {artist: "Katy Perry feat. Snoop Dogg", album: "Teenage Dream", releaseYear: 2010}

},

{

songId: 6,

title: "Bye, Bye, Bye",

details: {artist: "NSYNC", album: "No Strings Attached", releaseYear: 2000}

},

{

songId: 7,

title: "I Will Always Love You",

details: {artist: "Whitney Houston", album: "I Will Always Love You: The Best of Whitney Houston", releaseYear: 2012}

},

{

songId: 8,

title: "Baby One More Time",

details: {artist: "Britney Spears", album: "Baby One More Time", releaseYear: 1999}

},

{

songId: 9,

title: "Vogue",

details: {artist: "Madonna", album: "I'm Breathless", releaseYear: 1990}

},

{

songId: 10,

title: "Rolling in the Deep",

details: {artist: "Adele", album: "21", releaseYear: 2011}

},

{

songId: 11,

title: "1234",

details: {artist: "Feist", album: "The Reminder", releaseYear: 2007}

},

{

songId: 12,

title: "Elastic Heart",

details: {artist: "Sia", album: "The Hunger Games: Catching Fire Soundtrack", releaseYear: 2015}

},

{

songId: 13,

title: "Oops! I Did It Again",

details: {artist: "Britney Spears", album: "Oops! I Did It Again", releaseYear: 2000}

},

{

songId: 14,

title: "Bad Romance",

details: {artist: "Lady Gaga", album: "The Fame Monster", releaseYear: 2009}

},

{

songId: 15,

title: "Lose Control",

details: {artist: "Missy Elliot", album: "The Cookbook", releaseYear: 2005}

},

{

songId: 16,

title: "U Can't Touch This",

details: {artist: "MC Hammer", album: "Please Hammer, Don't Hurt 'Em", releaseYear: 1990}

},

{

songId: 17,

title: "Thriller",

details: {artist: "Michael Jackson", album: "Thriller", releaseYear: 1982}

},

{

songId: 18,

title: "Single Ladies",

details: {artist: "Beyonce", album: "I am... Sasha Fierce", releaseYear: 2008}

},

{

songId: 19,

title: "Rhythm Nation",

details: {artist: "Janet Jackson", album: "Janet Jackson's Rhythm Nation 1814", releaseYear: 1989}

}

])

Results:



Next, your company has run into some things they would like to be changed within the database:

* The title field needs to be renamed to songTitle, so it is clearer to the developers working with the data.
* They would like to have the artist field to be outside the details document but the album and releaseYear should stay within that document.

Queries:

db.songs.updateMany({}, {$rename: {"title": "songTitle"}})

db.songs.updateMany({}, {$rename: {"details.artist": "artist"}})

Results:

\* Final MongoDB Atlas output in appendix.





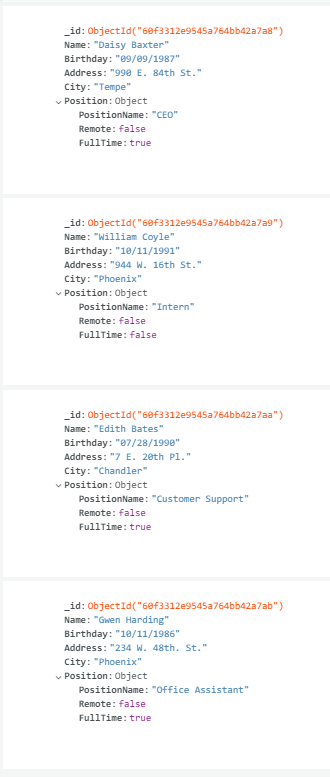
## Appendix

### Part 1 Final Output









### Part 2 Final Output







