T.C. İSTANBUL KÜLTÜR ÜNİVERSİTESİ

COM5041 - DATABASE DESIGN AND DEVELOPMENT

LAB 11 – Working with MongoDB

After completing this Lab, you will be able to

- Install the MongoDB.
- Work with MongoDB Compass.
- Create a new collection.
- Insert new document.
- Filter and navigate the data.
- Create a multi-stage aggregation pipeline.
- Visualize the collection's schema using the schema visualization tool.

PROCEDURE 1 – Installation of MongoDB

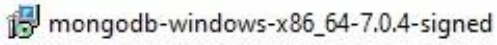
We will use the manual to install MongoDB 5.0 Community Edition on Windows using the default installation wizard. This manual installs MongoDB 5.0 Community Edition. You can get help from MongoDB's website to install a different version of MongoDB Community.

MongoDB 5.0 Community Edition supports the following 64-bit versions of Windows on x86_64 architecture: Windows Server 2019, Windows 10 / Windows Server 2016. MongoDB only supports the 64-bit versions of these platforms.

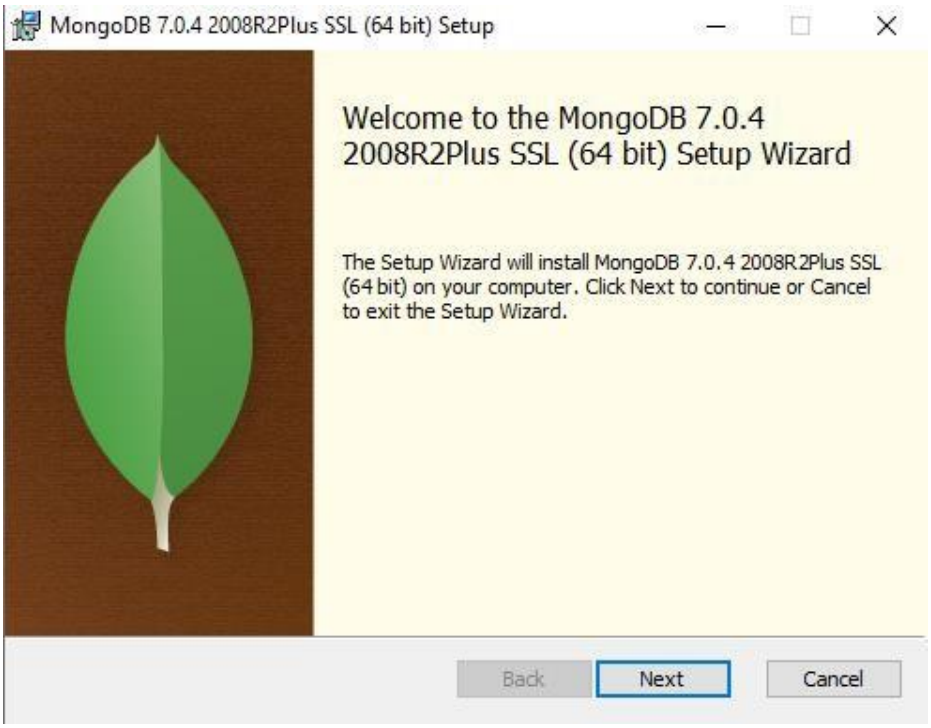
Step 1 - Open the MongoDB website. Go to https://www.mongodb.com/ in your computer's Internet browser. This is the website from which you'll download the MongoDB setup file.

- Click Products | Community Edition | Community Server | Select Package| Version can stay the current(7.0.4) | In the Platform dropdown, select Windows |In the Package dropdown, select msi | Click Download.

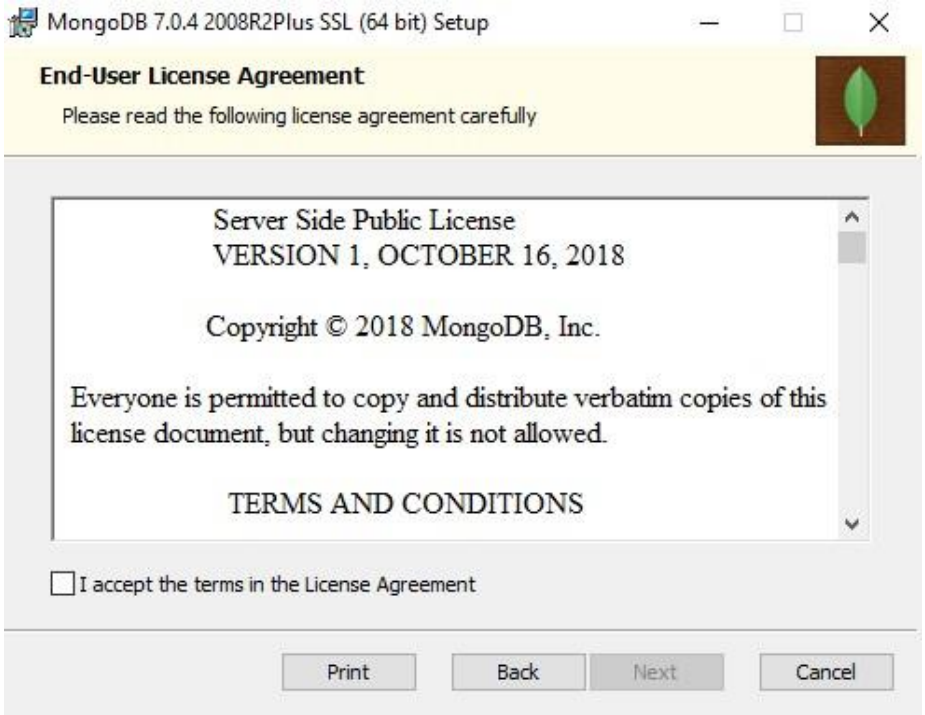
Step 2 - Open the mongodb-windows-x86_64-5.0.5-signed.msi EXE file. Go to the location to which the EXE file downloaded and double-click the file. Doing so will open the MongoDB installation window.



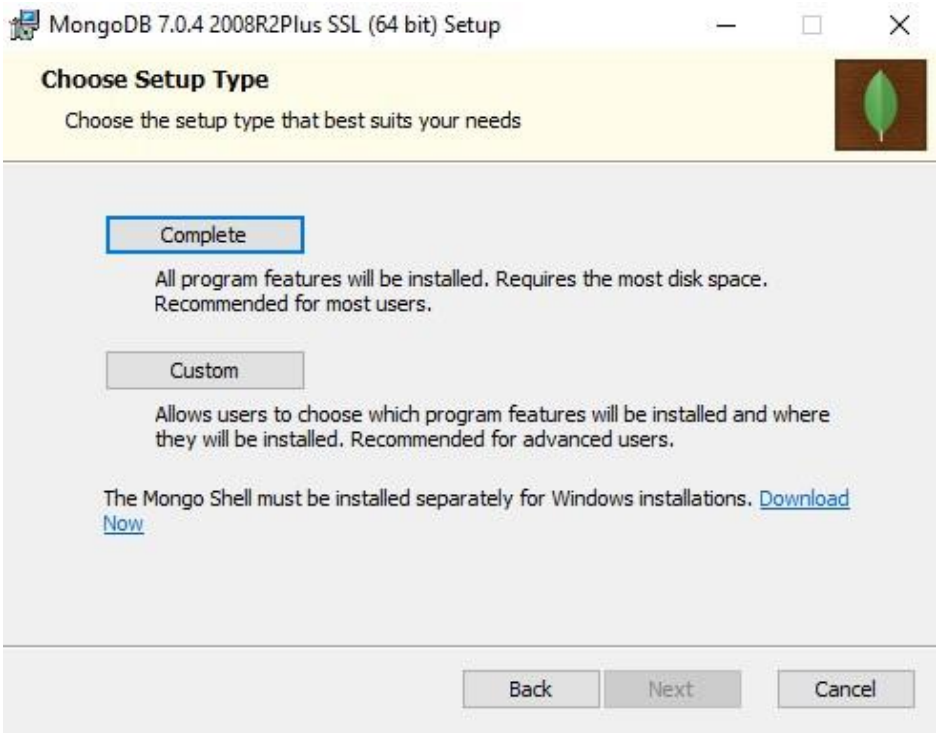
Step 3 - In the installation wizard that opens, you will see the MongoDB Community Edition installation wizard. Click Next.



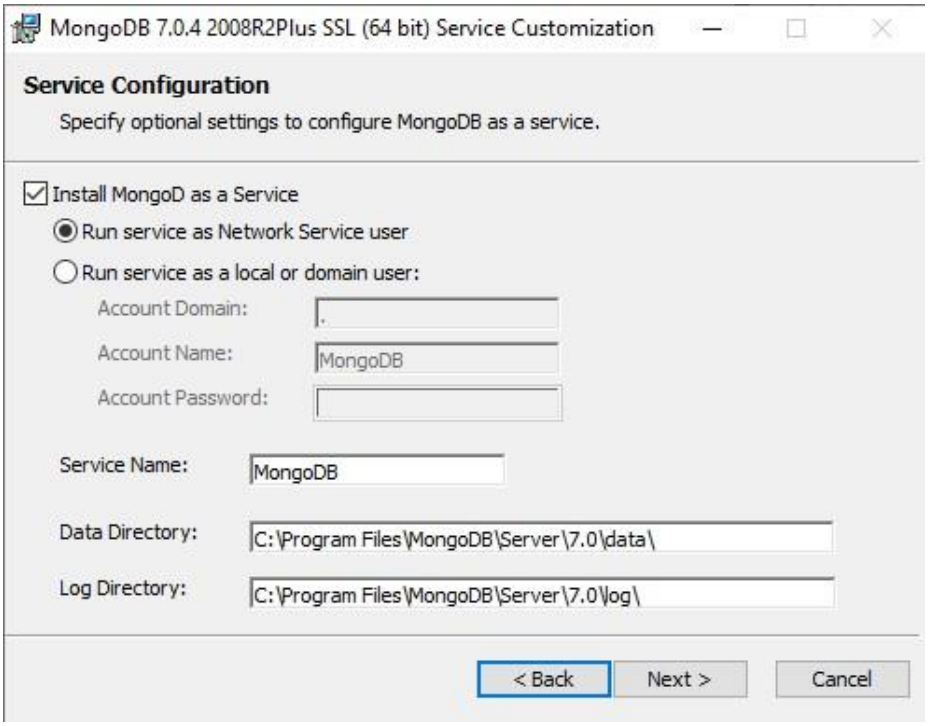
Step 4 – You should now see the license terms. Review these if you'd like and click Accept. Click Next.



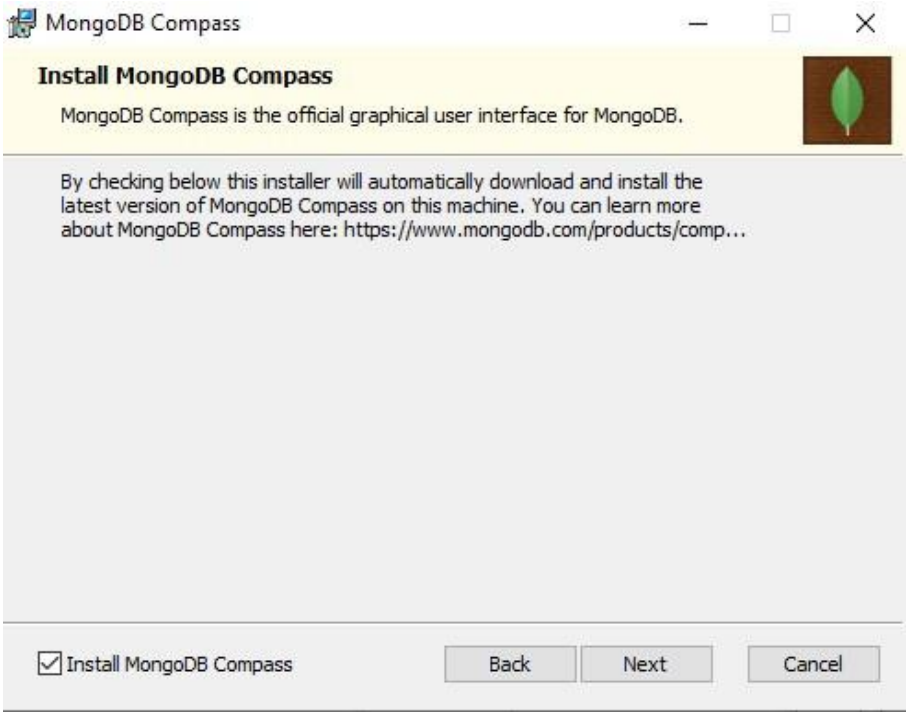
Step 5 - You will see the “Choose Setup Type” page. You can choose either the Complete (recommended for most users) or Custom setup type. The Complete setup option installs MongoDB and the MongoDB tools to the default location. The Custom setup option allows you to specify which executables are installed and where. You choose the “Complete” as setup type and click Next.



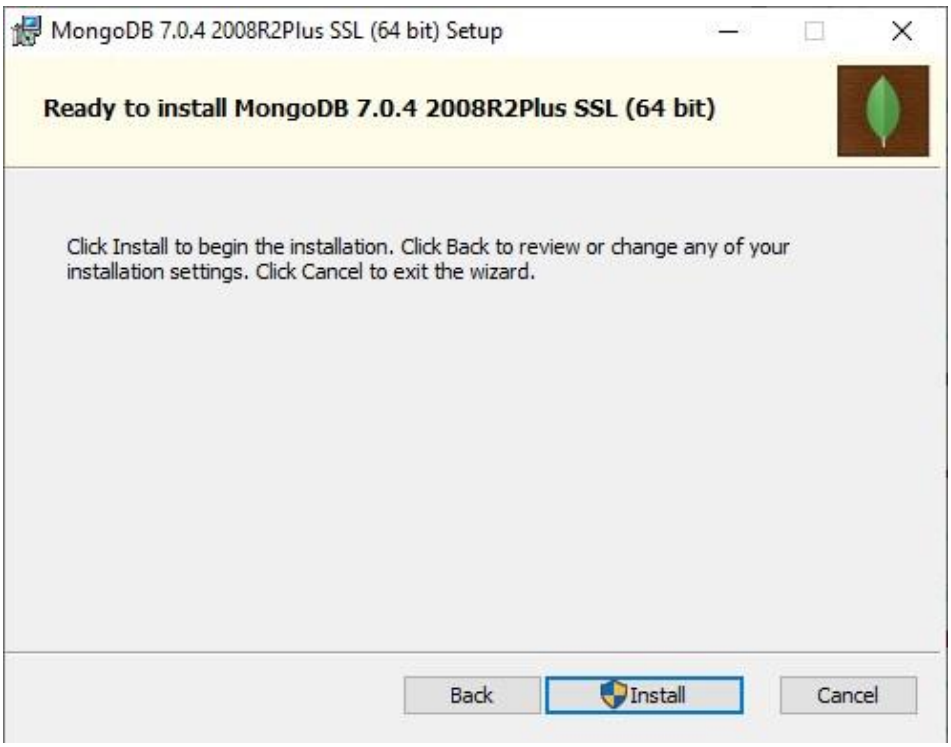
Step 6 – In the following page, you select “Install MongoD as a Service” to configure MongoDB as a service. Then, you select “Run service as Network Service user” and the other features as Default.



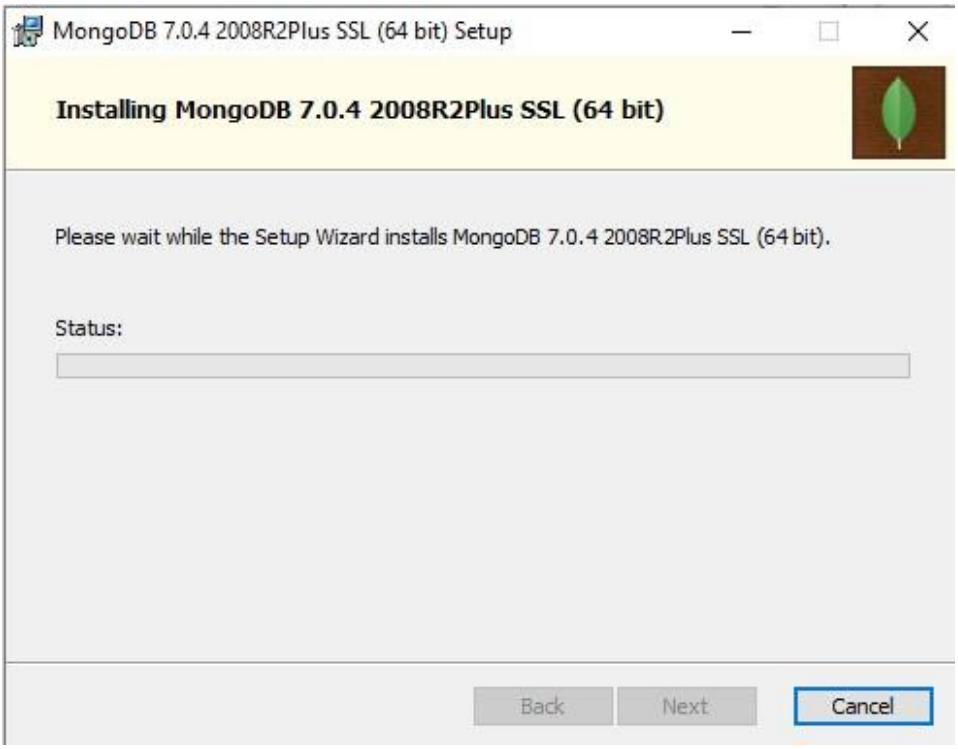
Step 7 – In the following page, you will see the “Install MongoDB Compass” page. To have the wizard install MongoDB Compass, select Install MongoDB Compass (Default). Then, click Next.



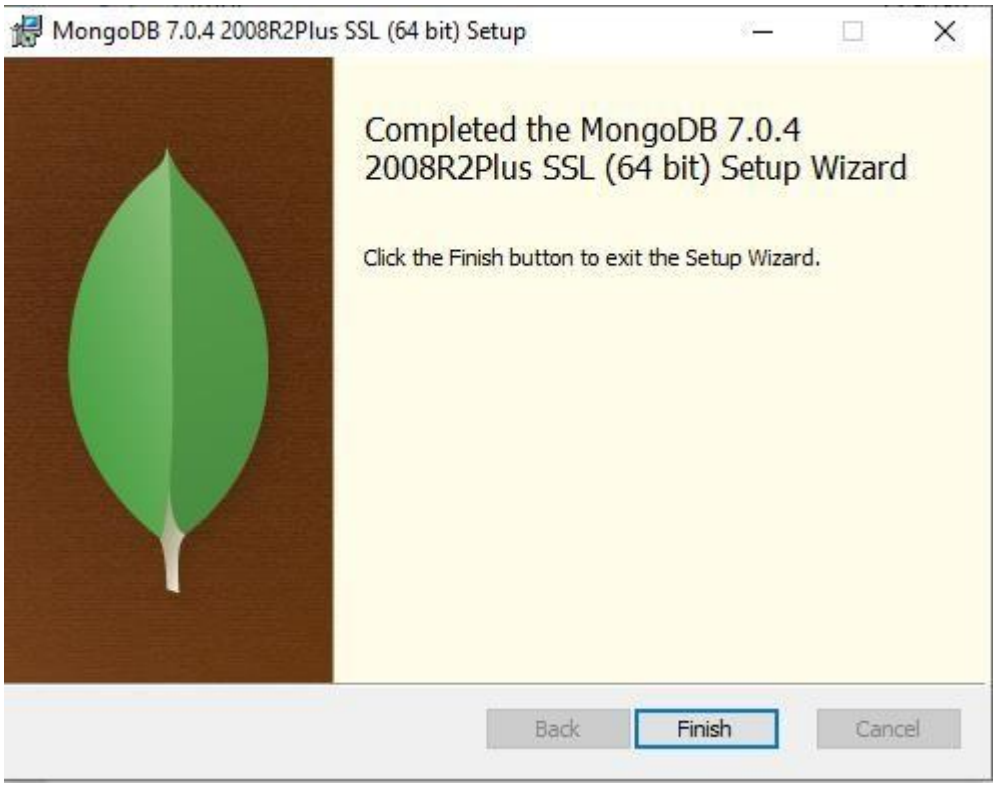
Step 8 – Click Install to begin installation.



Step 9 – Installation will begin. Below screen will show the copying new files progress.



Step 10 – Once Completed, setup will show the below screen with "Setup Completed" message. Click Finish.



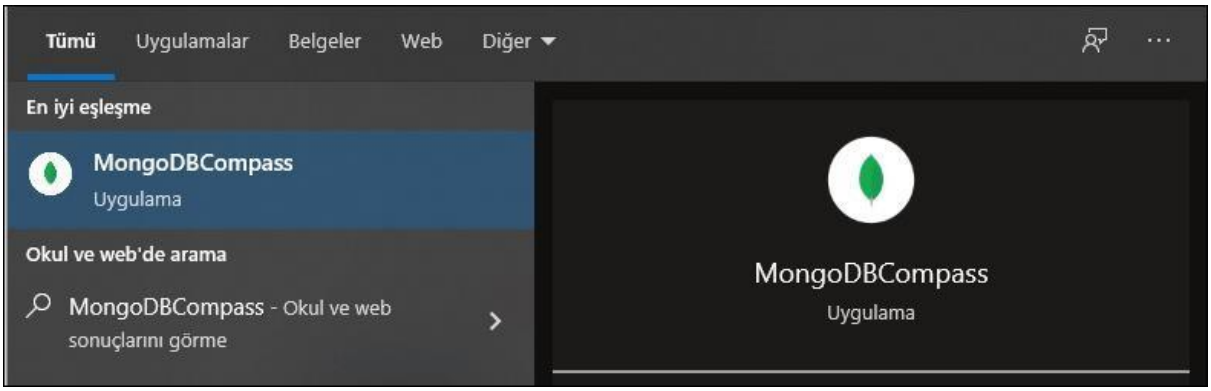
PROCEDURE 2 – Working with MongoDB

NoSQL databases are non-tabular databases and store data differently than relational tables. NoSQL databases come in a variety of types based on their data model. The main types are document, keyvalue, wide-column, and graph. They provide flexible schemas and scale easily with large amounts of data and high user loads.

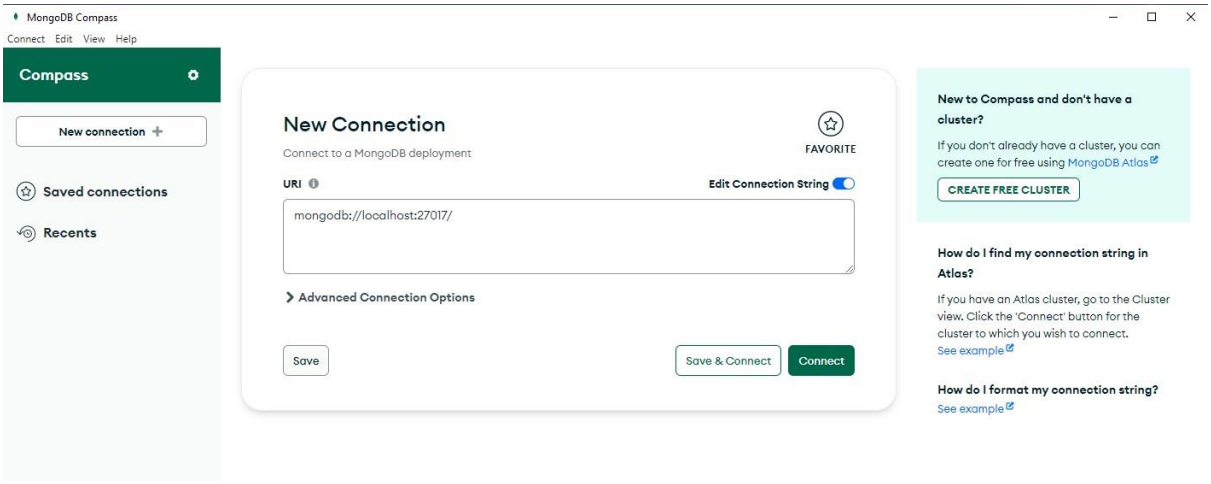
MongoDB is a NoSQL-based database that stores data as documents in JSON format. The MongoDB shell allows you to access a database as long as you already have access to the server on which MongoDB is running. However, a command line interface isn't always ideal for working with a database, as it may not be clear how one can find or analyze their data. Some may find it helpful to instead use a visual tool to view, manipulate, and analyze their data. To this end, MongoDB Compass is an official graphical user interface. With MongoDB Compass, you can access most of the features the MongoDB database engine offers through an intuitive visual display. You can glance through databases, collections and individual documents, interactively create queries, manipulate existing documents, and design aggregation pipelines through a dedicated interface.

Step 1 - We will start using MongoDB Compass.

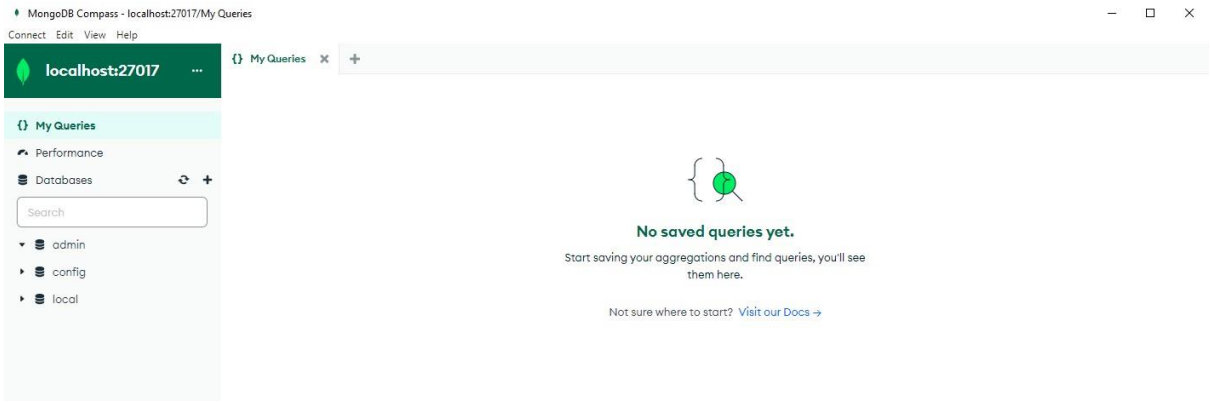
- To open MongoDB Compass, click Start Menu| All Programs | MongoDB Compass.



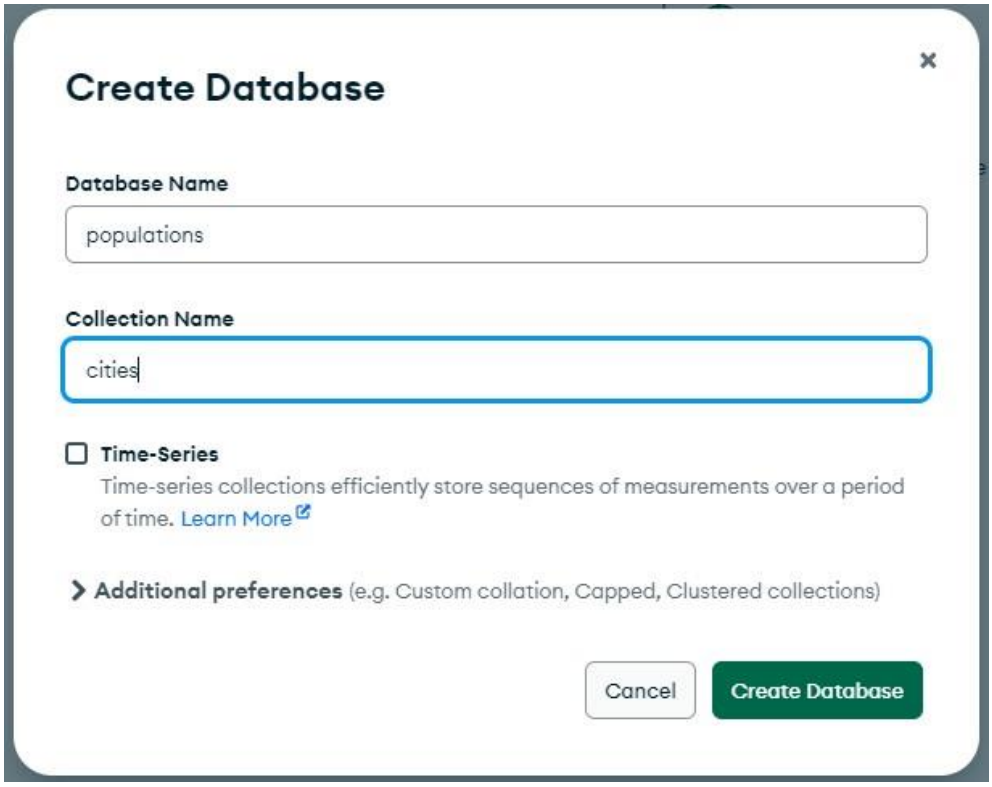
- Compass allows you to connect using a connection string that is a single line of text containing all necessary database connection information. You must click Connect button in order to connect to the MongoDB Server.



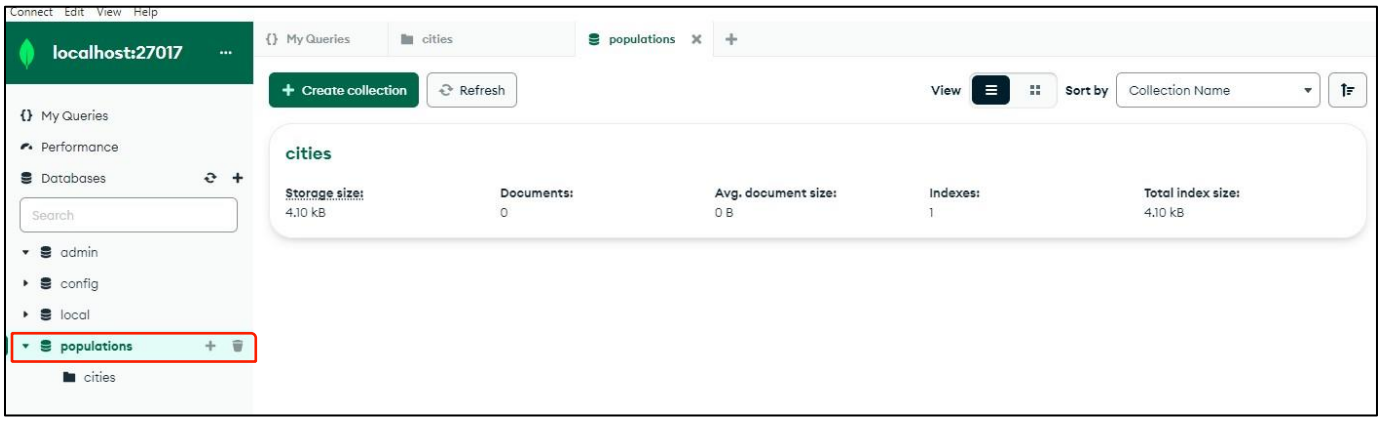
Step 2 - After clicking the Connect button, Compass will attempt to connect to the MongoDB instance. If it succeeds, you'll be taken to the Home screen showing the list of all the databases on the instance.

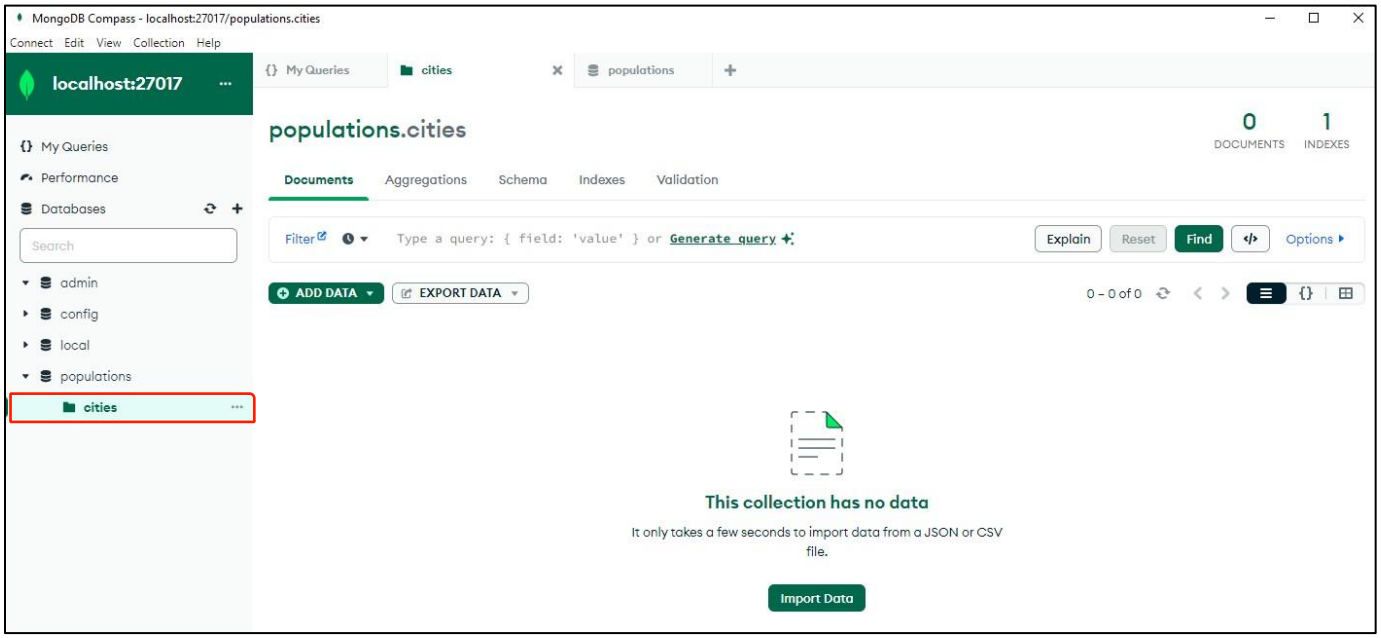


Step 3 - We will create a collection and insert a set of sample data into it. This sample collection includes documents representing the twenty most populous cities in the world. To create a new database, click on the plus sign(+) next to the DATABASE button at the top left corner of the Home screen. Type populations into the Database Name field and cities into the Collection Name field, leaving all other fields with their default values and then click Create Database:



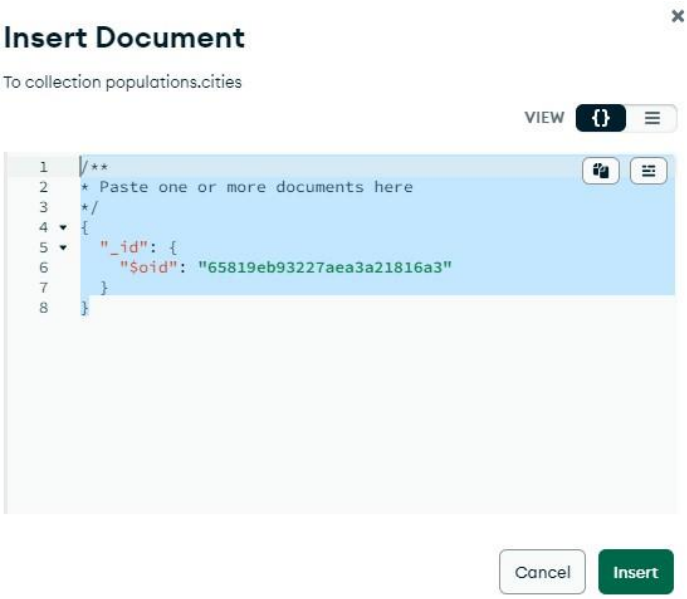
Step 4 – Compass will create the database. Click on the populations database to reach the database view. Then, click on the cities to reach the empty collection view:





Step 5 - Now that the database and collection have been created, you can insert a list of unsorted documents into the cities collection. Click the ADD DATA button and then select the Insert Document option. A window will appear in which you can enter one or more data documents, in JSON format. Enter the following set of documents into the field, replacing any content that was there by default.

Delete the marked area:



- Add the information below to the collection:

[

{"name": "Seoul", "country": "South Korea", "continent": "Asia", "population": 25.674},

{"name": "Mumbai", "country": "India", "continent": "Asia", "population": 19.98},

{"name": "Lagos", "country": "Nigeria", "continent": "Africa", "population": 13.463},

{"name": "Beijing", "country": "China", "continent": "Asia", "population": 19.618},

{"name": "Shanghai", "country": "China", "continent": "Asia", "population": 25.582},

{"name": "Osaka", "country": "Japan", "continent": "Asia", "population": 19.281},

{"name": "Cairo", "country": "Egypt", "continent": "Africa", "population": 20.076},

{"name": "Tokyo", "country": "Japan", "continent": "Asia", "population": 37.4},

{"name": "Karachi", "country": "Pakistan", "continent": "Asia", "population": 15.4},

{"name": "Dhaka", "country": "Bangladesh", "continent": "Asia", "population": 19.578},

{"name": "Rio de Janeiro", "country": "Brazil", "continent": "South America", "population": 13.293},

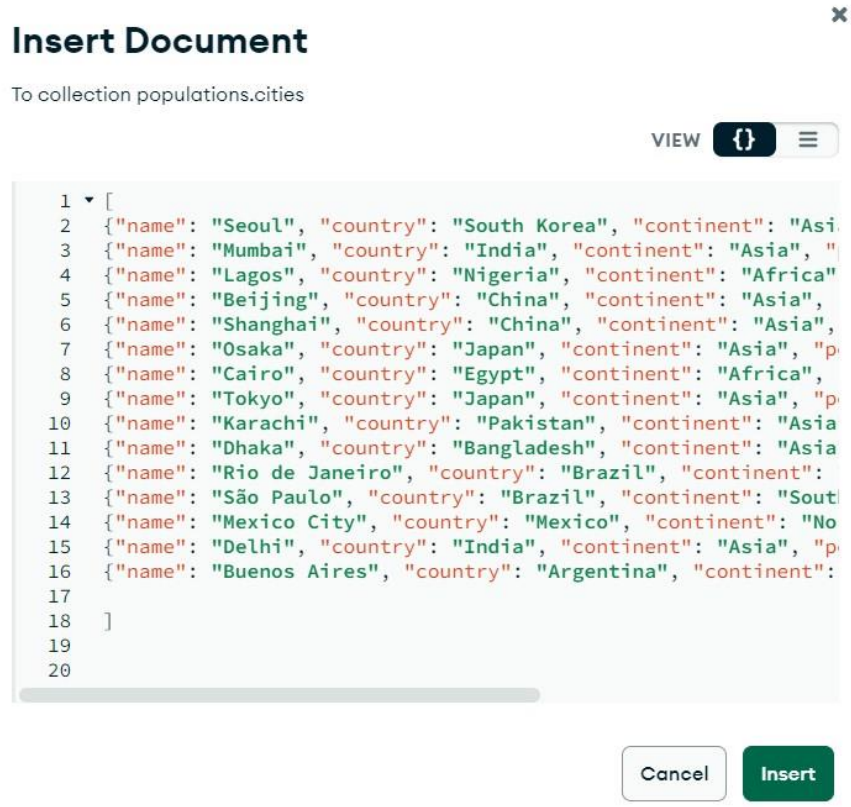
{"name": "São Paulo", "country": "Brazil", "continent": "South America", "population": 21.65},

{"name": "Mexico City", "country": "Mexico", "continent": "North America", "population": 21.581},

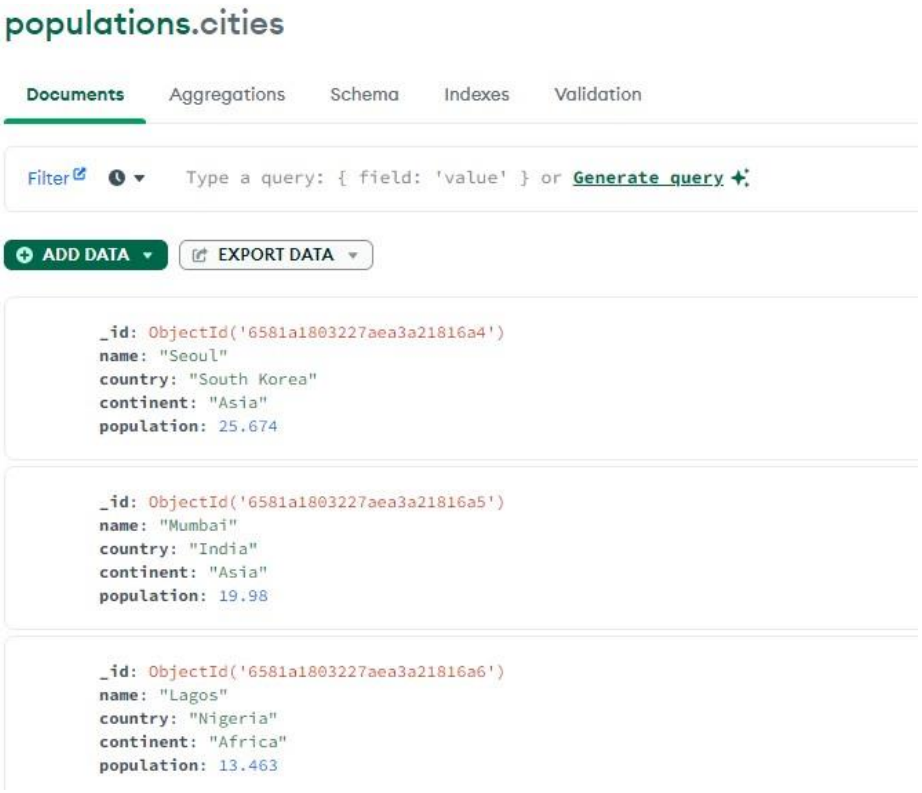
{"name": "Delhi", "country": "India", "continent": "Asia", "population": 28.514},

{"name": "Buenos Aires", "country": "Argentina", "continent": "South America", "population": 14.967}

]



Step 6 - When you click the **Insert** button, Compass adds a list of documents and then automatically displays them in the collection browser. Thus, you have successfully created a set of sample documents representing the world's most populated cities.



Step 7 - By default, Compass will show the first 20 unfiltered results returned by an empty query on the selected collection. At the top right corner, you'll find a View section with three separate display modes you can choose from:

- **List view:** the default view showing documents as key-value pairs shown one in a row. This display mode resembles the JSON document format, but its syntax coloring and additional interface features, such as collapsible nested documents, help make it more readable:

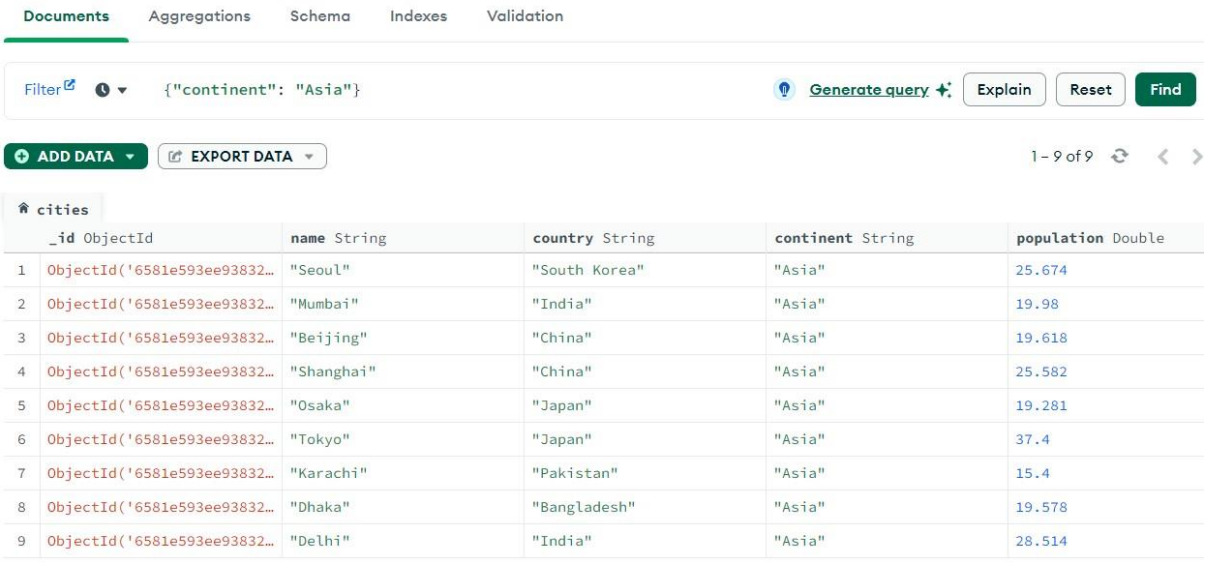
- **JSON view:** this view shows the actual document structure as represented in JSON:

- **Table view:** showing the data in a tabular interface, similar to that found in relational databases. This can be handy if the documents follow a well-defined flat structure but is not as legible when embedded documents or arrays appear in the documents:

cities

	_id ObjectId	name String	country String	continent String	population Double
1	ObjectId('6581a1803227ae…	"Seoul"	"South Korea"	"Asia"	25.674
2	ObjectId('6581a1803227ae…	"Mumbai"	"India"	"Asia"	19.98
3	ObjectId('6581a1803227ae…	"Lagos"	"Nigeria"	"Africa"	13.463
4	ObjectId('6581a1803227ae…	"Beijing"	"China"	"Asia"	19.618
5	ObjectId('6581a1803227ae…	"Shanghai"	"China"	"Asia"	25.582
6	ObjectId('6581a1803227ae…	"Osaka"	"Japan"	"Asia"	19.281
7	ObjectId('6581a1803227ae…	"Cairo"	"Egypt"	"Africa"	20.076
8	ObjectId('6581a1803227ae…	"Tokyo"	"Japan"	"Asia"	37.4
9	ObjectId('6581a1803227ae…	"Karachi"	"Pakistan"	"Asia"	15.4
10	ObjectId('6581a1803227ae…	"Dhaka"	"Bangladesh"	"Asia"	19.578

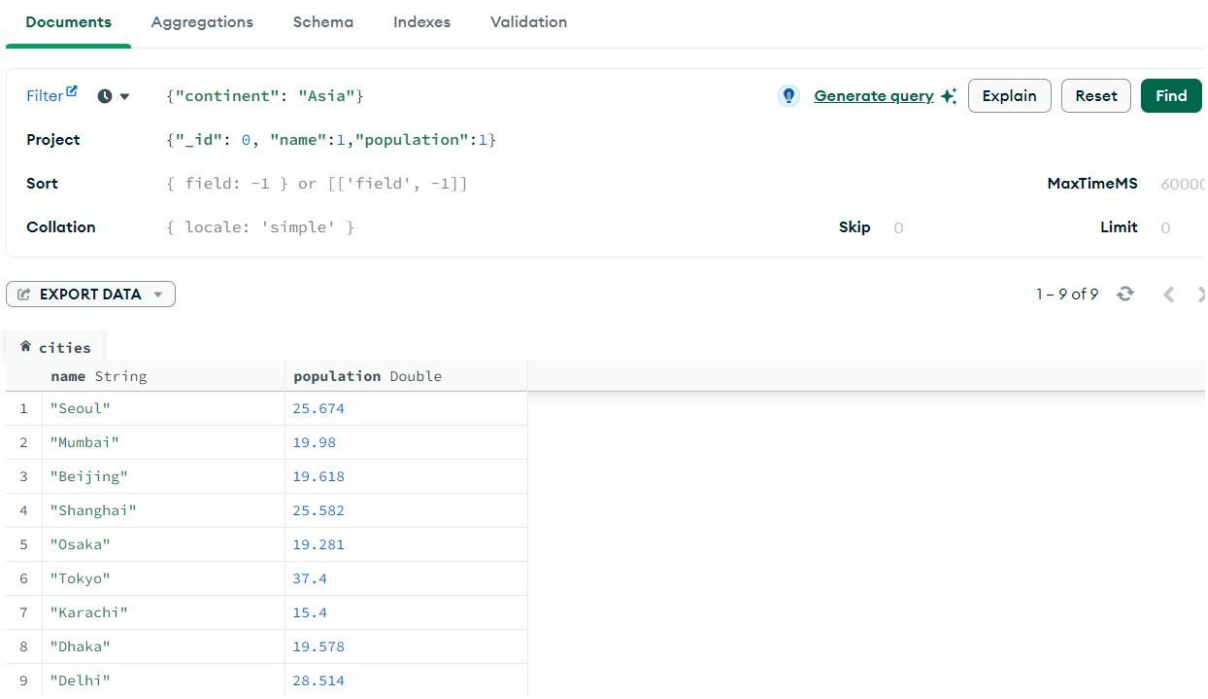
Step 8 - You might query your collection to find all the documents representing cities in North America by running the {"continent": "Asia"} operation in the MongoDB Compass. You can enter this into MongoDB Compass's FILTER field at the top of the collection window to filter your data. Go ahead and enter this query document into the FILTER field, and then press FIND:



	_id ObjectId	name String	country String	continent String	population Double
1	ObjectId('6581e593ee93832…	"Seoul"	"South Korea"	"Asia"	25.674
2	ObjectId('6581e593ee93832…	"Mumbai"	"India"	"Asia"	19.98
3	ObjectId('6581e593ee93832…	"Beijing"	"China"	"Asia"	19.618
4	ObjectId('6581e593ee93832…	"Shanghai"	"China"	"Asia"	25.582
5	ObjectId('6581e593ee93832…	"Osaka"	"Japan"	"Asia"	19.281
6	ObjectId('6581e593ee93832…	"Tokyo"	"Japan"	"Asia"	37.4
7	ObjectId('6581e593ee93832…	"Karachi"	"Pakistan"	"Asia"	15.4
8	ObjectId('6581e593ee93832…	"Dhaka"	"Bangladesh"	"Asia"	19.578
9	ObjectId('6581e593ee93832…	"Delhi"	"India"	"Asia"	28.514

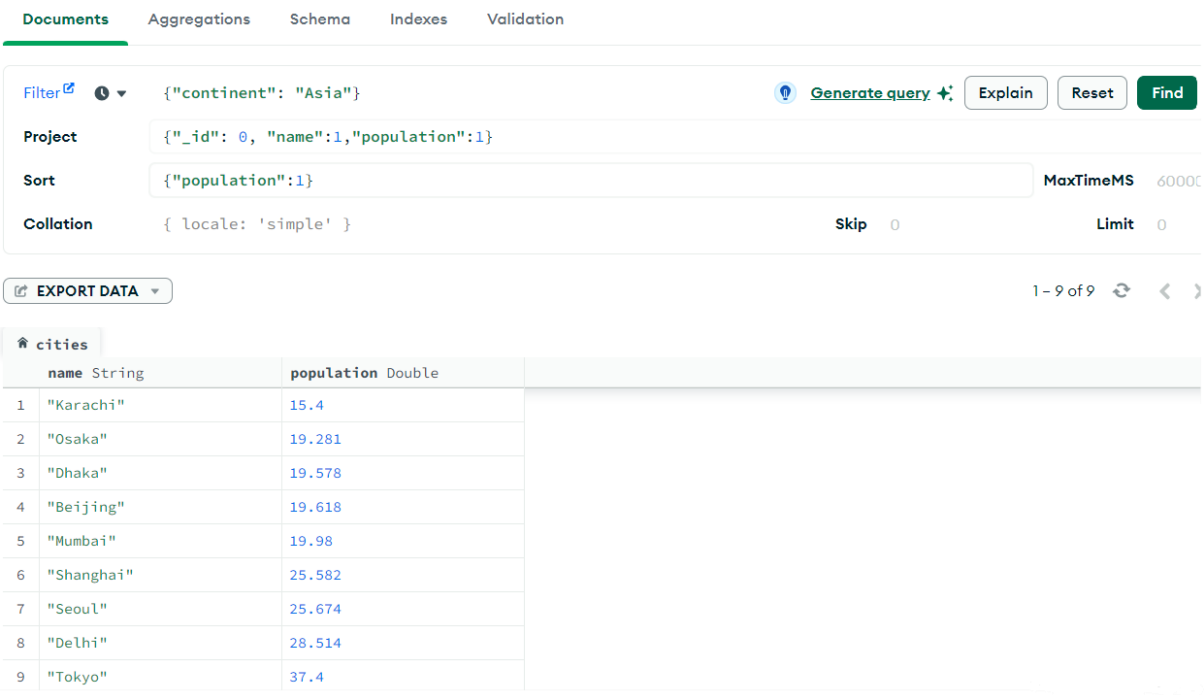
Step 9 - You can sort the results and apply projections to return only a limited subset of fields using the data browser interface. Click on the OPTIONS button near the filtering query bar to reveal further options. The PROJECT and SORT fields will appear below the FILTER field.

- To limit the list of fields to name and population, add the following projection document to the PROJECT field:



	name String	population Double
1	"Seoul"	25.674
2	"Mumbai"	19.98
3	"Beijing"	19.618
4	"Shanghai"	25.582
5	"Osaka"	19.281
6	"Tokyo"	37.4
7	"Karachi"	15.4
8	"Dhaka"	19.578
9	"Delhi"	28.514

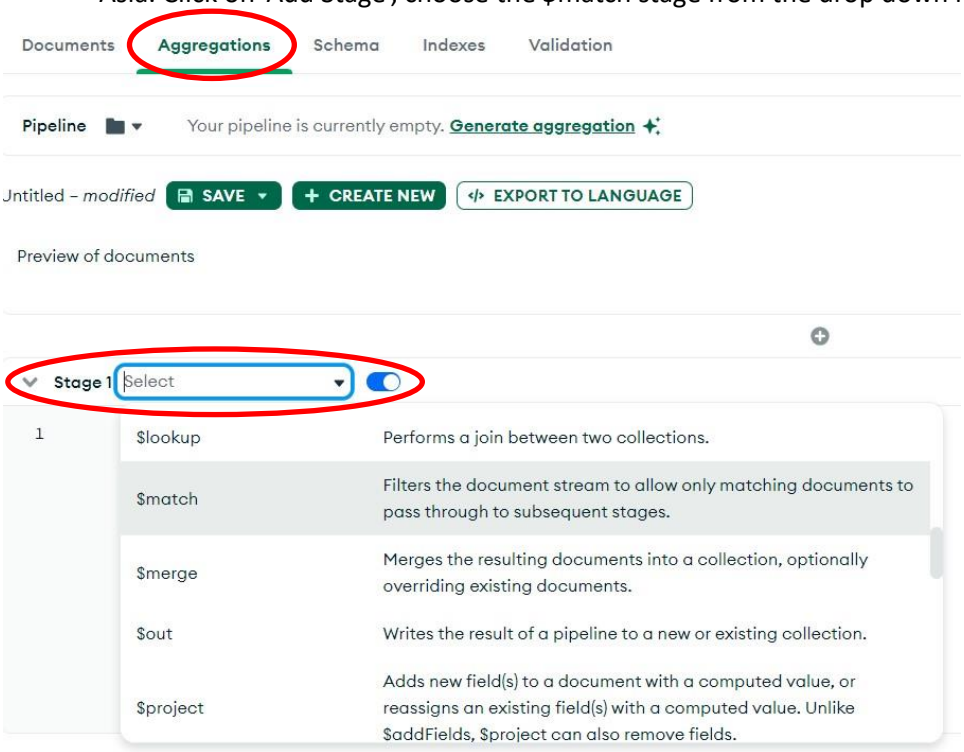
- To sort the list by population in ascending order, add the following sort document to the SORT field and press FIND:



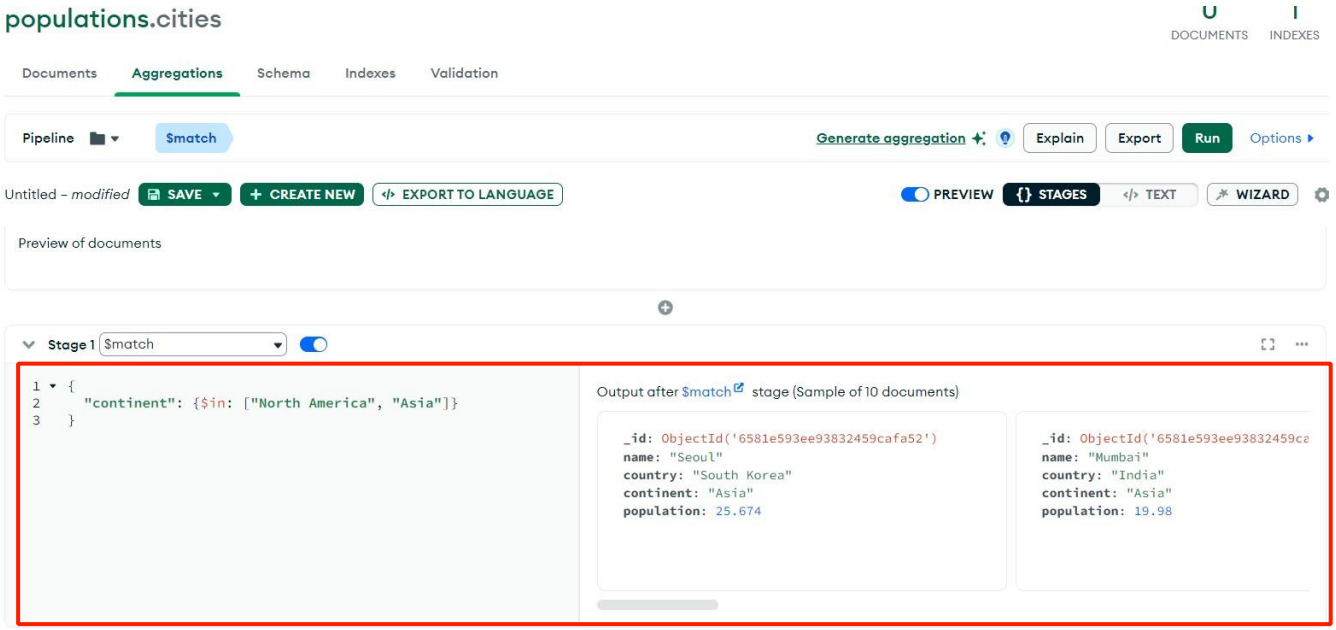
	name String	population Double
1	"Karachi"	15.4
2	"Osaka"	19.281
3	"Dhaka"	19.578
4	"Beijing"	19.618
5	"Mumbai"	19.98
6	"Shanghai"	25.582
7	"Seoul"	25.674
8	"Delhi"	28.514
9	"Tokyo"	37.4

Step 10 - Compass's aggregation pipeline builder is a graphical tool aiding with the creation of multistep aggregation pipelines. Let's list the most populated cities of countries represented in the collection, but only those found in Asia and North America. The pipeline should only return the cities' names and populations. The results should be sorted in descending order by population, returning countries with the largest cities first. Lastly, the documents' structure should replicate the following:

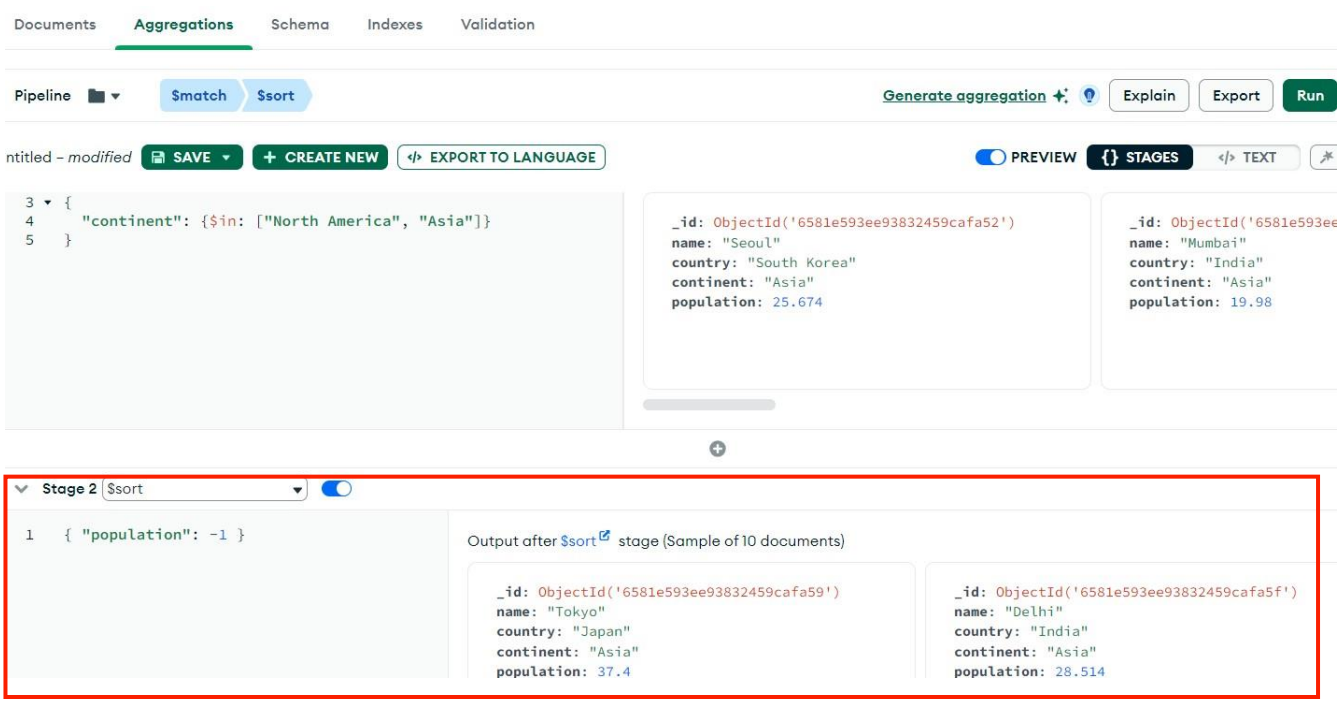
- The first step to building this example pipeline is to filter the initial documents coming from the cities collection so they only contain documents representing cities in North America and Asia. Click on 'Add Stage', choose the $match stage from the drop down menu.



- Enter the {"continent": {$in: ["North America", "Asia"]}} to match only cities from North America and Asia.



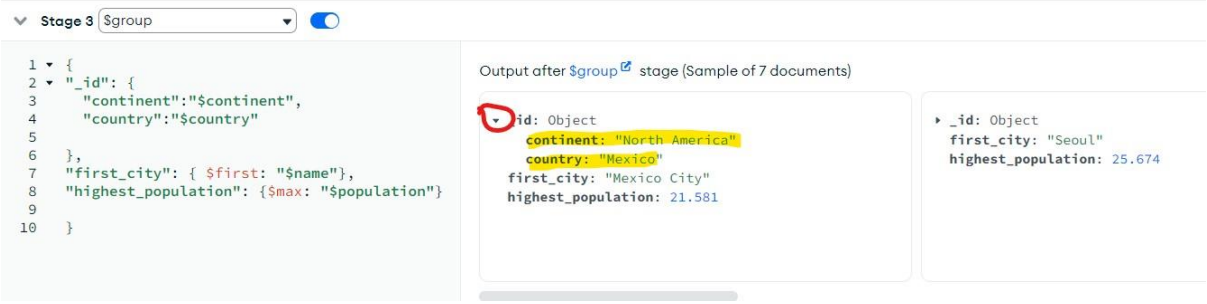
- The second step is to order the documents by population in descending order. Add an additional stage using the ADD STAGE button. Another empty stage row will appear. There, select the $sort stage and enter the { "population": -1 } as the stage settings:



** The returned documents will have the same structure, but Tokyo comes first in the preview pane that now shows the sorted results instead.

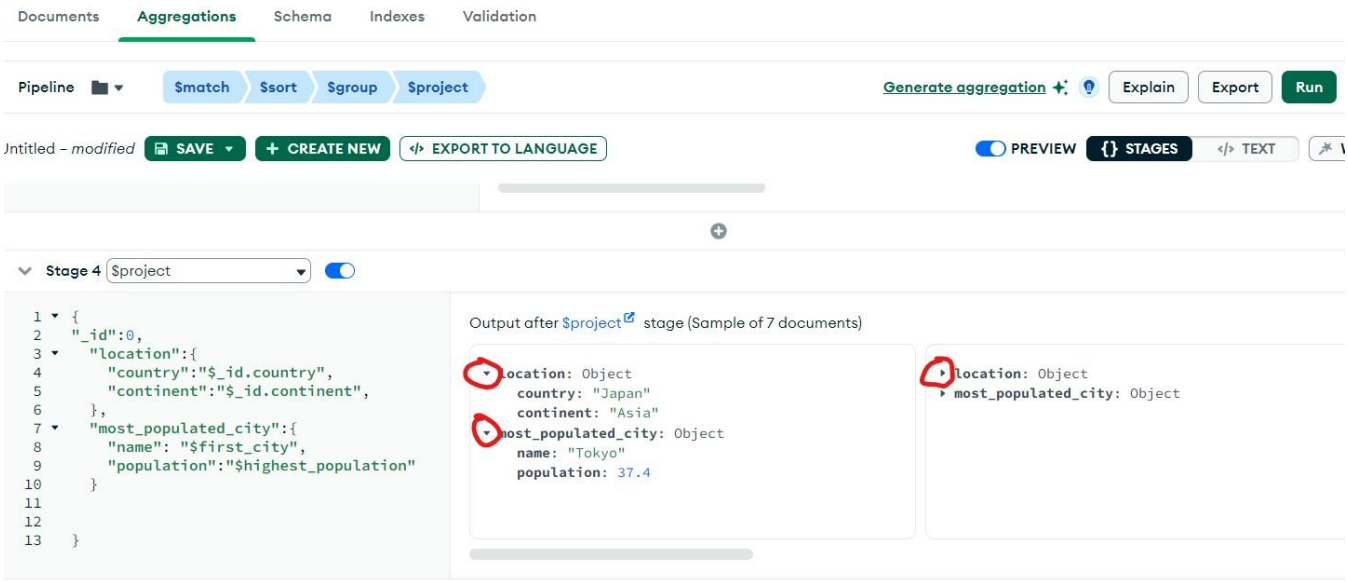
- Since the list of cities is now sorted by the population coming from the expected continents, the next necessary step is to group cities by their countries, choosing only the most populated city from each group. Add another stage, this time selecting $group as the stage type. To group cities by unique continent and country pairs and summarize these pairs by only showing the name and population of their most populated cities, enter the following grouping settings:

** The highest_population value uses the $max accumulator operator to find the highest population in the group, while the first_city gets the name of the first city.

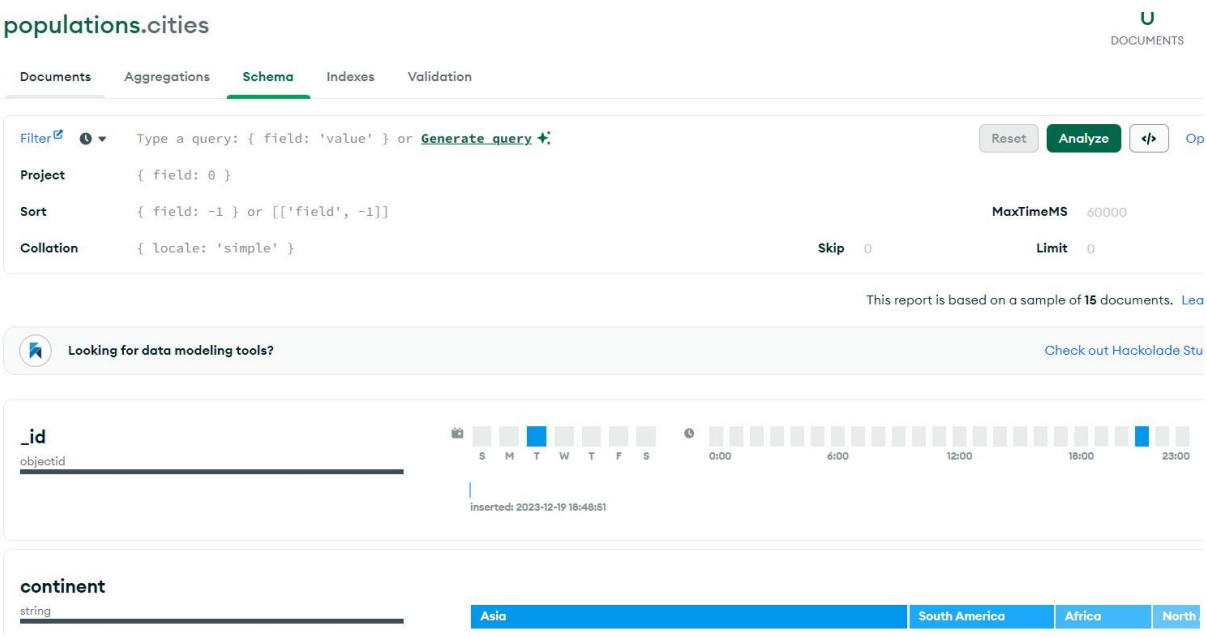


- The last step to satisfying the requirements described at the beginning of this section is to transform the document's structure. You can do that by adding another stage row and selecting $project as the stage. To achieve the specified document structure, enter the following projection document:

** This document first suppresses the _id field so it won't appear in the output at all. Next, it creates a location field written as a nested document with two fields: country and continent. Each of these refers to the values from the input document. Most_populated_city follows a similar principle, nesting the name and population fields inside. Both of these fields refer to the top-level fields first_city and highest_population. This projection stage effectively constructs an entirely new structure for the output:



Step 11 – Schema visualizer interface tool can help you understand the structure of data within your collections. To use it, first select the Schema tab in the cities collection view. The view will initially be empty, but when you press Analyze button, Compass will churn the data to reveal insights about its form, size, and contents.



SUMMARY

In this LAB, you familiarized yourself with MongoDB Compass, a GUI that allows you to manage your MongoDB data through a convenient visual display. You used the tool to create a new collection, insert new documents, filter and navigate the data, create a multi-stage aggregation pipeline, and visualize the collection's schema using the schema visualization tool.

PROCEDURE 3 – ASSIGNMENT (Upload the solution to the CATs by your ID)

You can access your assignment at the time your section starts. Then, you can download the assignment from the Assignments section in CATS.