



## SESSION 11

### MONGODB CLOUD

---

#### Learning Objectives

In this session, students will learn to:

- Explain how to create an Atlas account and set up a cluster
- Describe how to access the MongoDB Atlas cluster
- Explain how to import data into the Atlas cluster from a MongoDB instance and manage the imported data
- Explain how to export data from the Atlas cluster into a MongoDB instance and manage the exported data
- Describe how to perform administrative tasks in the MongoDB cluster

MongoDB Atlas is a cloud-based MongoDB service that provides a reliable and convenient platform for storing and accessing data. It offers a user-friendly interface and eliminates the requirement for manual infrastructure management. MongoDB Atlas is a popular cloud-based database used by organizations to store large amounts of data, such as user data for social media, e-commerce, or gaming. For example, Netflix and Spotify use MongoDB Cloud to store user profiles, watch history, recommendations, songs, artists, and playlists.

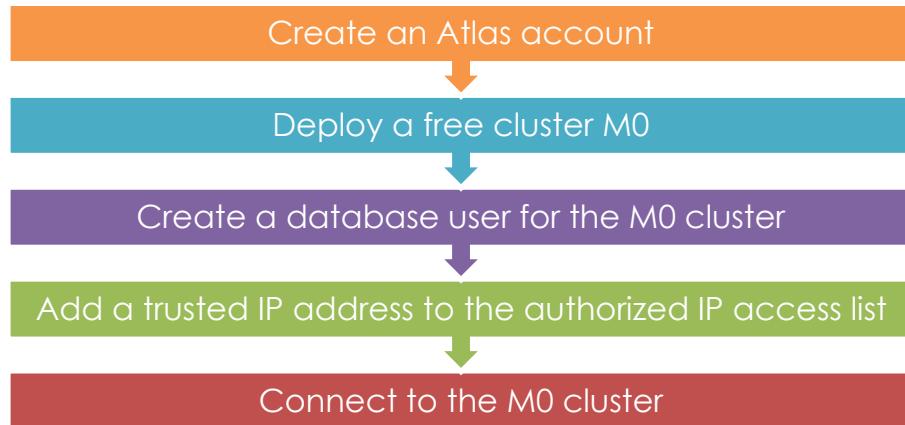
This session will walk through the entire process of setting up MongoDB Atlas, from creating an account to accessing the cluster. In addition, this session will

explore the methods for importing and exporting data between a local MongoDB instance and the MongoDB Atlas cluster. It will also cover how to perform essential administrative tasks such as pausing, resuming, and terminating the MongoDB cluster.

## 11.1 Get Started with MongoDB Atlas

---

Steps to set up MongoDB Atlas are:



### 11.1.1 Create an Atlas Account

MongoDB Atlas is a cloud-based service that requires an account to access its servers. With an Atlas account, the Atlas dashboard can be accessed to deploy MongoDB clusters, create database users, and manage databases.

To create an Atlas account, use any one of the three options:



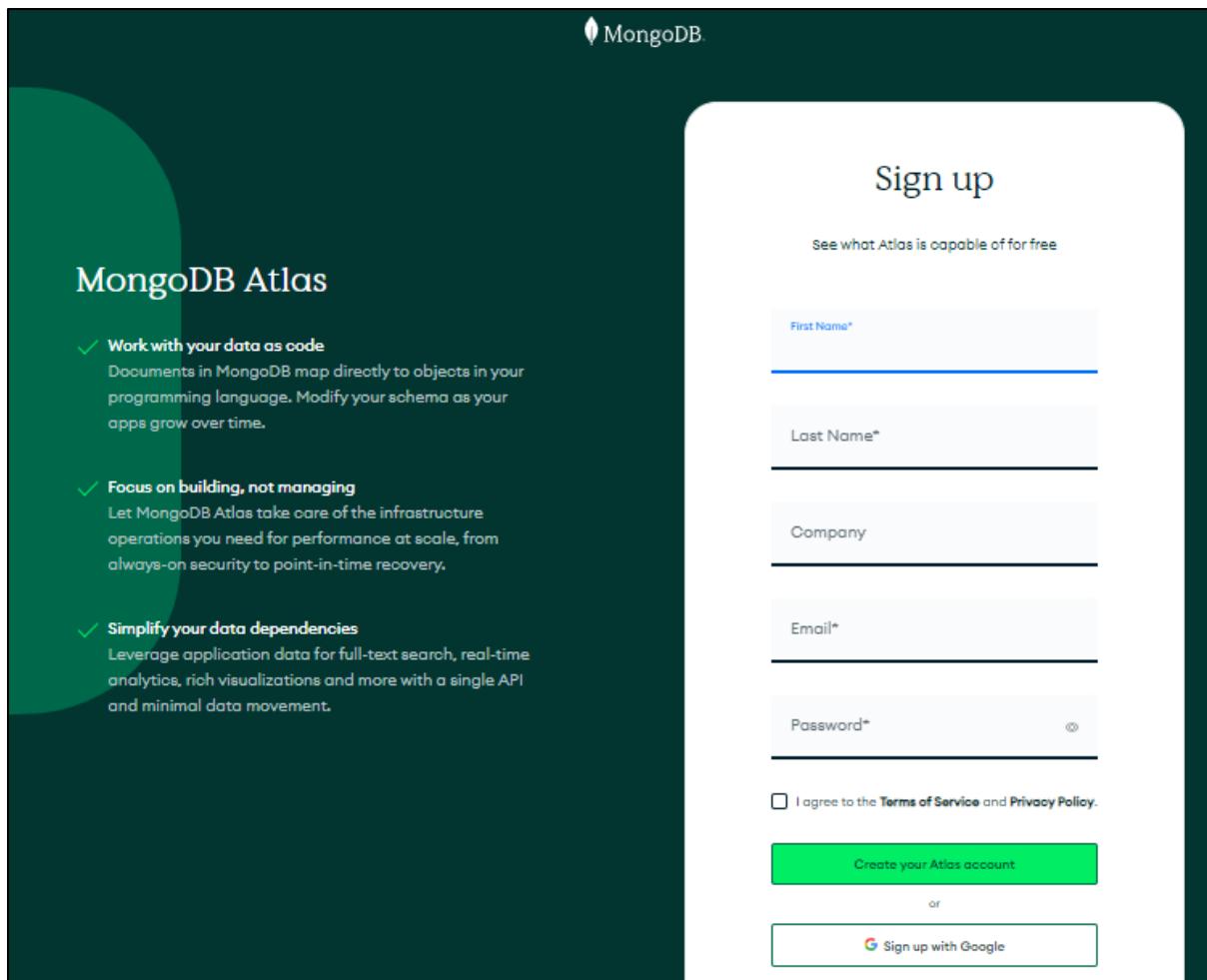
---

Google account is the preferred method. However, email address can also be used.

To create the Atlas account:

1. Open the browser and navigate to the URL:  
<https://www.mongodb.com/cloud/atlas/register>

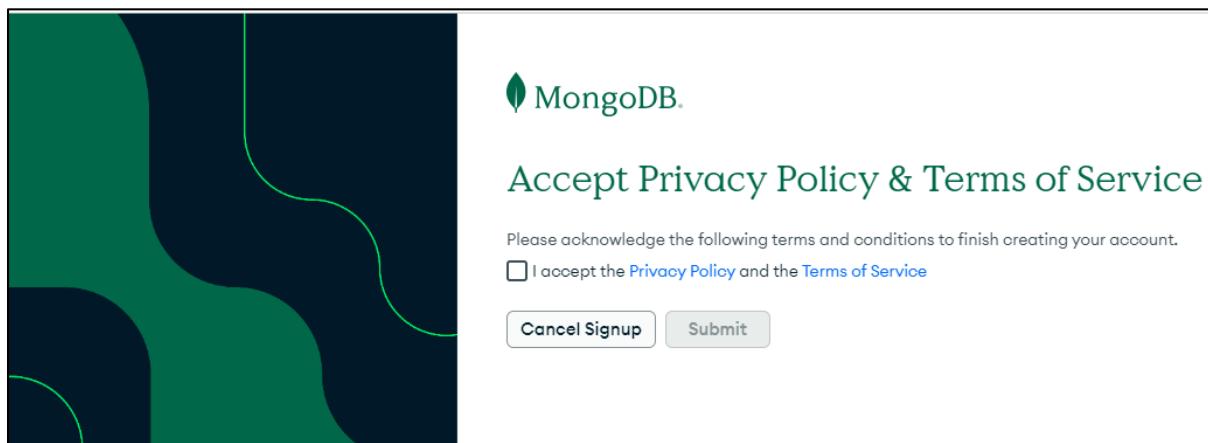
The **MongoDB Atlas Landing** page opens as shown in Figure 11.1.



**Figure 11.1: MongoDB Atlas Landing Page**

2. To sign up using the preferred method or Google account, on this page, scroll down and click **Sign up with Google**.

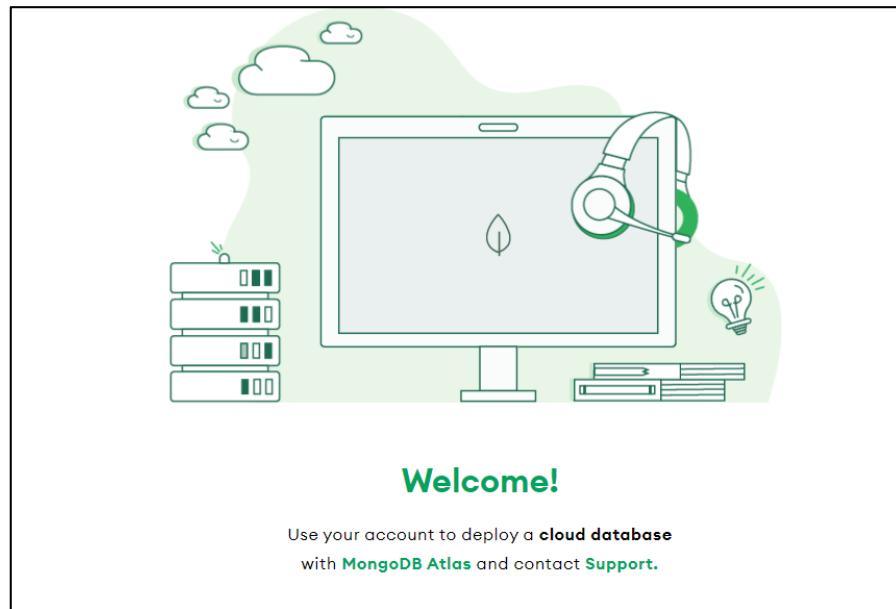
The **Accept Privacy Policy and Terms of Service** page opens as shown in Figure 11.2.



**Figure 11.2: Accept Privacy Policy and Terms of Service Page**

3. To understand how the data is collected, used, and shared, review the privacy policy and terms of service. To proceed, select the **I accept the Privacy Policy and Terms of Service** check box.
4. Click **Submit**.

This completes the sign-up process and welcomes the user to the account as shown in Figure 11.3.



**Figure 11.3: Welcome to Account Page**

Atlas automatically creates a default organization and a project. For first time users, Atlas prompts to respond to a few queries related to the user and the project as shown in Figure 11.4.

**Welcome to Atlas!**

Tell us a few things about yourself and your project.

What is your goal today?

Your answer will help us guide you to successfully getting started with MongoDB Atlas.

Build a new application  
 Migrate an existing application  
 Explore what I can build  
 Learn MongoDB

---

What type of application are you building?

I'm just exploring ▾

---

What is your preferred language?

We'll use this to customize code samples and content we share with you. You can always change this later.

Other ▾

---

**Finish**

**Figure 11.4: Welcome to Atlas Page**

5. On this page, respond to all the queries appropriately and click **Finish**. This completes the Atlas account creation process. The next step is to create a cluster.

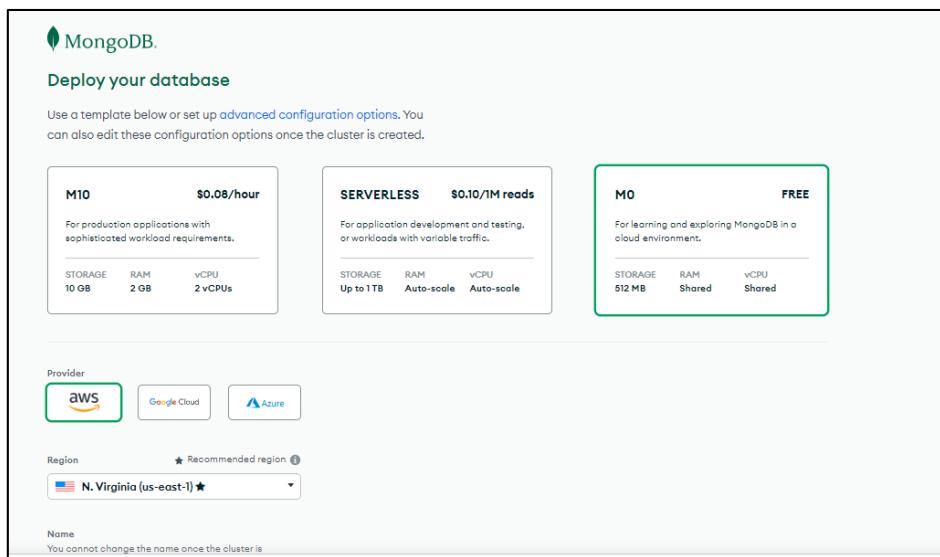
### 11.1.2 Deploy a Free Cluster

After creating an Atlas account, a cluster can be deployed. A cluster is a group of servers that work together to store and manage data. To create a MongoDB cluster in Atlas, the user must specify the required size and configuration, and Atlas takes care of the rest.

M0 clusters are a free, entry-level option for MongoDB users who are learning the database or developing small, proof-of-concept applications. These clusters are limited in terms of storage and features, but they are a great way to get started with MongoDB without any upfront costs. These clusters are small-scale, but they never expire, and they provide access to a subset of Atlas features.

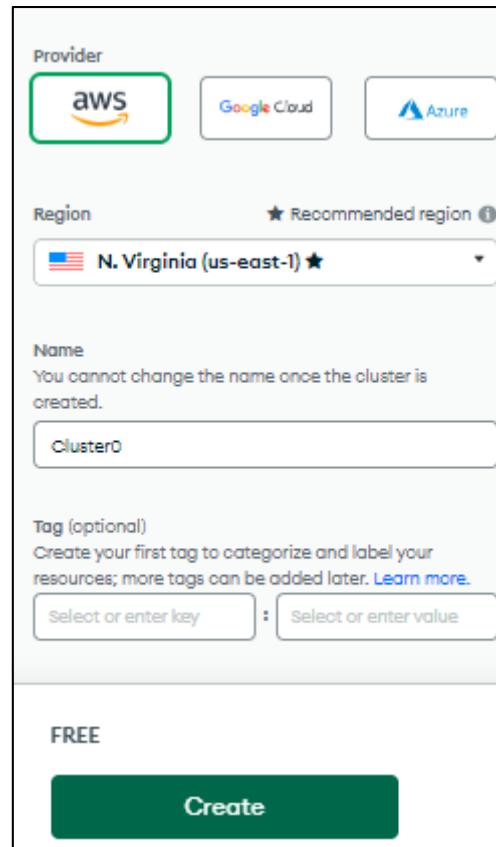
To deploy M0 or the free cluster:

1. From the **Deploy your database** page shown in Figure 11.5, select the **M0** option.



**Figure 11.5: Deploy Your Database Page**

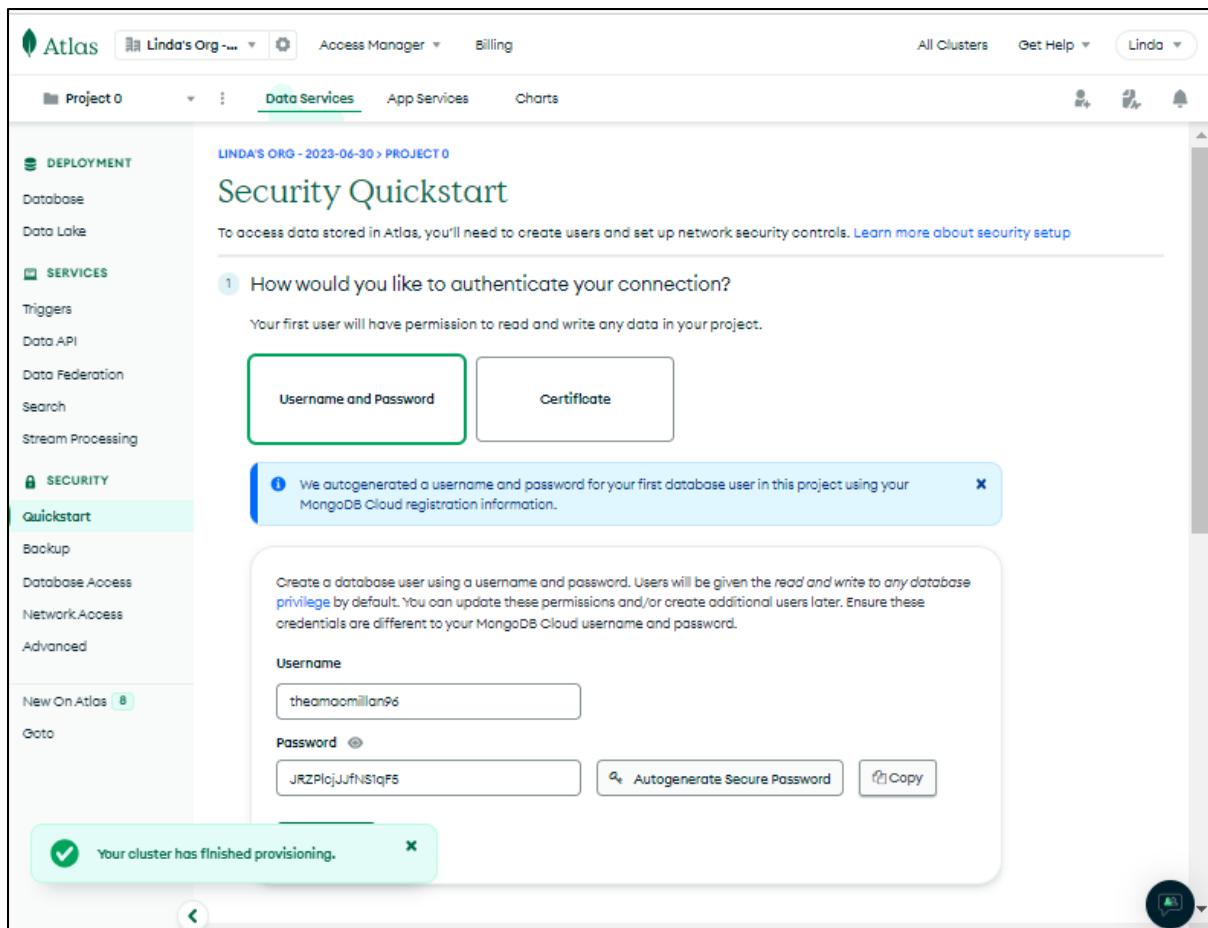
2. Scroll down to make more selections as shown in Figure 11.6.
  - Select the preferred **Provider**.  
Atlas supports M0 free clusters on Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure. In this case, the **AWS** option is selected.
  - Select the preferred **Region**.  
Atlas displays only the cloud provider regions that support M0 free clusters. Here, **N. Virginia (us-east-1)** is selected.
  - Enter a name for the cluster in the **Name** box.  
Users can specify any name for the cluster, as long as it contains American Standard Code for Information Interchange or ASCII letters, numbers, and hyphens. Retain **Cluster0** as given here.



**Figure 11.6: Deploy Your Database Page**

6. Click **Create**.

The **Security Quickstart** page opens as shown in Figure 11.7 with the success message.



**Figure 11.7: Success Message on the Security Quickstart Page**

This completes the deployment of free cluster M0.

### 11.1.3 Create a Database User for the M0 Cluster

To protect data, Atlas requires users to authenticate as valid MongoDB database users before they can access the cluster. To allow users to access the MongoDB database on the cluster, a database user must be created after deploying the cluster. The database user can then be granted permission to access the cluster and the database hosted on the cluster. The database user accounts can be created for different users and roles, and each account can be granted different permissions.

There is a difference between database users and Atlas users in MongoDB Atlas:

Database Users

Can access databases hosted in Atlas.

Atlas Users

Can log in to Atlas but cannot access the MongoDB databases.



---

A database user account can ONLY be created by a user who has either Organization Owner or Project Owner permission.

---

MongoDB Atlas autogenerates a username and password for the first database user in the project using the already submitted MongoDB Cloud registration information.

To manually add a user account:

1. On the **Security Quickstart** page, as shown in Figure 11.8, enter the new username and credentials.

## Security Quickstart

To access data stored in Atlas, you'll need to create users and set up network security controls. [Learn more about security setup](#)

- How would you like to authenticate your connection?

Your first user will have permission to read and write any data in your project.

[Username and Password](#)

[Certificate](#)

**ⓘ** We autogenerated a username and password for your first database user in this project using your MongoDB Cloud registration information. **X**

Create a database user using a username and password. Users will be given the *read and write to any database privilege* by default. You can update these permissions and/or create additional users later. Ensure these credentials are different to your MongoDB Cloud username and password.

Username

theo

Password ⓘ

JRZPlcJJfNS1qF5

[Autogenerate Secure Password](#)

[Copy](#)

[Create User](#)

**Figure 11.8: Security Quickstart Page**

This username and password combination is used to grant a user access to databases and collections in the cluster within the Atlas project.

- To use a password that is auto generated by Atlas, click **Autogenerate Secure Password**.
- Click **Create User**.

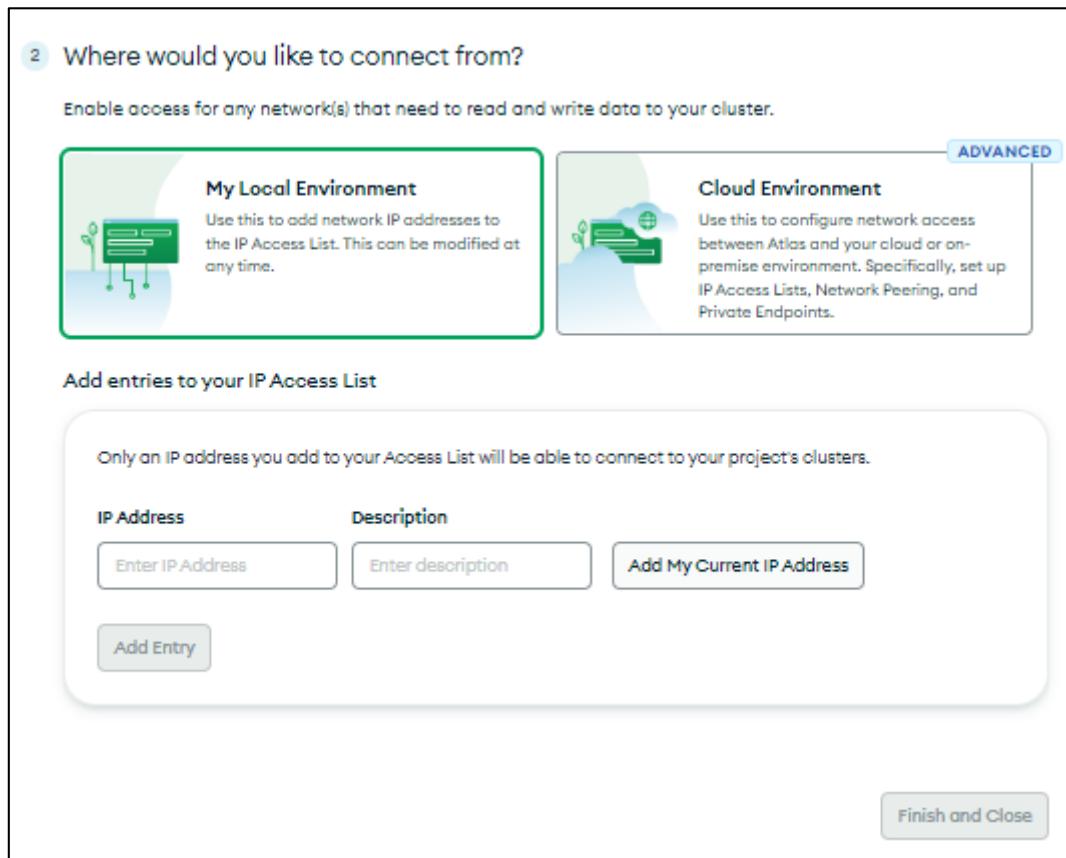
This creates an Atlas user.

### 11.1.4 Add a Trusted IP Address to the Authorized IP Access List

After creating a database user, the deployed cluster is ready to be connected. Each device that connects to a network is assigned a unique identifier called an Internet Protocol or IP address. It is used to identify and locate devices on the network, and to route data between devices.

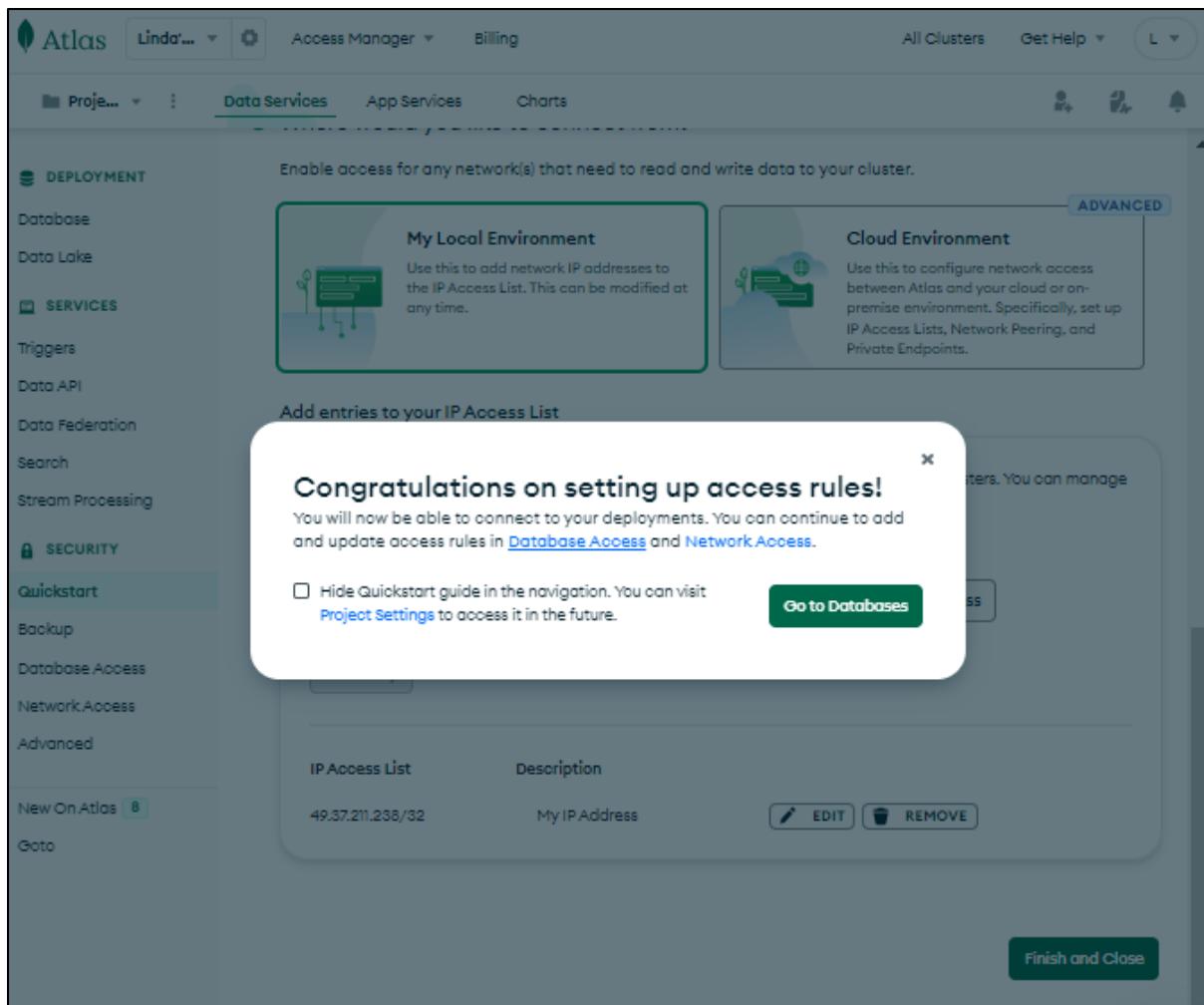
In MongoDB Atlas, an IP access list is a list of trusted IP addresses that are allowed to connect to a MongoDB Atlas cluster. By limiting access to specific IPs, unauthorized connections can be prevented so as to ensure that only trusted sources can connect to the cluster.

To secure the MongoDB Atlas cluster, access should be restricted only to specified IP addresses as shown in Figure 11.9. The current IP address is automatically added. If not added, the user can add it to the IP Access List.



**Figure 11.9: Add entries to your IP Access List Page**

1. To add the current IP address, on this page, click **Add My Current IP Address**.  
The IP Access List is updated with the current IP address.
2. Click **Finish and Close**.  
A success message appears as shown in Figure 11.10.



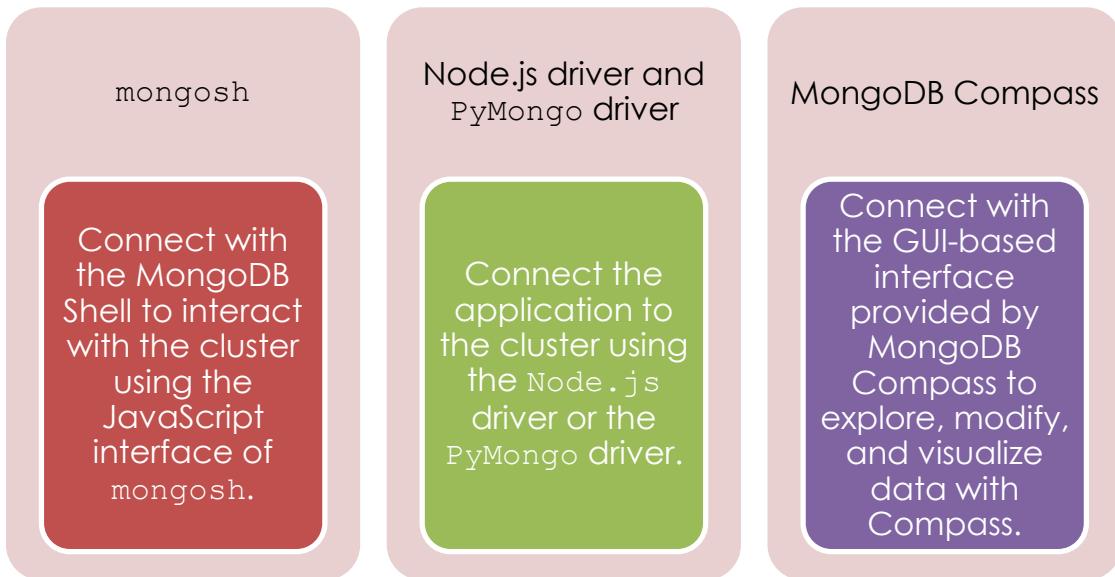
**Figure 11.10: Cluster Provisioning Success Message Page**

3. Click **Go to Databases**.

Cluster provisioning is completed.

### 11.1.5 Connect Mongo Shell to M0 Cluster

Finally, the user must connect to the cluster using one of these methods:



There are a few prerequisites for connecting to the cluster:

- An Atlas account must exist.
- An active cluster must be created in this account.
- An organization with a project must exist.
- An IP address must be added to the IP access list.
- A database user must exist on the cluster.

To connect to the M0 cluster using `mongosh`:

1. Click **Database** in the upper-left corner of the Atlas screen.  
The **Database Deployments** page opens as shown in Figure 11.11.

LINDA'S ORG - 2023-06-30 > PROJECT 0

## Database Deployments

Find a database deployment...

 Load sample datasets to Cluster0.  
Atlas provides sample data you can load into your Atlas clusters. You can use this data to quickly get started exploring with data in MongoDB.

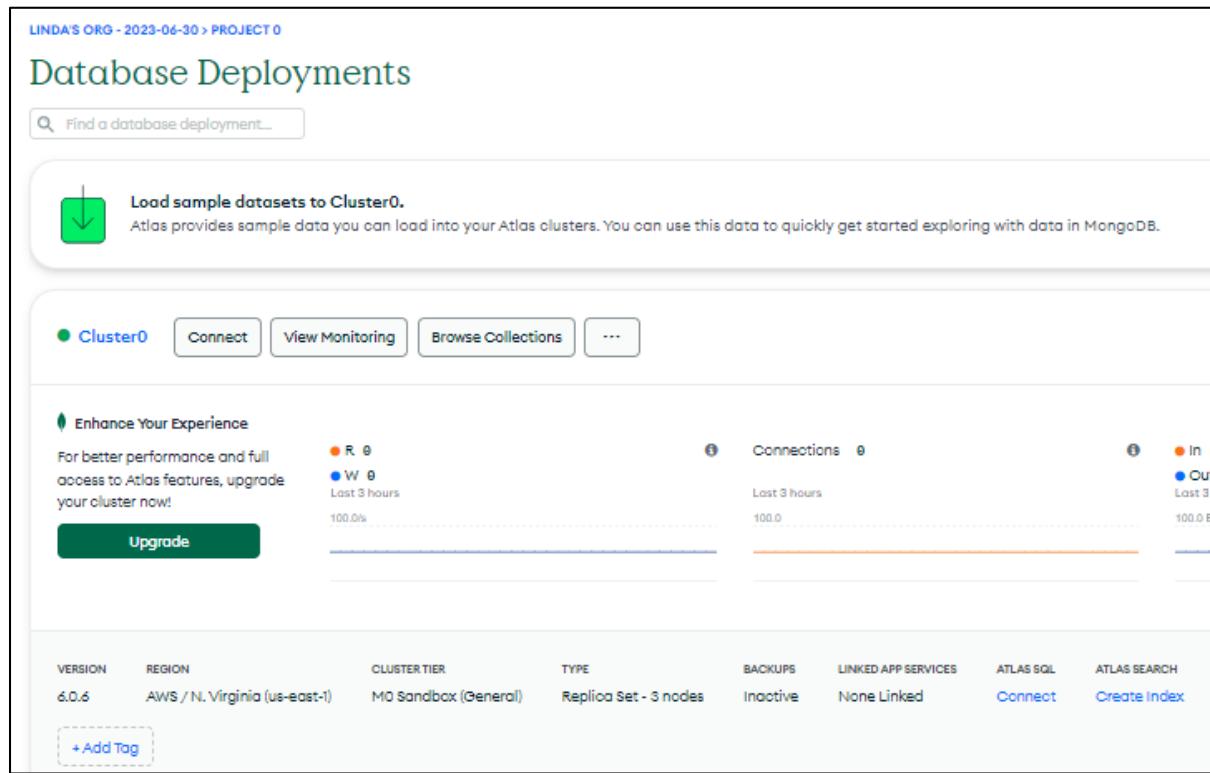
Cluster0 Connect View Monitoring Browse Collections ...

Enhance Your Experience  
For better performance and full access to Atlas features, upgrade your cluster now!  
 Upgrade

R: 0 W: 0 Last 3 hours 100.0% Connections: 0 Last 3 hours 100.0% In: 0 Out: 0 Last 3 hours 100.0%

VERSION	REGION	CLUSTER TIER	TYPE	BACKUPS	LINKED APP SERVICES	ATLAS SQL	ATLAS SEARCH
6.0.6	AWS / N. Virginia (us-east-1)	MO Sandbox (General)	Replica Set - 3 nodes	Inactive	None Linked	Connect	Create Index

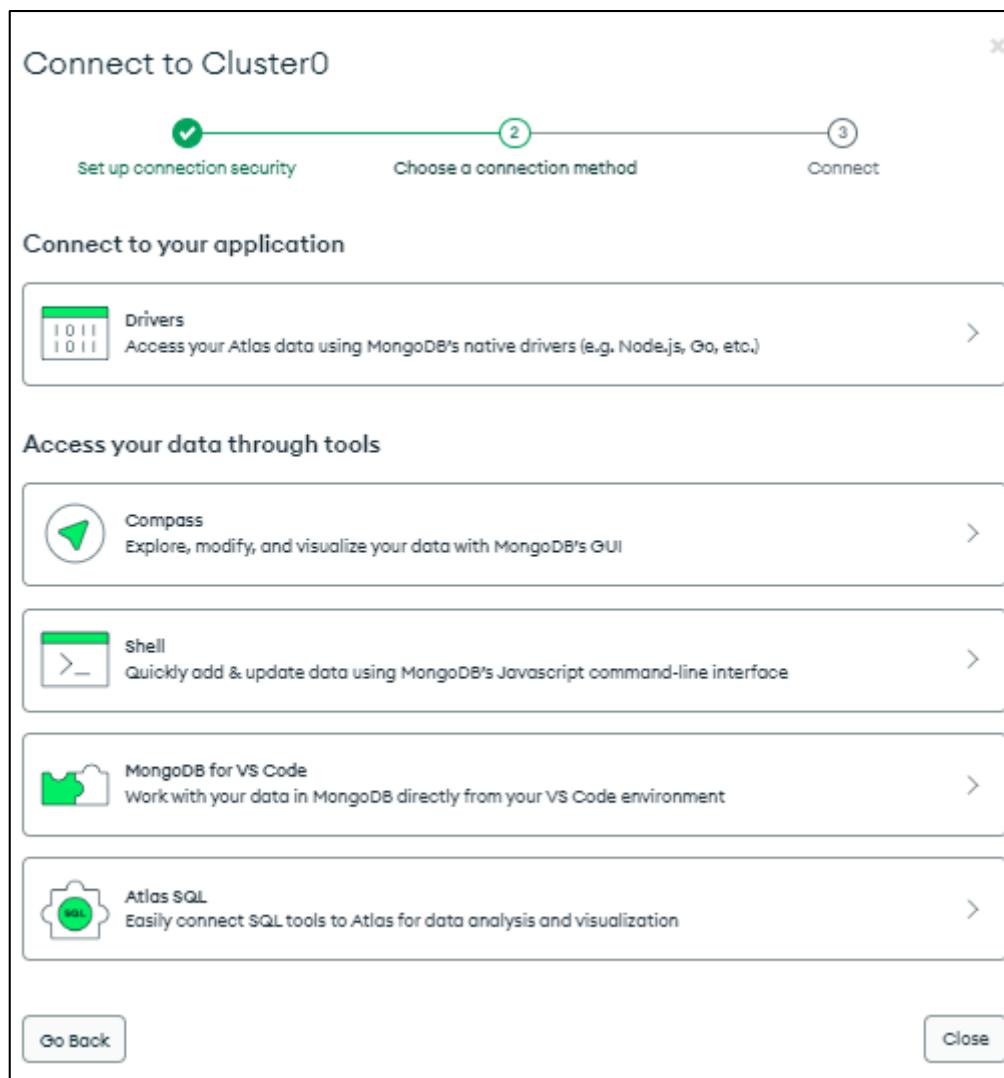
+ Add Tag



**Figure 11.11: Database Deployments Page**

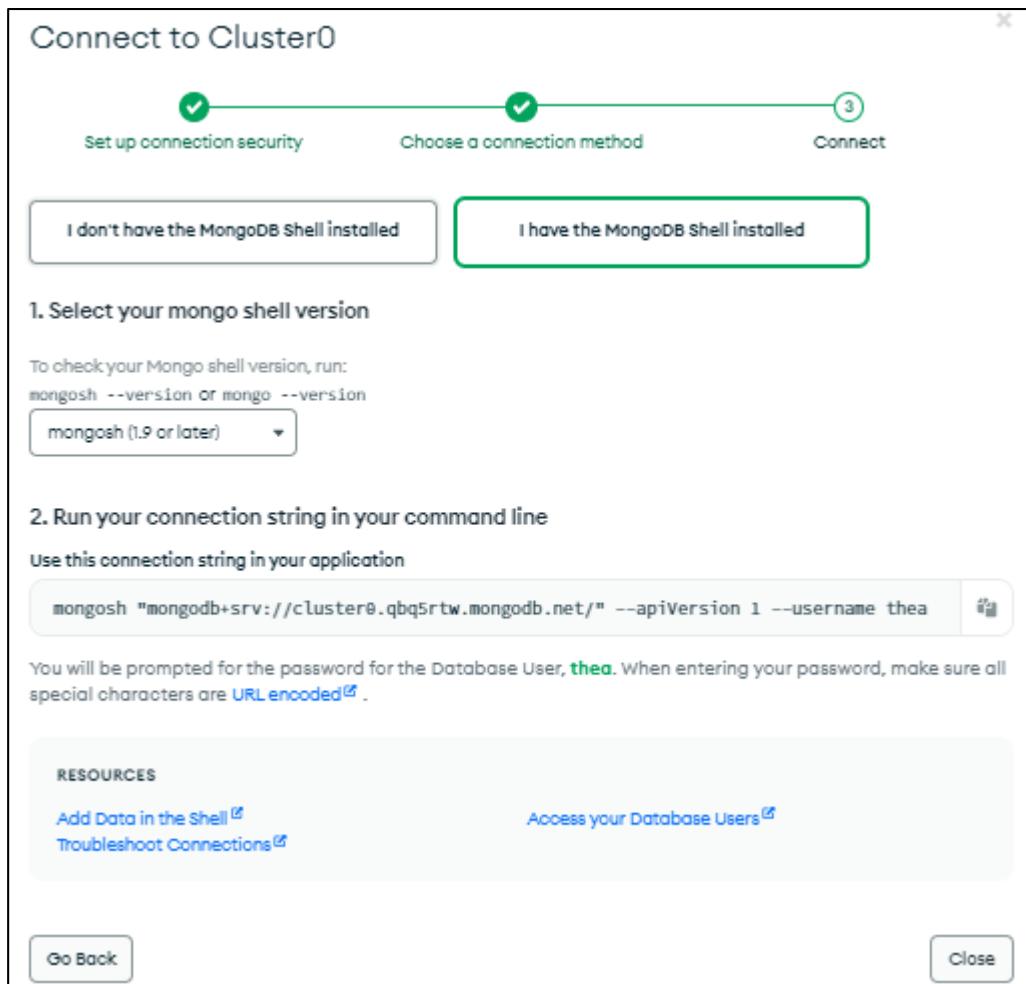
2. To deploy the desired deployment, on this page, click **Connect** for **Cluster0**.

The **Choose a Connection Method** page opens as shown in Figure 11.12.



**Figure 11.12: Choose a Connection Method Page**

3. To use MongoDB Shell, on this page, under **Access your data through tools** section, click **Shell**.  
The **Connect Page** opens as shown in Figure 11.13.
4. As the MongoDB shell is already installed, click **I have the MongoDB Shell installed**.



**Figure 11.13: Connect Page**

5. To use `mongosh`, under **Select your mongo shell version** section, from the drop-down list, select the `mongosh` version. The connection string is created automatically.
6. To use this connection string, from the **Run your connection string in your command line** box, copy the connection string and click **Close** to close the **Connect to Cluster0** dialog box.
7. Open the command prompt. Paste and run the copied connection string as shown in Figure 11.14.

```
C:\Users\Linda>mongosh "mongodb+srv://cluster0.qbq5rtw.mongodb.net/" --apiVersion 1 --username thea
Enter password: *****
```

**Figure 11.14: Starting the Connection**

When prompted, enter the password as specified when creating the Atlas database user.

The command is executed as shown in Figure 11.15.

```
C:\Users\Linda>mongosh "mongodb+srv://cluster0.qbq5rtw.mongodb.net/" --apiVersion 1 --username thea
Enter password: *****
Current Mongosh Log ID: 649ffbd0abfb793124fe7987
Connecting to:      mongodb+srv://<credentials>@cluster0.qbq5rtw.mongodb.net/?appName=mongosh+1.8.2
Using MongoDB:      6.0.6 (API Version 1)
Using Mongosh:      1.8.2

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

Atlas atlas-xihz70-shard-0 [primary] test> |
```

**Figure 11.15: Connected to the Atlas Cluster**

Now, cluster0 is connected within the mongosh. Keep this command prompt open.



In the same way, MongoDB Atlas can be connected with MongoDB Compass using the connection string.

## 11.2 Access MongoDB Atlas Cluster

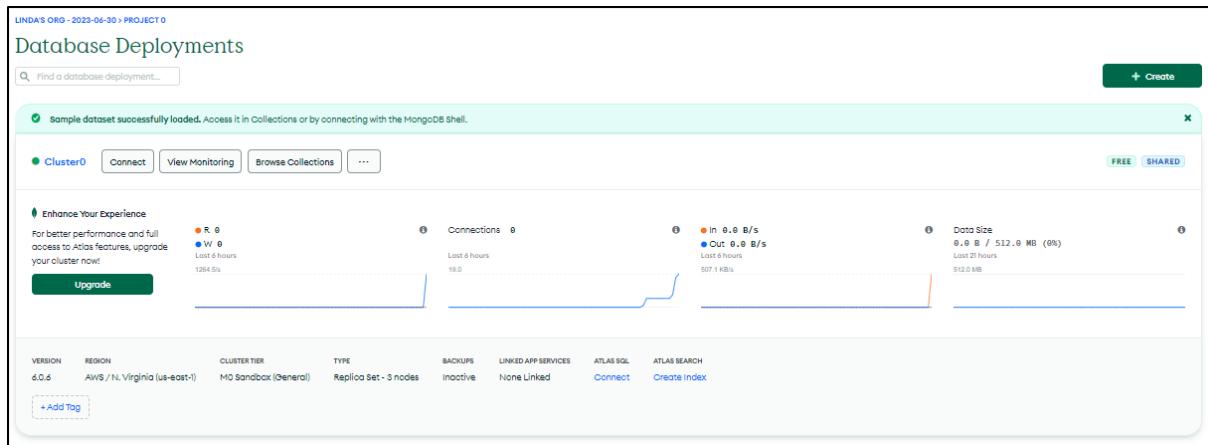
After successfully setting up MongoDB, it is time to start using the database. Atlas provides sample data in a variety of formats, including JavaScript Object Notation (JSON), Comma-Separated Values (CSV), and Tab-Separated Values (TSV). As a beginner to MongoDB, sample data is a great way to learn about MongoDB and see how it can be used to store and manage data.

### 11.2.1 Load and View Sample Data in MongoDB Atlas

Currently, the cluster0 deployed has no databases or collections.

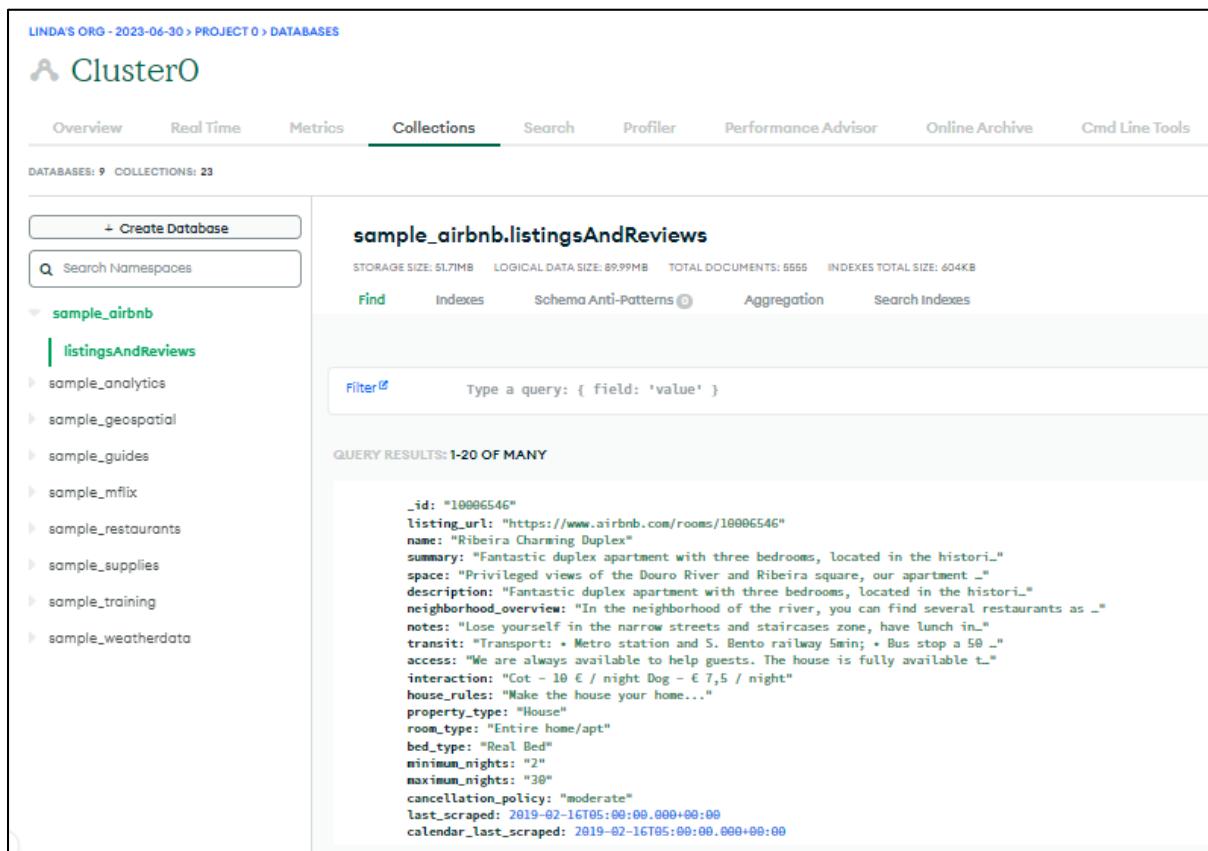
To load a sample dataset into cluster0:

1. If not there already, navigate to the **Database Deployments** page shown in Figure 11.11.
2. To load the available sample datasets, on this page, click **Load sample dataset**. A confirmation message appears when the sample dataset is loaded into cluster0 as shown in Figure 11.16.



**Figure 11.16: Loaded Sample Dataset Confirmation Message Page**

3. To view the sample dataset, on this page, click **Browse Collections**.  
The **Loaded Sample Dataset Lists** page opens as shown in Figure 11.17.



**Figure 11.17: Loaded Sample Dataset Lists Page**

### 11.2.2 View the Sample Data Using Mongo Shell

The loaded sample datasets can be viewed in Mongo Shell.

To view the databases loaded on cluster0 using Mongo Shell:

1. In the command prompt window that runs mongosh, execute the command as:

```
show dbs
```

All sample databases available on the MongoDB Atlas are displayed as shown in Figure 11.18.

```
Atlas atlas-xihz70-shard-0 [primary] test> show dbs
sample_airbnb      52.30 MiB
sample_analytics   9.14 MiB
sample_geospatial  1.35 MiB
sample_guides      40.00 KiB
sample_mflix        120.20 MiB
sample_restaurants  6.57 MiB
sample_supplies     1.12 MiB
sample_training     51.76 MiB
sample_weatherdata  2.75 MiB
admin              372.00 KiB
local              6.17 GiB
Atlas atlas-xihz70-shard-0 [primary] test> |
```

**Figure 11.18: Sample Dataset Page**

All Create, Read, Update, and Delete (CRUD) operations can be performed on the databases, and the changes will be reflected in the databases in cluster0.

### 11.2.3 Import Data From the Local System to MongoDB Atlas

Any database from the local system can be imported to MongoDB Atlas. This is very useful to migrate data from a previous database platform or to import backed up data. Let us import a database named sales which contains the product\_sales collection from the C:\Sample\_dataset folder to MongoDB Atlas.

To import data from a local system:

1. Navigate to the **Database Deployments** page and choose **Command Line Tools** for cluster0 as shown in Figure 11.19.

The screenshot shows the MongoDB Atlas Database Deployments page for cluster0. At the top, there's a success message: "Sample dataset successfully loaded. Access it in Collections or by connecting with the MongoDB Shell." Below this, there are buttons for "Connect", "View Monitoring", "Browse Collections", and a "..." menu. The "Cluster0" button is highlighted with a green dot. A dropdown menu from the "... menu" is open, showing options: "Edit Configuration", "Command Line Tools" (which is selected and highlighted in grey), "Load Sample Dataset", and "Terminate". There's also a "Connections" section showing 0 connections over the last 6 hours. On the left, there's an "Enhance Your Experience" section with a "Upgrade" button. In the center, there are two line charts: one for "Latency (ms)" and one for "Throughput (ops/s)". At the bottom, there's a table with cluster details and a "+ Add Tag" button.

VERSION	REGION	CLUSTER TIER	TYPE	BACKUPS	LINKED APP SERVICES	ATLAS SQL
6.0.6	AWS / N. Virginia (us-east-1)	M0 Sandbox (General)	Replica Set - 3 nodes	Inactive	None Linked	Connect

**Figure 11.19: Database Deployments Page**

The **Cmd Line Tools** page opens as shown in Figure 11.20.

The screenshot shows the MongoDB Atlas interface for Cluster0. The top navigation bar includes links for Overview, Real Time, Metrics, Collections, Search, Profiler, Performance Advisor, Online Archive, and Cmd Line Tools (which is underlined, indicating it's the active tab). Below the navigation, there are three main sections: 'Connect To Your Cluster', 'Manage Your Cluster From the Command Line Interface', and 'Manage Your Cluster From the Command Line'. Each section contains descriptive text and a 'Install' button. The 'Binary Import and Export Tools' section at the bottom contains examples of mongorestore and mongodump commands. A sidebar on the left lists 'LINDA'S ORG - 2023-06-30 > PROJECT 0 > DATABASES'.

LINDA'S ORG - 2023-06-30 > PROJECT 0 > DATABASES

## Cluster0

Overview    Real Time    Metrics    Collections    Search    Profiler    Performance Advisor    Online Archive    **Cmd Line Tools**

### Connect To Your Cluster

Methods to connect your application to your cluster via MongoShell, URI, or Compass can be found in the connect modal.

[Connect Instructions](#)

### Manage Your Cluster From the Command Line Interface

Create and manage MongoDB Atlas resources from your command line and easily automate them using scripts. [Learn more](#)

[Install Atlas CLI](#)

### Manage Your Cluster From the Command Line

Use command line utilities to import and export data, restore backups, and view diagnostics

[Install MongoDB Database Tools](#)

### Binary Import and Export Tools

Replace `PASSWORD` with the password for the admin user and `DATABASE` with the name of the database you wish to import/export to your cluster.

`mongorestore` | creates a new database or adds data to an existing database. By default, `mongorestore` reads the database dump in the `dump` sub-directory of the current directory; to restore from a different directory, pass in the path to the directory as a final argument.

```
mongorestore --uri mongodb+srv://thea:<PASSWORD>@cluster0.qbq5rtw.mongodb.net/
```

`mongodump` | creates a binary export of the contents of a database

```
mongodump --uri mongodb+srv://thea:<PASSWORD>@cluster0.qbq5rtw.mongodb.net/<DATABASE>
```

**Figure 11.20: Cmd Line Tools Page**

2. Scroll down to the **Data Import and Export Tools** section of the page.

The **Data Import and Export Tools** section appears as shown in Figure 11.21.

### Data Import and Export Tools

Replace **PASSWORD** with the password for the admin user, **DATABASE** with the name of the database you wish to import/export to your cluster, and **COLLECTION** with the name of the collection you wish to import/export to your cluster. Replace **FILETYPE** with "json" or "csv" to specify the file type. Where applicable, replace **FILENAME** with the location and name of the output file (for export) or data source (for import).

NOTE: When exporting or importing CSV data, an additional --fields flag is often required. See documentation for the specific tool for additional details.

[mongoimport](#) | imports content from an Extended JSON, CSV, or TSV export

```
mongoimport --uri mongodb+srv://thea:<PASSWORD>@cluster0.qbq5rtw.mongodb.net/<DATABASE> --collection <COLLECTION> --type <FILETYPE> --file <FILENAME>
```

[mongoexport](#) | produces a JSON or CSV export of data stored in a MongoDB instance

```
mongoexport --uri mongodb+srv://thea:<PASSWORD>@cluster0.qbq5rtw.mongodb.net/<DATABASE> --collection <COLLECTION> --type <FILETYPE> --out <FILENAME>
```

**Figure 11.21: Data Import and Export Tools Section**

3. From the **Data Import and Export Tools** section, copy the mongoimport connection string template to the clipboard.

```
mongoimport --uri  
mongodb+srv://<USERNAME>:<PASSWORD>@cluster0.qbq5rtw.mongodb.net/<DATABASE> --collection <COLLECTION> --type  
<FILETYPE> --file <FILENAME>
```

The connection string template includes placeholder values for certain options. These placeholders must be replaced with the appropriate values for the Atlas cluster as described in Table 11.1.

Placeholder	Values
<PASSWORD>	<p>Replace with the password for the user specified in --username</p> <p>The template includes the database user for the project as the --username. To authenticate as a different user, replace the value of --username and specify the password for that user in --password.</p> <p>Enclose the password within double or single quotes. For example, if the password is @bc123, it must be surrounded within quotes, such as "@bc123".</p>

Placeholder	Values
<DATABASE>	Replace with the name of the database to which data is being imported.
<COLLECTION>	Replace with the name of the collection to which data is being imported.
<FILETYPE>	Replace with the file type of the data source from which data is being imported.
<FILENAME>	Replace with the name of the data source from which data is being imported.

**Table 11.1: Connection String Template Values**

Table 11.1 does not include <USERNAME> as the connection string will automatically contain the username with which you logged in to Atlas.

4. To import the sales database with product\_sales collection, open a new command prompt.
5. Run the connection string command as:

```
mongoimport --uri
mongodb+srv://thea:tea99@cluster0.qbq5rtw.mongodb.net/sales
--collection product_sales --type json --file
C:\Sample_dataset\sales\product_sales.json
```

The command executes as shown in Figure 11.22.

```
C:\Users\Linda>mongoimport --uri mongodb+srv://thea:tea99@cluster0.qbq5rtw.
mongodb.net/sales --collection product_sales --type json --file C:\Sample_d
ataset\sales\product_sales.json
2023-07-01T21:58:14.135+0530      connected to: mongodb+srv://[**REDACTED**]@
cluster0.qbq5rtw.mongodb.net/sales
2023-07-01T21:58:14.712+0530      3 document(s) imported successfully. 0 docu
ment(s) failed to import.
```

**Figure 11.22: Imported Sales Data**

A message shows that the three documents from the local system are imported to the MongoDB Atlas successfully.

6. To view the imported sales database, open the command prompt window that runs mongosh and execute the command as:

```
show dbs
```

The sales database is displayed as shown in Figure 11.23.

```
Atlas atlas-xihz70-shard-0 [primary] test> show dbs
sales          40.00 KiB
sample_airbnb   52.36 MiB
sample_analytics 9.14 MiB
sample_geospatial 1.35 MiB
sample_guides    40.00 KiB
sample_mflix     120.28 MiB
sample_restaurants 6.57 MiB
sample_supplies   1.12 MiB
sample_training   51.78 MiB
sample_weatherdata 2.75 MiB
admin           372.00 KiB
local            6.17 GiB
Atlas atlas-xihz70-shard-0 [primary] test> |
```

**Figure 11.23: Imported Sales Database**

7. To view more details, switch to sales database by executing the command:

```
use sales
```

The command is executed as shown in Figure 11.24.

```
Atlas atlas-xihz70-shard-0 [primary] test> use sales
switched to db sales
Atlas atlas-xihz70-shard-0 [primary] sales>
```

**Figure 11.24: Switched to Sales Database**

8. To view the documents of product\_sales collection, execute the command as:

```
db.product_sales.find()
```

The command is executed as shown in Figure 11.25.

```
Atlas atlas-xihz70-shard-0 [primary] sales> db.product_sales.find()
[
  {
    _id: ObjectId("64a0541e43840751334bbc80"),
    product_id: 338,
    sales_person: 'Amelia',
    products_sold: [ 4, 8, 9, 6 ]
  },
  {
    _id: ObjectId("64a0541e43840751334bbc81"),
    product_id: 302,
    sales_person: 'Smith',
    products_sold: [ 5, 3, 7, 2 ]
  },
  {
    _id: ObjectId("64a0541e43840751334bbc82"),
    product_id: 371,
    sales_person: 'Robert',
    products_sold: [ 3, 8, 6, 5 ]
  }
]
Atlas atlas-xihz70-shard-0 [primary] sales> |
```

**Figure 11.25: product\_sales Collection Database**

#### 11.2.4 Export Data from Atlas Cluster to the Local System

Data from the Atlas cluster can also be exported to the local system for backup, offline analysis, sharing, or troubleshooting. Let us export the sample\_airbnb database from MongoDB Atlas to the C:\Sample\_dataset folder for offline analysis.

To export data to the local system:

1. Navigate to the **Database Deployments** page.
2. For **Cluster0**, click the three dots, and then click **Command Line Tools**.
3. Scroll down to the **Data Import and Export Tools** section as shown in Figure 11.26.

## Data Import and Export Tools

Replace `PASSWORD` with the password for the admin user, `DATABASE` with the name of the database you wish to import/export to your cluster, and `COLLECTION` with the name of the collection you wish to import/export to your cluster. Replace `FILETYPE` with "json" or "csv" to specify the file type. Where applicable, replace `FILENAME` with the location and name of the output file (for export) or data source (for import).

**NOTE:** When exporting or importing CSV data, an additional `--fields` flag is often required. See documentation for the specific tool for additional details.

`mongoimport` | imports content from an Extended JSON, CSV, or TSV export

```
mongoimport --uri mongodb+srv://thea:<PASSWORD>@cluster0.qbq5rtw.mongodb.net/<DATABASE> --collection <COLLECTION> --type <FILETYPE> --file <FILENAME>
```

`mongoexport` | produces a JSON or CSV export of data stored in a MongoDB instance

```
mongoexport --uri mongodb+srv://thea:<PASSWORD>@cluster0.qbq5rtw.mongodb.net/<DATABASE> --collection <COLLECTION> --type <FILETYPE> --out <FILENAME>
```

**Figure 11.26: Data Import and Export Tools Section**

1. From the **Data Import and Export Tools** section, copy the `mongoexport` connection string to the clipboard.

```
mongoexport --uri  
mongodb+srv://thea:<PASSWORD>@cluster0.qbq5rtw.mongodb.net/<DATABASE> --collection <COLLECTION> --type <FILETYPE> --  
out <FILENAME>
```

The connection string template includes placeholder values for certain options. These placeholders must be replaced with the appropriate values for the Atlas cluster as described in Table 11.1.

2. To export the `sample_airbnb` database with `listingsAndReviews` collection, open a new command prompt and execute the command:

```
mongoexport --uri  
mongodb+srv://thea:tea99@cluster0.qbq5rtw.mongodb.net/sample_airbnb --collection listingsAndReviews --type json --out  
C:\Sample_dataset\airbnb_listings.json
```

The command executes as shown in Figure 11.27.

```
C:\Users\Linda>mongoexport --uri mongodb+srv://thea:tea99@cluster0.qbq5rtw.mongodb.net/sample_airbnb --collection listingsAndReviews --type json --out C:\Sample_dataset\airbnb_listings.json
2023-07-01T22:24:32.076+0530      connected to: mongodb+srv://[**REDACTED**]@cluster0.qbq5rtw.mongodb.net/sample_airbnb
2023-07-01T22:24:33.303+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:34.305+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:35.297+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:36.298+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:37.304+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:38.292+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:39.302+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:40.303+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:41.305+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:42.299+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:43.298+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:44.302+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:45.304+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:46.295+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:47.293+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:48.299+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:49.292+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:50.292+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:51.305+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:52.297+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:53.293+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:54.292+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:55.292+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:56.294+0530      [.....] sample_airbnb.listingsAndReviews 0/5555 (0.0%)
2023-07-01T22:24:57.244+0530      [#####] sample_airbnb.listingsAndReviews 5555/5555 (100.0%)
2023-07-01T22:24:57.244+0530      exported 5555 records
```

**Figure 11.27: Exported sample\_airbnb Database**

A total of 5555 records from the sample\_airbnb database is exported into the local system successfully. The file will be displayed in the local system as shown in Figure 11.28.

Name	Date modified	Type
sales	7/1/2023 7:30 PM	File folder
sample_analytics	5/30/2023 11:14 AM	File folder
sample_geospatial	5/30/2023 11:14 AM	File folder
sample_guides	5/30/2023 11:14 AM	File folder
sample_mflix	5/30/2023 11:14 AM	File folder
sample_restaurants	5/30/2023 11:14 AM	File folder
sample_supplies	5/30/2023 11:14 AM	File folder
sample_training	5/30/2023 11:14 AM	File folder
sample_weatherdata	5/30/2023 11:14 AM	File folder
airbnb_listings	7/1/2023 10:24 PM	JSON Source File

**Figure 11.28: Exported `airbnb_listings.json` in the Local System**

### 11.3 Administering MongoDB Cluster

MongoDB clusters are complex systems that require regular administration to ensure that they are running smoothly and efficiently. The clusters can be:

Paused	Resumed	Terminated
<ul style="list-style-type: none"> <li>Temporarily disable the cluster and prevent users from accessing the data</li> <li>Useful for maintenance or troubleshooting purposes</li> </ul>	<ul style="list-style-type: none"> <li>Bring the cluster back online and make the data available to users</li> </ul>	<ul style="list-style-type: none"> <li>Permanently delete the cluster and all of its data</li> <li>Should only be done as a last resort, such as when the cluster is no longer required or when there is a serious problem with the cluster</li> </ul>

### 11.3.1 Pause a Cluster

MongoDB Atlas automatically pauses all inactive M0, M2, and M5 clusters after 60 days of inactivity. For clusters other than M0, M2, and M5 type, users can manually initiate a pause.

When a cluster is paused in Atlas, the user cannot modify the cluster configuration and cannot read or write data to the cluster. Atlas stops collecting monitoring information for these clusters completely and does not allow any connections to the cluster until the cluster is resumed. Configured alerts will no longer trigger. All backups will be stopped, but any existing snapshots will be retained until they expire.

Prior to pausing, Atlas sends an email notification seven days in advance and another email after the pause.



---

Atlas does not allow the user to initiate a pause on M0 cluster. However, an automatically paused M0 cluster can be resumed or terminated at any time.

---

### Backup Compliance Policy

A Backup Compliance Policy is a set of rules that govern how backups are created and managed for MongoDB Atlas clusters. This policy specifies details such as frequency of backups, the retention period for backups, and the encryption used for backups.



---

If a paused cluster does not have the **Encryption at Rest** option enabled, the **Require Encryption at Rest using Customer Key Management for all clusters** option cannot be toggled to **ON** in a Backup Compliance Policy.

---

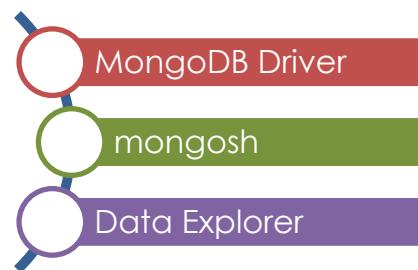
In general, pausing a cluster will NOT affect the existing backups that have been created for the cluster. However, if the cluster is resumed and the Backup Compliance Policy has the Continuous Cloud Backup option enabled, then Atlas will create a new backup for the cluster.

### 11.3.2 Resume a Cluster with Backup Compliance Policy

When **Backup Compliance Policy** is enabled, resuming a cluster in Atlas automatically enables Cloud backup. If the **Require Point in Time Restore to all clusters** option is set to **ON**, Atlas will enable Continuous Cloud Backup and adjust the restore window according to the Backup Compliance Policy.

Additionally, Atlas will automatically modify the backup to meet the minimum requirements specified by the Backup Compliance Policy.

To resume collecting monitoring information for an Atlas M0 cluster that has been paused for monitoring, connections can be made to the cluster using one of the three options:



To resume a paused cluster, navigate to the **Database Deployments** page and choose **Resume** for the desired cluster. Note that this option will be enabled only when the cluster is paused.

### 11.3.3 Terminate a Cluster

To terminate a cluster:

1. Navigate to the **Database Deployments** page and choose **Terminate** for the desired cluster as shown in Figure 11.29.

## Database Deployments

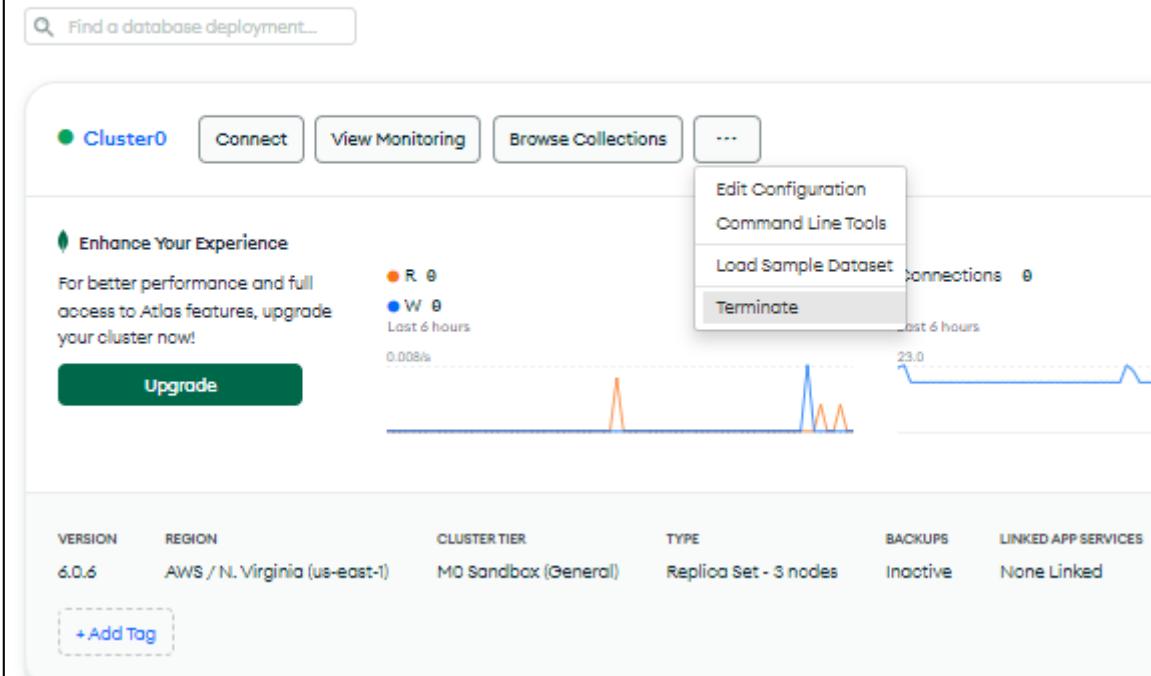


Figure 11.29: Database Deployments Page



Terminating a cluster will also delete all backup snapshots for that cluster.

## **11.4 Summary**

---

- To get started with MongoDB Atlas, begin to create an Atlas account, set up a cluster and then, create a user for that cluster.
- MongoDB Atlas provides sample datasets to load into the Atlas database deployments.
- A dataset can be imported from the local system into the MongoDB Atlas using the `mongoimport` command.
- A dataset can be exported to the local system from the MongoDB Atlas Database using the `mongoexport` command.
- As part of administration tasks, the MongoDB clusters can be paused, resumed, and terminated.

## Test Your Knowledge

---

1. Which of the following connection methods does MongoDB Atlas provide to connect your clusters with deployment?
  - a. Connect via Compass
  - b. Connect via mongosh
  - c. Connect via Application
  - d. Connect via VS Code
2. Which of the following service providers are supported by Atlas for M0-free clusters?
  - a. Amazon Web Services
  - b. Google Cloud Platform
  - c. IBM Cloud
  - d. Microsoft Azure
3. Which of the following cluster is a free cluster used for learning and exploring MongoDB in a cloud environment?
  - a. M10
  - b. Serverless
  - c. M0
  - d. M9
4. Which of the following statements is not true about database users for clusters in MongoDB Atlas?
  - a. It is necessary to create a database user to access the cluster.
  - b. Atlas allows you to create a database user only if you are an **Organization Owner** or a **Project Owner**.
  - c. Database users are separate from Atlas users.
  - d. Atlas users can access MongoDB databases.
5. Which of the following statements is true about pausing a cluster in MongoDB Atlas?
  - a. It is not possible to initiate a pause for M0 clusters.
  - b. It is not possible to resume or terminate an automatically paused cluster at any time.
  - c. After 60 days of inactivity in a cluster, Atlas automatically pauses that cluster.
  - d. It is possible to pause M10 or larger clusters.

## **Answers to Test Your Knowledge**

---

1	a, b, c, d
2	a, b, d
3	C
4	D
5	C

## Try it Yourself

---

1. In MongoDB Atlas, perform the given tasks:
  - a. Create an Atlas account.
  - b. Deploy a free M0 cluster.
  - c. Add the current IP address to the cluster IP access list.
  - d. Create a database user for your cluster.
  - e. Connect MongoDB mongosh to your Atlas cluster using the connection string.
2. Load the sample dataset in MongoDB Atlas cluster and view the loaded datasets.
3. Create a database named Customer\_detail and a collection named Customer\_personal in your local system. The documents in the collection are:

```
[  
  {  
    Customer_id:94608  
    Customer_name:Richard David  
    Address:5858 Horton Street, USA  
    Gender:Male  
  },  
  {  
    Customer_id:94590  
    Customer_name:Rita Edward  
    Address:401 Quarry Road, Stanford,USA  
    Gender:Female  
  },  
  {  
    Customer_id:94305  
    Customer_name:Adam Smith  
    Address:#170, Emeryville valley, USA  
    Gender:Male  
  }  
]
```

4. Import the Customer\_detail database and the Customer\_personal collection from the local system to the MongoDB Atlas cluster.
5. Export the sample\_mflix database which is loaded as part of the sample dataset from the MongoDB Atlas cluster to your Local system.