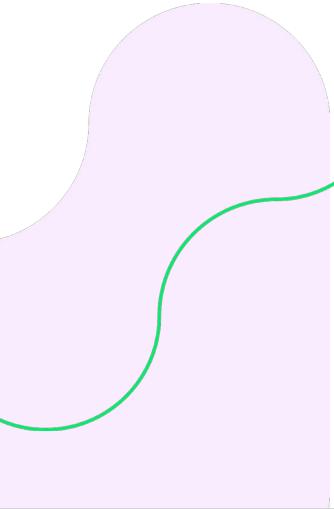




DF100

# Accessing MongoDB

MongoDB Developer Fundamentals



# Topics we cover

- 
- The MongoDB Database
  - Atlas Managed MongoDB Clusters
  - Launch an Atlas Cluster
  - Configure Atlas Cluster Security
  - Connect to MongoDB with mongosh
  - Useful basic commands
  - Other tools



# MongoDB Database

MongoDB's original product: A Document model

- Similarities to RDBMS
- BSON not JSON - Data stored as binary typed objects
- Container types: Document and Array
- Fewer 'tables', more denormalization
- 3rd normal form isn't optimal in many cases
- Optimized for Availability, Usability, Scaling, and Speed
- Idiomatic development drivers

Recap

Emphasis on :

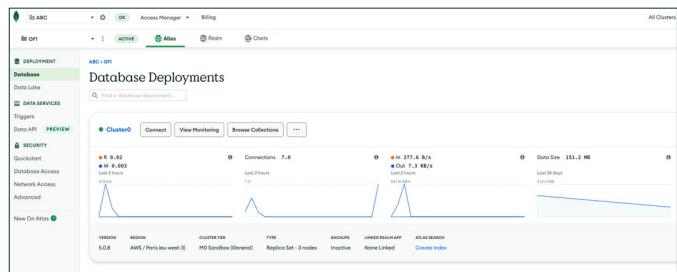
- Terminology understanding - Documents / Collections / Databases - Database.Collection  
= Namespace
- Denormalization is "normal" but 3rd normal form not optimal is many cases



# Atlas Managed Clusters

MongoDB Atlas is MongoDB as a Service.

- Highly available, scalable clusters in AWS/GCP/Azure
- Global availability and scaling
- Development and Business focus
- Manage Database infrastructure



M0 clusters are free sandbox replica set clusters. You can deploy one M0 cluster per Atlas project.

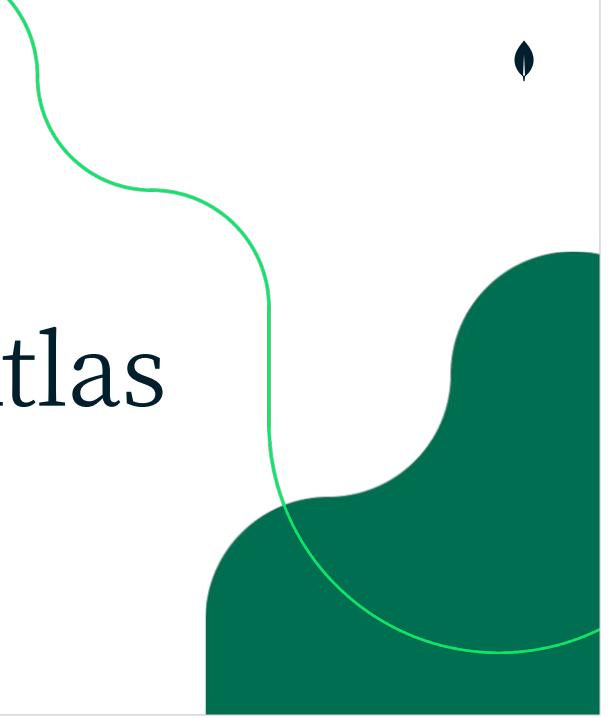
If you do not have an account / corporate login, you can create one:

<https://www.mongodb.com/cloud>

We are going to configure a 3-node replica set for this training.

A Node is synonymous to an instance or a single MongoDB server.

A Cluster is a group of nodes.



Launch an Atlas  
cluster





# Setting-up MongoDB Atlas

We will set up an Atlas Free Tier cluster

- 3 Nodes in different datacenters
- Highly available
- 512 MB of storage
- Secure by default
- Using TLS network encryption

Atlas free tier is on a shared server.

Some functionalities are not available/limited (throughput, number of connections, data transfer limits, monitoring, alerting, API access, etc.)

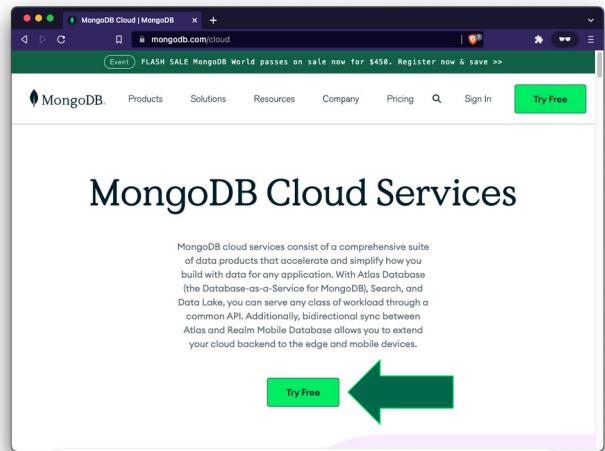
It is technically large enough to run most small business applications and is free for life.



# Go to MongoDB Cloud

Navigate to [cloud.mongodb.com](https://cloud.mongodb.com)

Try Free



Navigate to <https://www.mongodb.com/cloud> in a browser (ideally Chrome).

Click the button **Try MongoDB Cloud Now.**

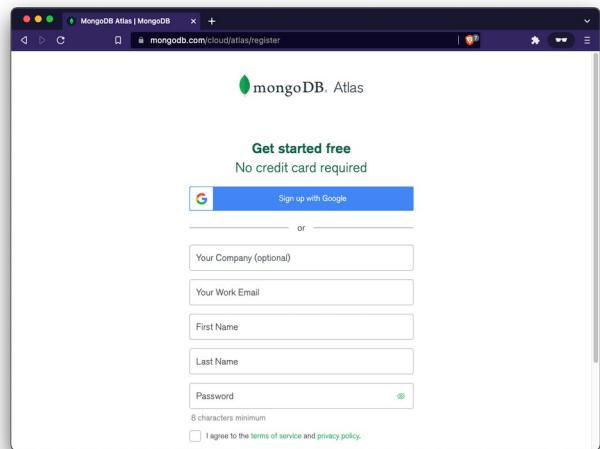


# Sign up or Sign in

Fill details

Complete the Captcha

Confirm Email



A screenshot of a web browser showing the MongoDB Atlas sign-up page. The URL in the address bar is `mongodb.com/cloud/atlas/register`. The page features the MongoDB logo and the text "Get started free" with the note "No credit card required". It includes a "Sign up with Google" button and several input fields for "Your Company (optional)", "Your Work Email", "First Name", "Last Name", and "Password". A note indicates "8 characters minimum". At the bottom, there is a checkbox for agreeing to the "terms of service and privacy policy".

Sign up with your name and email



# Organization and Project

Create an Organization

Create a Project

The screenshot shows a web browser window titled "Create Organization | MongoDB". The URL is "cloud.mongodb.com/v2#/preferences/organizations/create". The left sidebar has a "PREFERENCES" section with "Organizations" selected. The main content area is titled "Create Organization" with tabs "Name and Service" and "Add Members". A "Next" button is visible. The "Name Your Organization" field contains "OrgName". The "Select Cloud Service" section compares "MongoDB Atlas" and "Cloud Manager" across several features:

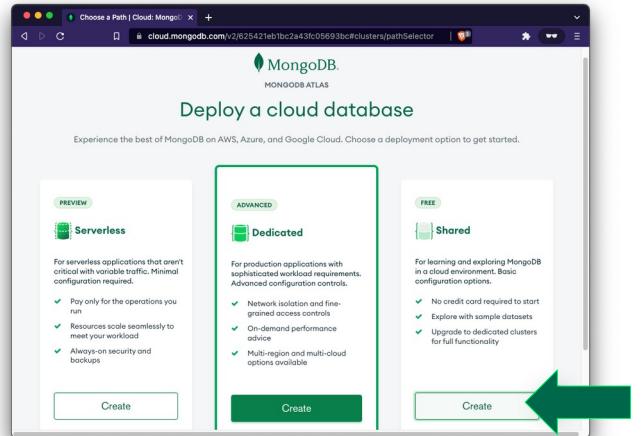
Features	MongoDB Atlas	Cloud Manager
Automated database configuration	✓	✓
Continuous backup and point-in-time recovery	✓	✓
Queryable backup snapshots	✓	✓
Fine grained database monitoring &	✓	✓



# Choose the Free Cluster

Build New Cluster

Free Shared Cluster (M0)



Clusters can be Shared (small), Dedicated (production-ready), or very large indeed and/or distributed globally.

Atlas can go very, very large. You could have a cluster of 50 shards (data partitioned for scaling out) with 768GB of RAM each, 4TB of storage, and 96 CPU's each of which has 7-way replication (350 Massive servers). Of course, this costs \$4200 an hour (approx)

Potential to go larger still if required.

Pick the free one (or similar if you already have an account) - you may not get this screen; if not, click 'Build a New Cluster' to get to the next page.

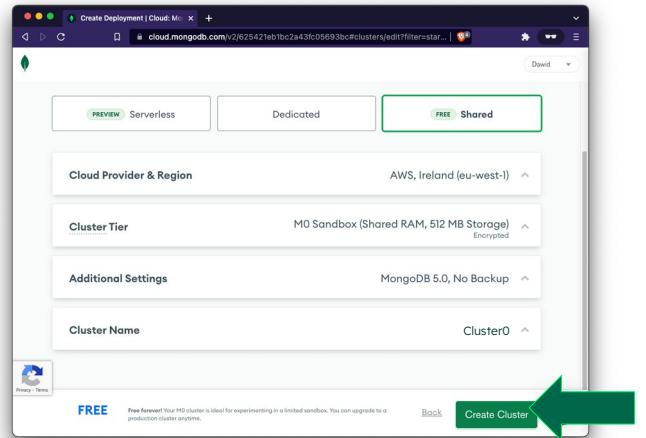
# Create the Cluster

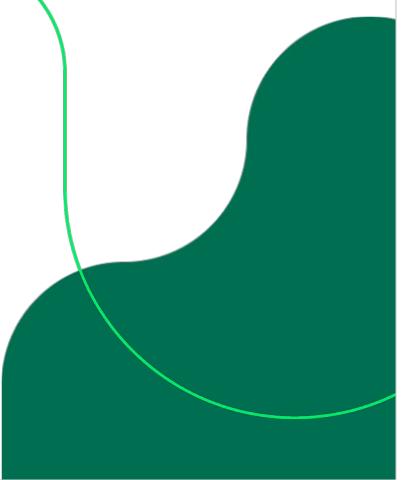
**Provider:** AWS

**Region:** Ireland (eu-west-1)

**Version:** Latest available

**Cluster Name:** Cluster0





Configure Atlas  
cluster security



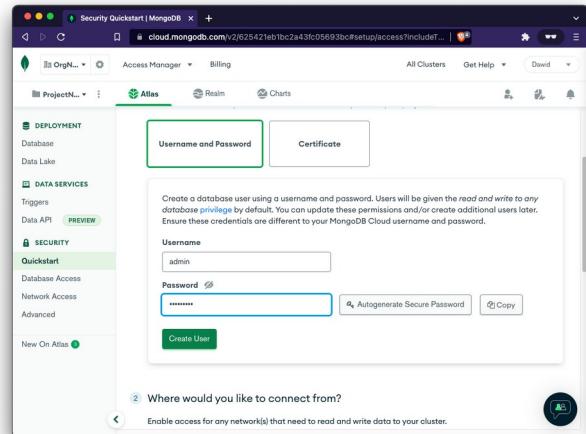


# Add a user (Quick Start)

**User:** admin

**Password:** qwerty123

**NOTE:** Never use bad passwords for any public facing databases! The above is for training purposes only.



Add a database user with:

- USERNAME: admin
- PASSWORD: qwerty123

Click **Create Database User** and then click **Choose a connection method** to go to the next step.

Atlas has a lot of default security mechanisms in place:

- We need users created
- We have to use TLS
- Data is on encrypted disks
- We need to add IPs to the Access List



# Network Setup (Quick Start)

## Add Strigo IP Address

The screenshot shows the MongoDB Atlas Security Quickstart interface. The left sidebar has sections for Deployment, Data Services, and Security. Under Security, 'Quickstart' is selected. The main area is titled 'IP Access List' with a sub-section 'My Local Environment'. It explains how to add network IP addresses to the IP Access List. Below this is another section 'Cloud Environment' with similar instructions. At the bottom, there's a table to 'Add entries to your IP Access List' with columns for IP Address and Description. An entry for '123.123.123.123' with the description 'My Local IP' is shown, along with buttons for 'Add Entry' and 'Add My Current IP Address'.

### Click Connect

Atlas requires us to specify what hosts can connect to our cluster for security reasons. Ideally, it is good to always select **Add a Different IP Address**.

We can add the Strigo public IP here, we can get it by running `curl ifconfig.me` on the Strigo Terminal.



# Wait for cluster to start up

Can take upto 5 minutes.

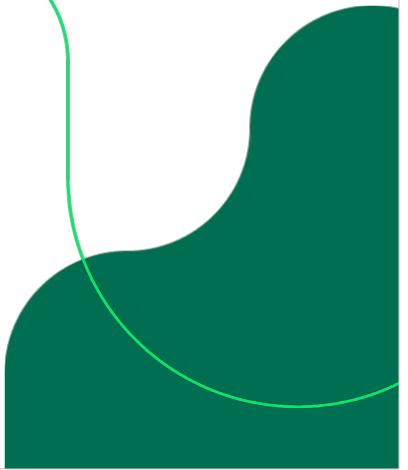


Wait for the cluster to be created (up to 5 mins approx) - Wait for the blue bar at the top to disappear

If you are redirected to the quickstart wizard, go back to the Databases page shown above as we would set up users later.



# Connect to MongoDB with mongosh





# Connect to the cluster

The screenshot shows the MongoDB Cloud interface for managing database deployments. On the left, there's a sidebar with sections like Deployment, Data Services, and Security. The main area is titled "Database Deployments" and shows a summary of a cluster. A green arrow points to the "View Monitoring" button, which is part of a row of buttons including "Cluster", "Connect", "View Monitoring", "Browse Collections", and "...". Below these buttons, there are performance metrics: R: 6, W: 9, Last 6 minutes; In: 6.0 B/s, Out: 6.0 B/s, Last 6 minutes; and Data Size: 0.0 B, Last 6 minutes. At the bottom, there's a table with columns: VERSION, REGION, CLUSTER TIER, TYPE, BACKUP, and LINKED REALM API. The cluster details shown are: VERSION 5.0.6, REGION AWS / Ireland (eu-west-1), CLUSTER TIER M0 Sandbox (General), TYPE Replica Set - 3 nodes, BACKUP Inactive, and LINKED REALM API None Linked.



# Network Setup (No Quick Start)

## Add Strigo IP Address

The screenshot shows the 'Add Strigo IP Address' setup page. At the top, there are three navigation buttons: 'Setup connection security' (highlighted in blue), 'Choose a connection method', and 'Connect'. Below these, a message states: 'You need to secure your MongoDB Atlas cluster before you can use it. Set which users and IP addresses can access your cluster now.' with a 'Read more' link. A yellow box at the bottom left says 'You can't connect yet. Set up your firewall access and user security permission below.' The main area has two sections: '1 Add a connection IP address' (with buttons for 'Add Your Current IP Address' (highlighted in green), 'Add a Different IP Address', and 'Allow Access from Anywhere') and '2 Create a Database User' (with fields for 'Username' (ex. dbUser) and 'Password' (ex. dbUserPassword, SHOW, Autogenerate Secure Password, Copy)).

### Click Connect

Atlas requires us to specify what hosts can connect to our cluster for security reasons. Ideally, it is good to always select **Add a Different IP Address**.

We can add the Strigo public IP here, we can get it by running `curl ifconfig.me` on the Strigo Terminal.



# Add a user (No Quick Start)

**User:** admin

**Password:** qwerty123

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 user security permission below.

① Add a connection IP address  
✓ An IP address has been added to the IP Access List. Add another address in the IP Access List tab.

② Create a Database User

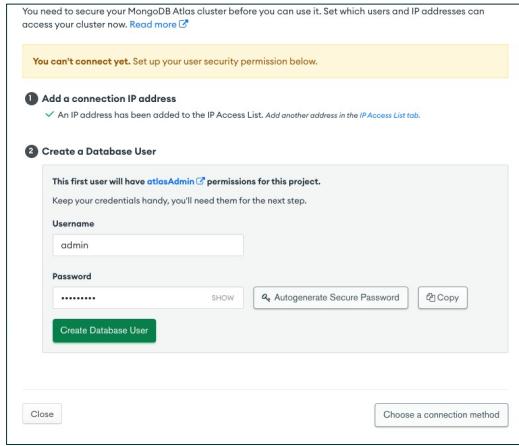
This first user will have [atlasAdmin](#) permissions for this project.  
Keep your credentials handy, you'll need them for the next step.

Username: admin

Password:  [SHOW](#) [Autogenerate Secure Password](#) [Copy](#)

[Create Database User](#)

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



Add a database user with:

- USERNAME: admin
- PASSWORD: qwerty123

Click **Create Database User** and then click **Choose a connection method** to go to the next step.

Atlas has a lot of default security mechanisms in place:

- We need users created
- We have to use TLS
- Data is on encrypted disks
- We need to add IPs to the Access List



# Connection method

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're ready to connect.** Choose how you want to connect in the next step.

**1 Add a connection IP address**  
✓ An IP address has been added to the IP Access List. [Add another address in the IP Access List tab.](#)

**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](#)



# Connect with mongosh

MongoDB shell (mongosh) -  
Javascript Read Evaluate Print Loop  
(REPL)

MongoDB Compass

The screenshot shows a step-by-step connection setup process. Step 1: 'Setup connection security' (done), 'Choose a connection method' (selected), 'Connect'. Step 2: 'Choose a connection method' (View documentation). Step 3: 'Get your pre-formatted connection string by selecting your tool below.' Three options are listed: '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), and 'Connect using MongoDB Compass' (Explore, modify, and visualize your data with MongoDB's GUI). At the bottom are 'Go Back' and 'Close' buttons.

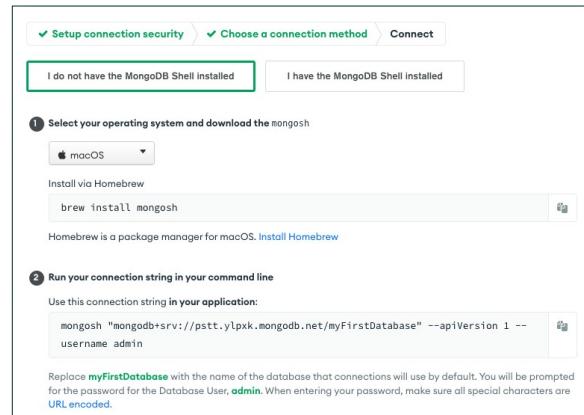
Now we have the cluster configured; applications can connect.  
One application is **mongosh** - a javascript REPL (Read Eval Print Loop) we will use to learn some basic MongoDB concepts.  
Details are also provided for connecting with a GUI tool such as Compass or a programming language driver.



# Install mongosh

If you are working in your own environment

- Specify Operating System
- Download and Install mongosh
- Run the connect command



If we are working locally we can install MongoDB Shell.

This step is not needed if we are using the Strigo VM as the mongosh has been installed already in the VM



# Connecting to MongoDB Atlas

If you have a mongosh already

- Select mongosh version from dropdown
- Run the connect command

✓ Setup connection security ✓ Choose a connection method Connect

I do not have the MongoDB Shell installed I have the MongoDB Shell installed

1 Select your mongo shell version

mongosh

(To check your shell version, run mongosh --version or mongo --version)

2 Run your connection string in your command line

Use this connection string in your application:

```
mongosh "mongodb+srv://pstt.ylpzx.mongodb.net/myFirstDatabase" --apiVersion 1 --username admin
```

Replace **myFirstDatabase** with the name of the database that connections will use by default. You will be prompted for the password for the Database User, **admin**. When entering your password, make sure all special characters are URL encoded.

Having trouble connecting? [View our troubleshooting documentation](#)

Select **I have the mongo shell installed**, make sure that **mongosh** is selected in the mongo shell version dropdown, and copy the command to connect to the cluster. We can close the dialog box after copying the command.

Now we can run the command shown in Atlas in our terminal.

- This specifies the DNS entry for our cluster (address knows about all the hosts)
- We specify a username and that we will be prompted for a password



# Connect to MongoDB via mongosh

Start a Terminal

**User:** admin

**Password:** qwerty123

The screenshot shows a terminal window titled "mongosh" within a "Virtual Machine" interface. The terminal displays the following text:

```
Welcome to ready-local
MongoDB Remote Training Host - Version 0.9
sampledata - Download sample data again if needed
Visit <hostname> to see someone else's terminal
Run "mongosh" or "mongosh --port 27017" or "mongosh "mongodb+srv://cluster0.rbxm4.mongodb.net/myFirstDatabase" --username admin"
Enter password: *****
Current Mongosh Log ID: 60efd94c858ce09479f37a66
Connecting to: mongodb+srv://cluster0.rbxm4.mongodb.net/myFirstDatabase
Using Mongosh: 1.0.0
For mongosh info see: https://docs.mongodb.com/mongosh-shell/
Atlas atlas-4s3wga-shard-0 [primary] myFirstDatabase>
```

Paste the command in your terminal and execute the same. Now we are connected to MongoDB.

# Essential commands





# Useful MongoDB Commands

**Show all databases** in the cluster

Check the **current database**

**Set db** to the **test** database

Check the current database again

**Show the collections** in the database

```
Atlas [primary] myFirstDatabase> show dbs
admin   377 kB
local   13.8 GB

Atlas [primary] myFirstDatabase> db
myFirstDatabase

Atlas [primary] myFirstDatabase> use test
switched to db test

Atlas [primary] test> db
test

Atlas [primary] test> show collections
Atlas [primary] test>
```

**show dbs** is a helper - it's equivalent to a small piece of javascript code to list the databases.

Some commands:

- **db** - this is a database object and is used at the beginning of most commands to run against a DB. Executing just db without any other methods, gives you the current database name.
- **show dbs** - shows available databases
- **use <database>** - selects database to use. Note that it does not need to already exist
- **show collections** - show collections available in the currently selected database. If there are no collections in a database, the command gives an empty response.



# Database Interaction

**Find all documents** in a collection

Create new document object **as variable**

**Insert a Document** in the collection

**Find Documents** based on a filter

```
Atlas [primary] test> db.employees.find({})  
  
Atlas [primary] test>  
var employee = { "name" : "Jon", "hungry" : true, "title" : "director" }  
{ name: 'Jon', hungry: true, title: 'director' }  
  
Atlas [primary] test> db.employees.insertOne(employee)  
{  
  acknowledged: true,  
  insertedId: ObjectId("60efe274ca75939ea79feeee")  
}  
  
Atlas [primary] test> db.employees.find({hungry:true})  
[  
  {  
    _id: ObjectId("60efe274ca75939ea79feeee"),  
    name: 'Jon',  
    hungry: true,  
    title: 'director'  
  }  
]  
  
Atlas [primary] test> show collections  
employees
```

Technically you are creating an object - and putting it in MongoDB as a Document, but Record is a generic term.

Some commands:

- **db** - this is a database object and is used at the beginning of most commands to run against a db
- **insertOne()** - this allows us to insert an object e.g.  
db.getCollection('test').insertOne({name:"bob"})
- **find()** - this allows us to query e.g. db.getCollection('test').find()

**Note:** The **employees** collection is automatically created when we execute the insert command.

Demo:

1. db.employees.find({})
2. var employee = { "name" : "Jon", "hungry" : true, "title" : "director" } { name: 'Jon', hungry: true , title: 'director' }
3. db.employees.insertOne(employee)
4. db.employees.find({hungry:true})
5. show collections



# A JavaScript REPL

**mongosh** - JavaScript and NodeJS REPL

```
Atlas [primary] test> for(let i=0; i<10; i++) {print(i);}
0
1
2
3
4
5
6
7
8
9

Atlas [primary] test>
```

We can run JavaScript code snippets in mongosh. It is a fully functional JavaScript and NodeJS REPL for interacting with MongoDB deployments.



# Help!

help

db.help

db.<collection>.help

A screenshot of a macOS-style terminal window titled "Atlas [primary] test". It shows three examples of the "help" command:

```
Atlas [primary] test> help
...
Atlas [primary] test> db.help
...
Atlas [primary] test> db.test.help
...
```

The window has a dark background with white text and standard OS X window controls at the top.

- help - Shell help
- db.help - Database class help
- db.col.help - Collection class help

# Useful tools





# VSCode Playground

Preview

Visual Studio Marketplace

Intelligent Autocomplete

The screenshot shows the VSCode interface with the MongoDB extension installed. The left sidebar displays 'CONNECTIONS' with one entry: 'pserv.ypxk.mongodb.net connected'. Below it is a 'PLAYGROUNDS' section stating 'No 'mongodb' playground files found in the workspace.' A 'Create New Playground' button is available. The main area is titled 'use["test"]; + Untitled-1' and contains the following MongoDB query:

```
use("test");
db.employees.find()
```

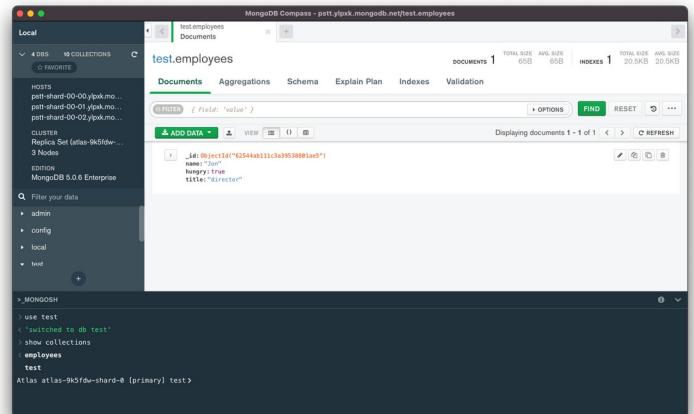
The bottom status bar indicates 'HELP AND FEEDBACK' with 0 errors and 0 warnings, and 'Initializing JS/TS language features'. It also shows 'Ln 4, Col 1' and 'Spaces: 2'.



# Compass

GUI tool that includes:

- Aggregation Builder
- Mongosh
- Visual explain Plans



You can download and install MongoDB Compass



# Quiz Time!





# #1. Which of the following statements describes best MongoDB Atlas?

A

Self-managed DB installed locally

B

Service to install a database on a server

C

Cloud provider used to deploy clusters

D

Managed database service on the cloud

E

Group of tools to work with a database

Answer in the next slide.



## #1. Which of the following statements describes best MongoDB Atlas?

A

Self-managed DB installed locally

B

Service to install a database on a server

C

Cloud provider used to deploy clusters

D

Managed database service on the cloud

E

Group of tools to work with a database



## #2. What three cloud providers we can use when creating a cluster in Atlas?

A

Google Cloud

B

IBM Cloud

C

Amazon Web Services

D

Adobe Creative  
Cloud

E

Microsoft Azure

Answer in the next slide.



#2. What three cloud providers we can use when creating a cluster in Atlas?

A

Google Cloud

B

IBM Cloud

C

Amazon Web Services

D

Adobe Creative  
Cloud

E

Microsoft Azure



### #3. Select the steps required to set up an Atlas database for the first time:

A

Register as an  
Atlas user

B

Add an IP  
address to the  
access list

C

Create a  
database user

D

Add a project to  
organization

E

Select the  
connection mode

Answer in the next slide.



### #3. Select the steps required to set up an Atlas database for the first time:

A

Register as an  
Atlas user

B

Add an IP  
address to the  
access list

C

Create a  
database user

D

Add a project to  
organization

E

Select the  
connection mode



## #4. Which of the following statements describes best the mongosh?

A

A GUI tool that works as a JS REPL

B

An app that interacts with the MongoDB Atlas GUI

C

A CLI to load data in MongoDB using Compass

D

A CLI that works as a JS REPL

E

A cloud application that manages Atlas

Answer in the next slide.



## #4. Which of the following statements describes best the mongosh?

A

A GUI tool that works as a JS REPL

B

An app that interacts with the MongoDB Atlas GUI

C

A CLI to load data in MongoDB using Compass

D

A CLI that works as a JS REPL

E

A cloud application that manages Atlas

# Recap

Atlas is MongoDB hosted as a service

Atlas is flexible and includes small, free clusters

Atlas always uses TLS and Authentication and a Firewall

A variety of tools are available to access the MongoDB database

`mongosh` is installed in the Training lab (Strigo VM)

# Appendix





# The MongoDB Web Shell

Hosted in a Browser

Auto-connects to shared cluster

Can connect to your cluster

<https://mws.mongodb.com>

```
db = new Mongo( <atlas URL> ).getDB("admin")
```

You **can access** a shell from a browser too. You can connect it to your own cluster by replacing the following value with your own

- ```
db = new
Mongo("mongodb+srv://admin:<password>@cluster0.bku7p.mongodb.net/myFirstDatabase?retryWrites=true&w=majority").getDB("admin")
```
- ```
Db = db.getSiblingDB("test")
```



# Legacy MongoDB Shell

mongo shell

Packaged with older servers

No syntax highlighting

```
df100.02 ~ mongo "mongodb+srv://pstt.ylpxk.mongodb.net/myFirstDatabase" --username admin --password
MongoDB shell version v4.4.8
Enter password:
connecting to: mongodb://pstt-shard-00-00.ylpxk.mongodb.net:27017,pstt-shard-00-01.ylpxk.mongodb.net:27017,pstt-shard-00-02.ylpxk.mongodb.net:27017?myFirstDatabase?authSource=admin&compressors=disabled&gssapiServiceName=mongodb&replicaSet=atlas-9k5fdw-shard-0&ssl=true
Implicit session: session { "_id" : UUID("11e42f1f-30bd-40b2-b3cc-1f9fb2612b4b") }
MongoDB server version: 5.0.6
WARNING: shell and server versions do not match
MongoDB Enterprise atlas-9k5fdw-shard-0:PRIMARY> show dbs
admin 0.000GB
local 1.467GB
test 0.000GB
MongoDB Enterprise atlas-9k5fdw-shard-0:PRIMARY> use test
switched to db test
MongoDB Enterprise atlas-9k5fdw-shard-0:PRIMARY> show collections
employees
MongoDB Enterprise atlas-9k5fdw-shard-0:PRIMARY> db.employees.find({})
{ "_id" : ObjectId("62544ab11c3a39538801ae5"), "name" : "Jon", "hungry" : true, "title" : "director" }
MongoDB Enterprise atlas-9k5fdw-shard-0:PRIMARY>
```

There is an older MongoDB shell (mongo) included in the core MongoDB installation that works in a similar way as the new mongosh.

If mongosh is not available, you can use the old MongoDB shell as well.

Major difference we note is that the old shell doesn't have syntax highlighting.