

ISYS1101/ 1102 Database Applications

Week 8: Tute/Lab – Getting Started with MongoDB

Semester 2 2022

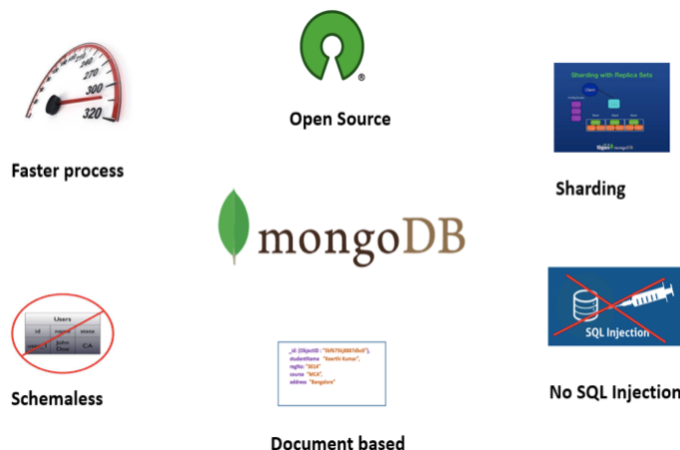
1 Objective

While tutors are conducting demos, please use the time to work through the following activities. The objective of this tute/lab session is to explore the features of MongoDB, learn how to build a document collection and retrieve data from a collection, and how to build a web database application with MongoDB backend.

NoSQL databases, such as MongoDB® (mongodb.com), were created in response to the limitations of traditional relational database technology. When compared against relational databases, NoSQL databases are more scalable and provide superior performance, and their data model addresses several shortcomings of the relational model.

The advantages of NoSQL include being able to handle:

- First and foremost, it is very easy to install and setup the MongoDB.
- The very basic feature of MongoDB is that it is a **schema-less database**. Since MongoDB is schema-free, your code defines your schema.
- Large volumes of structured, semi-structured, and unstructured data;
- Agile development;
- Inherently secure because no sql injection can be made.
- Compared to RDBMS, NOSQL database systems are **light-weight, very little overheads**, and, has a **smaller memory footprint** and as a result, provide a far superior performance.
- **Horizontal scaling (sharding)** – The support for Sharding is one of its key features. Sharding is the process of storing the data in different machines and MongoDB's ability to process the data, as and when the size of the data grows. This results in the horizontal scaling.
- **Replication** -- Auto data replication is also supported in NoSQL databases by default. Hence, if one DB server goes down, data is restored using its copy created on another server in network.



(<https://www.studytonight.com/mongodb/advantages-of-mongodb>)

Today, companies leverage NoSQL databases for a growing number of use cases. NoSQL databases also tend to be open-source and that means a relatively low-cost way of developing, implementing and sharing software.

Companies choose MongoDB for developing modern applications as it offers the advantages of relational databases along with the innovations of NoSQL.

In the second half of the Database Application course, we explore the features of MongoDB, learn how to build a document collection and retrieve data from a collection, and how to build a web database application with MongoDB backend.

In the second assignment you will build an application using MongoDB backend. The activities in this tuto/lab session will assist you to learn the basics required to start your assignment work.

2 Preparation Tasks

MongoDB is not available on School's servers or on mydesktop.rmit.edu.au. However, we can do all of the activities in these lab sessions and assignment activities on Mongo Atlas, their cloud-based database deployment. Mongo Atlas (<https://www.mongodb.com/cloud/atlas>) fully managed database-as-a-service (DBaaS) offering. It has a free tier that is suited for learning and exploring MongoDB in a sandbox environment.

In this course, you will make full use of this free-tier offering of the MongoDB installation.

In order to use the Mongo Atlas, you are required to have two components:

1. A database deployed on Mongo Atlas
2. Mongo Compass desktop client.

In this preparation task, we will complete the deployment/ installation of these two components.



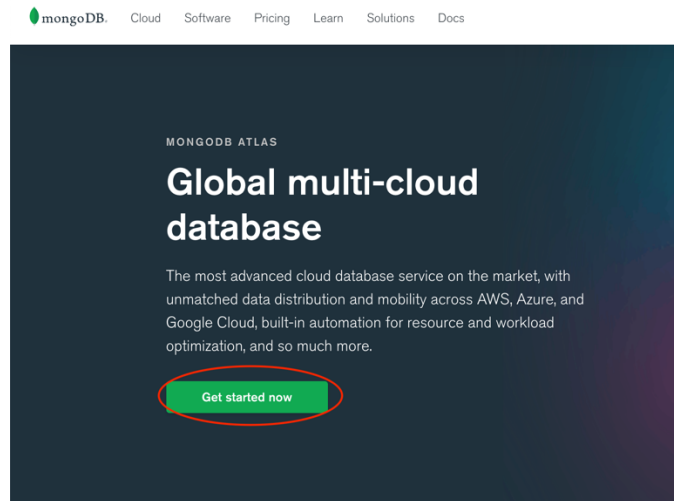
There are many alternative MongoDB clients available. However, Mongo Compass is custom-built for accessing MongoDB databases and comes with both a GUI and a Mongo Shell.

While you are free to choose any client application, we highly recommend the Mongo Compass.

2.1 Create and deploy an Atlas Cluster

MongoDB Atlas provides an easy way to host and manage your data in the cloud. This tutorial guides you through creating an Atlas cluster, connecting to it, inserting data, and querying data.


Visit: <https://www.mongodb.com/cloud/atlas> to get it started.



Follow these steps:

Step 1: Create an Atlas Account

You can register for an Atlas account using your Google Account (if you have one) or using your RMIT student email address. We recommend you use your RMIT student email address to create a new Atlas account. However, you should not use the same password as your RMIT password.



Atlas does not use RMIT's single sign-on.

Get started free

No credit card required

☒ Use your RMIT Email Address

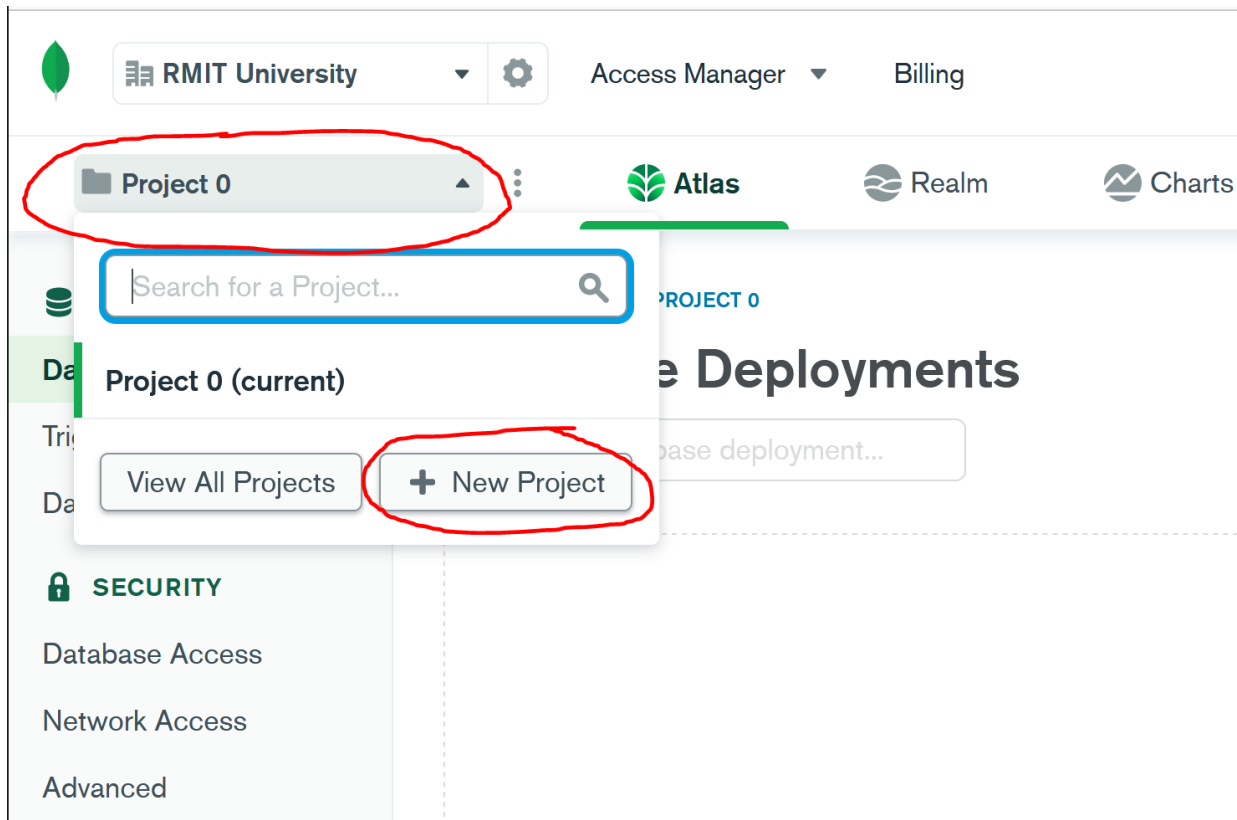
Do not use your RMIT Password.

Or

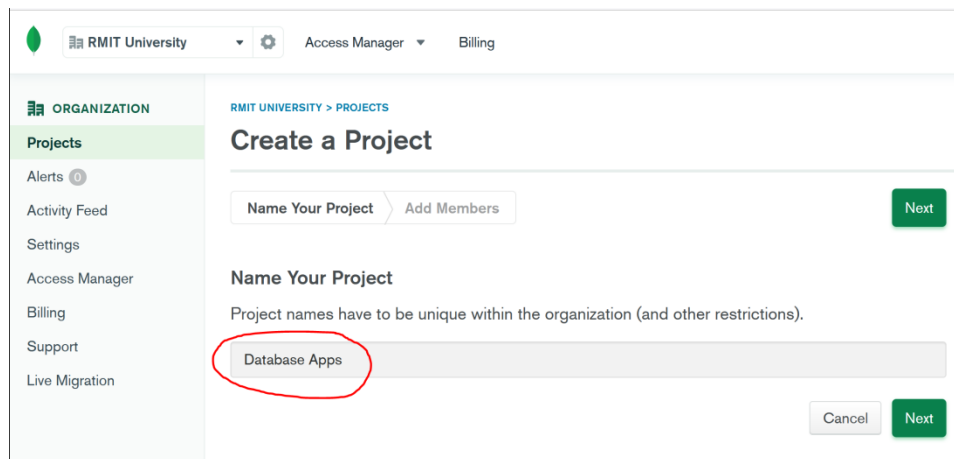
Enter your details and as the company, enter “RMIT University”.

You will receive an email from Mongo Atlas to verify your identity. Simply follow the steps described in the email, and you will be ready to go.

Once your account is created, the next step is to create a project.



Name your project as "Database Apps" and click next to create your project.



Step 2: Deploy a Free Cluster

Atlas free clusters provide a small-scale development environment to host your data. Free clusters never expire, and provide access to a [subset](#) of Atlas features and functionality.

Log into Atlas, at <https://www.mongodb.com/cloud/atlas>

Click **Build a Database**.

Database Deployments

Find a database deployment...



Create a database

Choose your cloud provider, region, and specs.

Build a Database

Once your database is up and running, live migrate an existing MongoDB database into Atlas with our [Live Migration Service](#).

Choose the Free and Hobby option. Choose **M0 Sandbox** tier that gives you free 350MB storage space, one virtual CPU and shared RAM. (other higher-spec'ed options are not free).

Deploy a cloud database

Experience the best of MongoDB on AWS, Azure, and Google Cloud. Choose a deployment option to get started.

PREVIEW

Serverless

For serverless applications that aren't critical with variable traffic. Minimal configuration required.

- ✓ Pay only for the operations you run
- ✓ Resources scale seamlessly to meet your workload
- ✓ Always-on security and backups

Create

Starting at
\$0.30/1M reads

Dedicated

For production applications with sophisticated workload requirements. Advanced configuration controls.

- ✓ Network isolation and fine-grained access controls
- ✓ On-demand performance advice
- ✓ Multi-region and multi-cloud options available

Create

Starting at
\$0.08/hr*
*estimated cost \$55.94/month

FREE

Shared

For learning and exploring MongoDB in a cloud environment. Basic configuration options.

- ✓ No credit card required to start
- ✓ Explore with sample datasets
- ✓ Upgrade to dedicated clusters for full functionality

Create

Starting at
FREE

Choose the Free and Shared option.

Select your preferred *Cloud Provider & Region*.

Cloud Provider: AWS

Region: Sydney (ap-southeast-2)

CLUSTERS > CREATE A SHARED CLUSTER

Create a Shared Cluster

Welcome to MongoDB Atlas! We've recommended some of our most popular options, but feel free to customize your cluster to your needs. For more information, check our [documentation](#).

PREVIEW Serverless

Dedicated

FREE Shared

For learning and exploring MongoDB in a sandbox environment. Basic configuration controls.

No credit card required to start. Upgrade to dedicated clusters for full functionality.
Explore with sample datasets. Limit of one free cluster per project.

Cloud Provider & Region

AWS, Sydney (ap-southeast-2) ▼



★ Recommended region ⓘ

NORTH AMERICA

🇺🇸 N. Virginia (us-east-1) ★

🇺🇸 Oregon (us-west-2) ★

EUROPE

🇮🇪 Ireland (eu-west-1) ★

🇩🇪 Frankfurt (eu-central-1) ★

ASIA

🇸🇬 Singapore (ap-southeast-1) ★

🇮🇳 Mumbai (ap-south-1)

AUSTRALIA

🇦🇺 Sydney (ap-southeast-2) ★

Cluster Tier

M0 Sandbox (Shared RAM, 512 MB Storage)

Encrypted

Additional Settings

MongoDB 4.4, No Backup

Cluster Name

DBA-Cluster ▼

One time only: once your cluster is created, you won't be able to change its name.

DBA-Cluster

Cluster names can only contain ASCII letters, numbers, and hyphens.

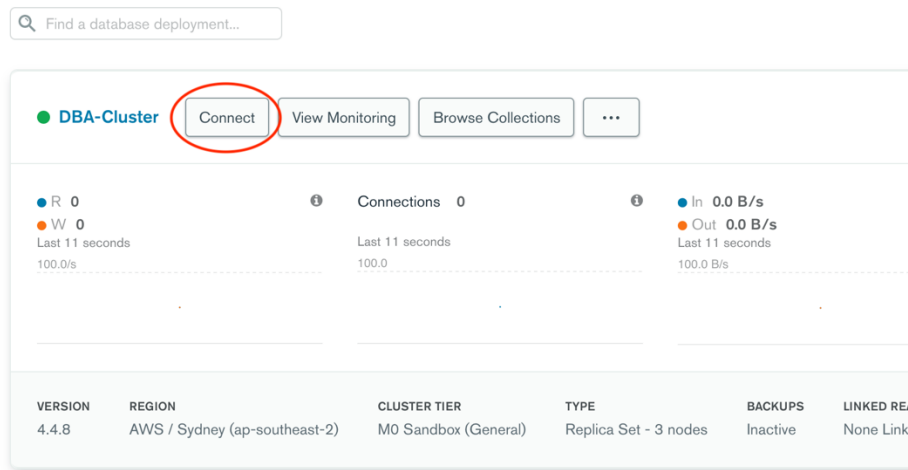
Give a suitable name (say: DBA-Cluster) to your newly created database cluster.

Click “Create Cluster” button.

The deployment will take a few minutes.

Once your 3-server database cluster is created, click on Connect to configure connection settings.

Database Deployments



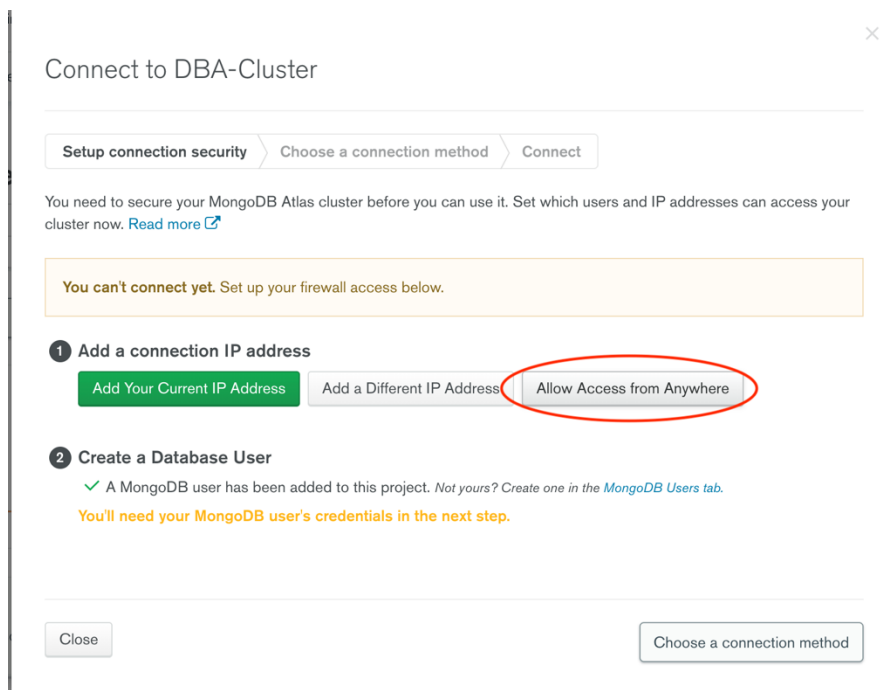
VERSION	REGION	CLUSTER TIER	TYPE	BACKUPS	LINKED RE/
4.4.8	AWS / Sydney (ap-southeast-2)	M0 Sandbox (General)	Replica Set - 3 nodes	Inactive	None Link

Under “Add a connection IP address”, choose “Allow Access from Anywhere” option. This is not the most secured access method and should not be used with real database deployments. However, assuming that you connect this database from many dynamically-allocated IP addresses (e.g. university network, free wifi at library, home wifi), it is acceptable for this sandbox environment.

On next page, default IP address 0.0.0.0/0 is shown.

0.0.0.0/0 denotes allowing access from anywhere.

Click on “Add IP Address” button.



Connect to DBA-Cluster

Setup connection security > Choose a connection method > Connect

You need to secure your MongoDB Atlas cluster before you can use it. Set which users and IP addresses can access your cluster now. [Read more](#)

You can't connect yet. Set up your firewall access below.

1 Add a connection IP address

Add Your Current IP Address Add a Different IP Address **Allow Access from Anywhere**

2 Create a Database User

✓ A MongoDB user has been added to this project. Not yours? Create one in the [MongoDB Users](#) tab.

You'll need your MongoDB user's credentials in the next step.

Close Choose a connection method

Create a Database User:

Use your student number and a separate password to create a database user. This user will be the admin user of your database cluster. You will be able to create more users later. However, to complete the configuration process, you will need at least one user.

2 Create a Database User

This first user will have [atlasAdmin](#) permissions for this project.
Keep your credentials handy, you'll need them for the next step.

Username: s3xxxxxx Password: [Autogenerate Secure Password](#)

DO NOT use your RMIT Password [SHOW](#)

[Create Database User](#)

[Close](#) [Choose a connection method](#)

Then, on next screen, choose, Mongo Compass option.

Connect to DBA-Cluster

[Setup connection security](#) [Choose a connection method](#) [Connect](#)

Choose a connection method [View documentation](#)

Get your pre-formatted connection string by selecting your tool below.

- [Connect with the MongoDB Shell](#)
Interact with your cluster using MongoDB's interactive Javascript interface
- [Connect your application](#)
Connect your application to your cluster using MongoDB's native drivers
- [Connect using MongoDB Compass](#)
Explore, modify, and visualize your data with MongoDB's GUI

[Go Back](#) [Close](#)

Note down the connection string. It will be used in the next section when you configure your Mongo Compass desktop client. It should be similar to:

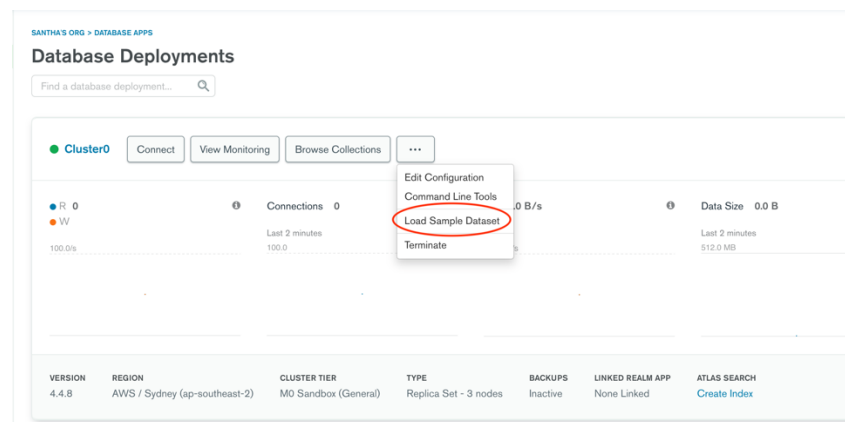
`mongodb+srv://s3xxxxxx:<password>@dba-cluster.wr0r0.mongodb.net/test`

If you haven't downloaded the Mongo Compass Desktop Client, at this page, you will get a link to the installation pack. You can download it directly, from Mongodb website, too.

Step 3: Load a sample dataset

MongoDB has several sample datasets containing real-life data. One of them is a subset of AirBnB data set. It contains an exact replica of the backend database used in AirBnB website.

Click on the three dots on Database Deployment page and choose “Load Sample Dataset” option.



This data set is about 350MB large and will take a substantially long time to load up. (depending on the server load, it can take more than 15 minutes).

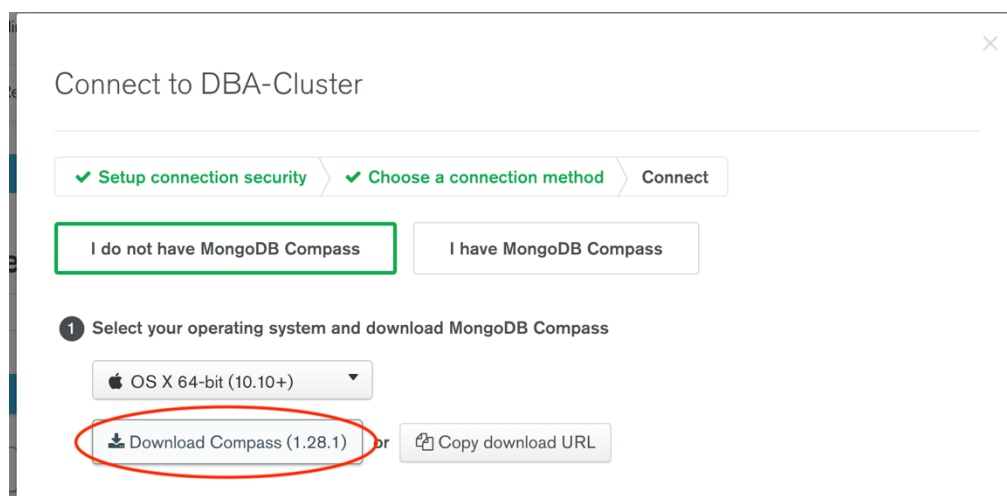
While this is happening on the “cloud”, you can start the installation of the Mongo Compass desktop client.

2.2 Install Mongo Compass Desktop Client

In the previous step when you set up the Atlas Connection settings, you have presented with a link to download the installation package (each operating system will have their own ways of download and install applications).

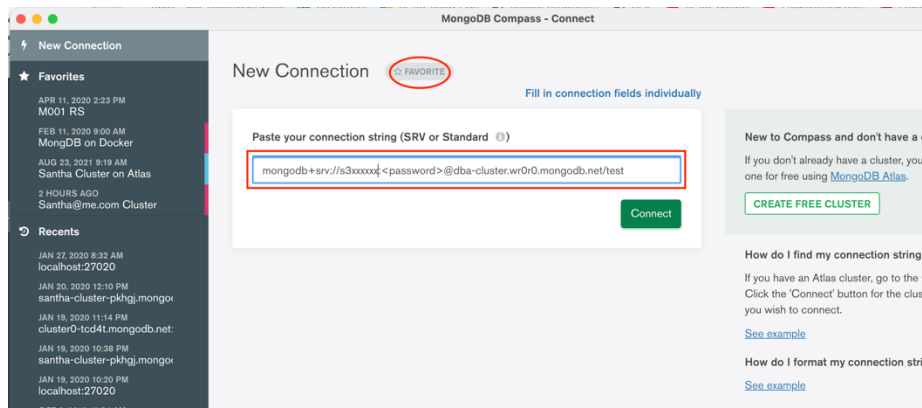
Download and install the application as per operating system’s application installation guidelines.

You may have to grant to run applications downloaded from third parties.



Start the application.

On main menu, choose Connect → Connect to option.

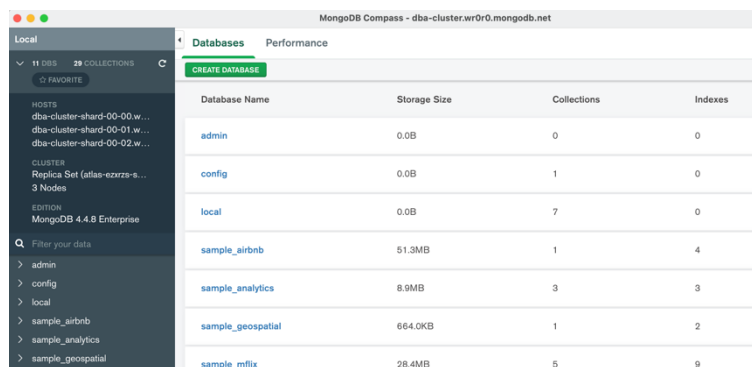


Enter the connection string you copied while configuring your Database cluster.

Replace <password> with the password you previously set up.

It is good idea to tick this connection as a “favourite”, so it will be saved and you are not required to enter the connection string every time you log in.

Now, you are ready to use your newly deployed database cluster.

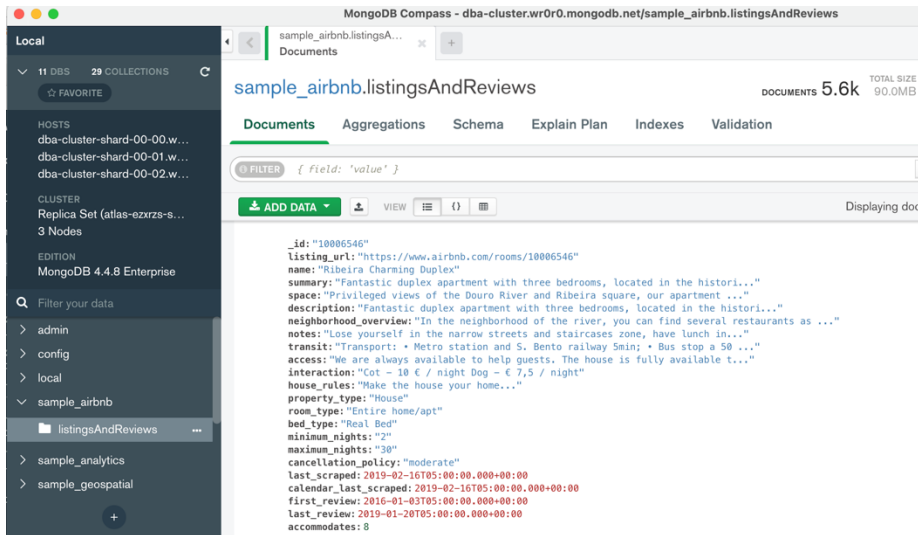


3 Working with MongoDB

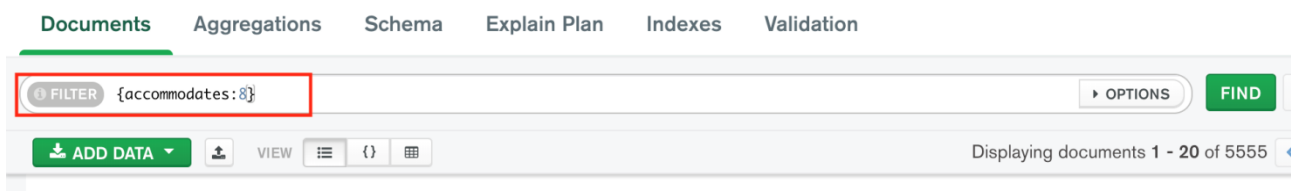
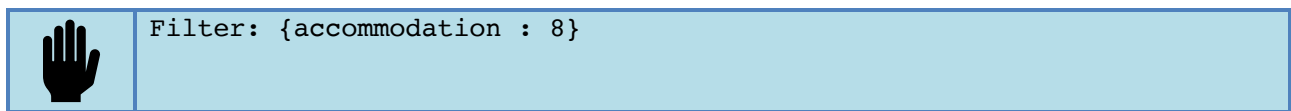
3.1 Exploring AirBnB database

The Mongo Compass allows you to browse and query document collections/databases on Mongo Atlas.

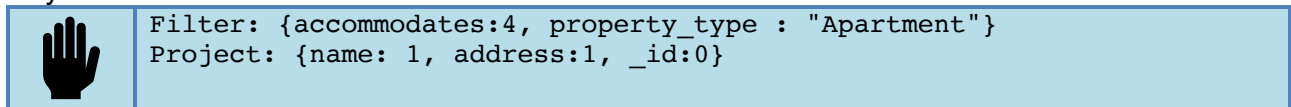
e.g. To browse **listingsAndReviews** collection, click the name of the database (**sample_airbnb**) from left menu, and choose the document collection from the dropdown list.



To filter on various filtering conditions, you can enter your filter condition here:
e.g. Find properties that can accommodate (exactly) 8 people.



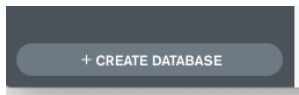
e.g. Find Apartments that can accommodate 4 people. Display the property names and addresses only.




3.2 Creating a small document collection

In this activity, we create a new database and a new document collection and populate it with few documents.

Open MongoDB Compass, on the main window, click the “Create Database” button at the bottom left corner (Hover the mouse over “+” symbol).



On the “Create Database” dialog box, enter



Database Name: MyDB
Collection Name: MyUsersCollection
Leave the tick boxes unticked

Create Database

Database Name

Collection Name

☐ Capped Collection ⓘ

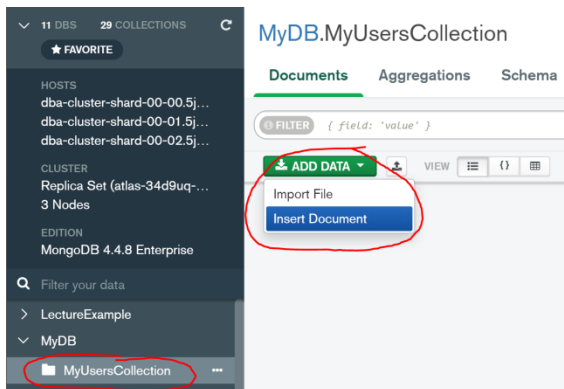
☐ Use Custom Collation ⓘ

Before MongoDB can save your new database, a collection name must also be specified at the time of creation. [More Information](#)

CANCEL

CREATE DATABASE

The new database with a new blank collection should appear on database list.



Click on “Insert Document” button to insert a new document.

When clicked, a document insertion template will appear with the document number already generated.

By default it goes into the raw comma separated file format view where you can populate the data:

Insert to Collection MyDB.MyUsersCollection

VIEW  

```
1 /**
2  * Paste one or more documents here
3  */
4 {
5   "_id": {
6     "$oid": "6134a0d3b8df12dbf6200ee7"
7   }
8 }
```

Alternatively you can go into the structured list format:

Insert to Collection MyDB.MyUsersCollection

VIEW



_id: ObjectId("6134a0d3b8df12dbf6200ee7")

ObjectId ▾

⊕ Add Field After _id

CANCEL

INSERT

Populate the template with the rest of the data. You are required to enter both field name AND data for the field.

Insert Document

1

_id : ObjectId("5d7507c528e5b1700b734868 ")

ObjectId

2

Name : "Sam"

String

3

State : "VIC"

String

4

Email : sam@mail.com

String

CANCEL

INSERT

The new document should appear on your document browser window:

MyDB.MyUsersCollection

DOCUMENTS 1

TOTAL SIZE 75B

AVG. SIZE 75B

INDEXES 1

TOTAL SIZE 4.0KB

AVG. SIZE 4.0KB

Documents

Aggregations

Explain Plan

Indexes

⌵ FILTER

⌵ OPTIONS

FIND

RESET

⋮

INSERT DOCUMENT

VIEW

LIST

TABLE

Displaying documents 0 - 1 of 1

⏪

⏩

🔄

_id: ObjectId("5d7507c528e5b1700b734868")

Name: "Sam"

State: "VIC"

Email: "sam@mail.com"

Insert another document. This time, with Name, State, and Telephone.

Insert Document

1 `_id : ObjectId("5d75091c28e5b1700b734869 ")`
2 `Name : "James "`
3 `State : "NSW "`
4 `Telephone : "0499 999 999"`

ObjectId

String

String

String

String

String

CANCEL

INSERT

Did you already notice a difference between (schema-based) relational database model and (schema-less) NOSQL model?

You can browse the document collection, as a list, a table or in raw file format:

INSERT DOCUMENT

VIEW

LIST

TABLE

Displaying documents 1 - 2 of 2

MyUsersCollection

	<code>_id</code> ObjectId	<code>Name</code> String	<code>State</code> String	<code>Email</code> String	<code>Telephone</code> String
1	5d7507c528e5b1700b734868	"Sam"	"VIC"	"sam@mail.com"	No field
2	5d75091c28e5b1700b734869	"James"	"NSW"	No field	"0499 999 999"

3.3 Access the database via MongoDB Shell

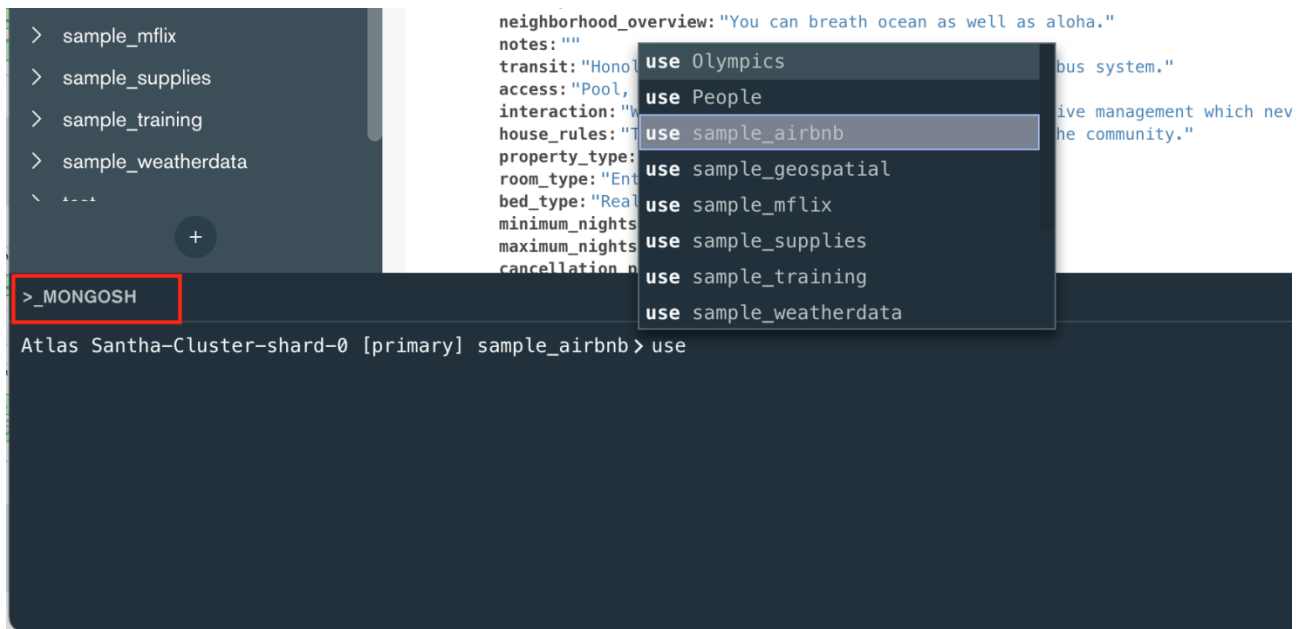
We can interact with the database via a command-line utility, called MongoDB Shell. The current version of Mongo Compass has in-built MongoDB shell.

Click on the small icon at the bottom left corner (">MONGOSH").

First, open an interactive shell, as follows:

```
>_MONGOSH
Atlas Santha-Cluster-shard-0 [primary] sample_airbnb>
```

At the prompt on Mongo Shell, try out the following Mongo Shell commands.



Activity 1: Check the list of databases.



> show databases

Activity 2: Use one of the databases



> use sample_airbnb

IMPORTANT: Unlike Oracle, MongoDB is case sensitive. So, sample_airbnb is different to SAMPLE_AirBnB or Sample_AIRBNB and so on. Use the exact case as displayed on Activity 1.

Activity 3: Check the list of collections.



> show collections

Activity 4: Check the contents of the 'listingsAndReviews' collection.



> db.listingsAndReviews.find()

Exercise:

Redo all the exercises in Section 3.2, this time using the Mongo Shell commands. Refer to lecture slides for corresponding Mongo Shell commands.