



TASK

MongoDB Overview

Visit our website

Introduction

WELCOME TO THE MONGODB OVERVIEW TASK!

You now know more or less what MongoDB is and how we will be using it. In this task, you will learn how to install MongoDB. You will also learn about using MongoDB to create a database as a service.

MONGODB AS A SERVICE

A few years ago, in order to run a database, you had to have a server (or servers) with all the necessary software installed and configured. You would also need someone who would act as the database administrator. In many organisations, this is still the case and this may be a justified expense and effort. Today though, there is an attractive alternative: database as a service.

Since cloud computing has become more popular, there are more cloud-based options for developers. You can host your web app on the cloud (HEROKU, Azure, etc) instead of on your own dedicated server. You can also use a database hosted by a cloud service provider (as in the case of MongoDB's Atlas), rather than having the hassle of setting up and maintaining your own database server. Below are some key benefits of this approach:

1. It is often cheaper than having your own database server because you only pay for what you use.
2. The cloud service provider deals with all the hassle of ensuring the configuration, backup, maintenance, security, etc of the database server.
3. It is quick and easy to start using a database with minimal configuration.

In this bootcamp, we will be using MongoDB's Database as a service solution: Atlas.

In this task, you are going to create your first database using MongoDB. Before you can do this though, you are first going to:

1. download and install MongoDB on your local machine so that you can use mongo, the administrative shell,
2. use Atlas to create and host a MongoDB on the cloud and
3. use Mongo to access and manipulate your database cluster on Atlas.

INSTALL MONGODB

To install MongoDB on your computer or server, download it from [MongoDB's download center](#). From this page, you will be presented with a few options of what you would like to install.

We will be installing the free Community Server for this course. Your other main option for download is the Enterprise Server which you can download on a free trial.

Click on the Community Server tab and download the latest release of MongoDB for either Windows, Linux or OSX.

The screenshot shows the MongoDB Download Center interface. At the top, there's a dark banner with the text "MongoDB Download Center". Below this, there are three tabs: "Cloud", "Server", and "Tools". The "Server" tab is selected. Under the "Server" tab, there are two options: "MongoDB Community Server" (labeled "FEATURE RICH. DEVELOPER READY.") and "MongoDB Enterprise Server" (labeled "ADVANCED FEATURES. PERFORMANCE GRADE."). Below these options, there are two dropdown menus: "Version" (set to "4.0.6 (current release)") and "OS" (set to "Windows 64-bit x64"). There is also a "Package" dropdown menu set to "MSI". A green "Download" button is prominently displayed. Below the button, a URL is shown: "https://fastdl.mongodb.org/win32/mongodb-win32-x86_64-2008plus-ssl-4.0.6-signed.msi". To the right of the download options, there is a list of links: "Release notes", "Changelog", "All version binaries", "Installation instructions", "Download source (tgz)", and "Download source (zip)".

Select the correct version and click on the Download button to begin the download.

Once the download is complete, follow the installation instructions for installing MongoDB as a service for your specific operating system:

- macOS:
<https://docs.mongodb.com/manual/tutorial/install-mongodb-on-os-x/>

- Windows:
<https://docs.mongodb.com/manual/tutorial/install-mongodb-on-windows/>
- Linux:
<https://docs.mongodb.com/manual/tutorial/install-mongodb-on-linux/>

The steps to setup and configure MongoDB differ slightly depending on the OS you are using. The installation instructions given above contain instructions for running MongoDB on your specific operating system. Below are the steps for running the Mongo shell on Windows (for Linux or macOS, see the documentation provided).

Once MongoDB has been installed, do the following:

- Step 1. Add the mongo executable to PATH, so the commands are accessible from outside the MongoDB bin folder.

To do this: Search for your MongoDB installation bin folder and copy the path (e.g.: C:\Program Files\MongoDB\bin). Right click My Computer > Properties > Advanced system settings > Environment Variables > System variables > Look for "Path" > Edit > New > Paste in the path to your MongoDB bin folder > Restart your terminal. More information about this step here.

- Step 3. Check that the Mongo shell has been correctly installed by typing "mongo --version" in your command line interface. Your output should be something similar to what is shown in the image below.

```
E:\>mongo --version
MongoDB shell version v3.6.2
git version: 489d177dbd0f0420a8ca04d39fd78d0a2c539420
OpenSSL version: OpenSSL 1.0.1u-fips 22 Sep 2016
allocator: tcmalloc
modules: enterprise
build environment:
  distmod: windows-64
  distarch: x86_64
  target_arch: x86_64
```

Your Mongo shell is now ready to be used to connect to a database server. In this course, we will be using MongoDB Atlas which will provide the platform and infrastructure we need for a database server on the cloud.



Take note:

Mongo is MongoDB's administrative shell. It is a C++ program that allows you to execute instructions on the database from a command line interface. Mongo allows you to use the MongoDB query language.


SETUP MONGODB ATLAS


In this bootcamp, we will be using MongoDB's Database as a service solution: Atlas. To get a quick (2 minute) overview of what Atlas is and why we are using it, watch this [short video](#).


To configure MongoDB Atlas, do the following:

- Go [here](#) and enter your information to get started with Atlas.
- You will then be taken to the 'Create New Cluster' page.
 - Under Cloud provider & Region select AWS and any free tier region.















Cloud Provider & Region AWS, N. Virginia (us-east-1) ▾


aws
FREE TIER AVAILABLE


Google Cloud Platform


Azure

Region availability changes depending on which cloud provider is chosen.
Regions marked with a ★ are recommended regions ⓘ

NORTH AMERICA	EUROPE	ASIA
<div> N. Virginia (us-east-1) ★ <small>FREE TIER AVAILABLE</small></div> <div> Ohio (us-east-2) ★</div> <div> N. California (us-west-1)</div> <div> Oregon (us-west-2) ★</div> <div> Montreal (ca-central-1)</div>	<div> Ireland (eu-west-1) ★</div> <div> London (eu-west-2) ★</div> <div> Frankfurt (eu-central-1) ★ <small>FREE TIER AVAILABLE</small></div> <div>SOUTH AMERICA <div> Sao Paulo (sa-east-1)</div></div> <div>AUSTRALIA <div> Sydney (ap-southeast-2) ★</div></div>	<div> Tokyo (ap-northeast-1)</div> <div> Seoul (ap-northeast-2)</div> <div> Singapore (ap-southeast-1)</div> <div> Mumbai (ap-south-1)</div>

- Under 'Cluster Tier' select the free M0 option.

Cluster Tier

M0 (Shared RAM, 512 MB Storage) ▼
Encrypted

Base hourly rate is for a MongoDB replica set with 3 data bearing servers.

Shared Clusters ?

✓ M0	Shared RAM	512 MB Storage	Shared VCPUs	FREE
M2	Shared RAM	2 GB Storage	Shared VCPUs	from \$0.012/hr
M5	Shared RAM	5 GB Storage	Shared VCPUs	from \$0.035/hr

Dedicated Development Clusters ?

M10	2 GB RAM	10 GB Storage	0.2 vCPUs	from \$0.08/hr
M20	4 GB RAM	20 GB Storage	0.4 vCPUs	from \$0.20/hr

Dedicated Production Clusters ?

M30	8 GB RAM	40 GB Storage	2 vCPUs	from \$0.54/hr
> M40	16 GB RAM	80 GB Storage	4 vCPUs	from \$1.04/hr
> M50	32 GB RAM	160 GB Storage	8 vCPUs	from \$2.00/hr
> M60	64 GB RAM	320 GB Storage	16 vCPUs	from \$3.95/hr
> M100	160 GB RAM	1000 GB Storage	40 vCPUs	from \$9.16/hr
> M200	256 GB RAM	1500 GB Storage	64 vCPUs	from \$14.59/hr
M400	488 GB RAM	3000 GB Storage	64 vCPUs	from \$22.40/hr

- You can rename your cluster under 'Cluster Name'.

Cluster Name

Hyperion-dev-1234 ▼

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

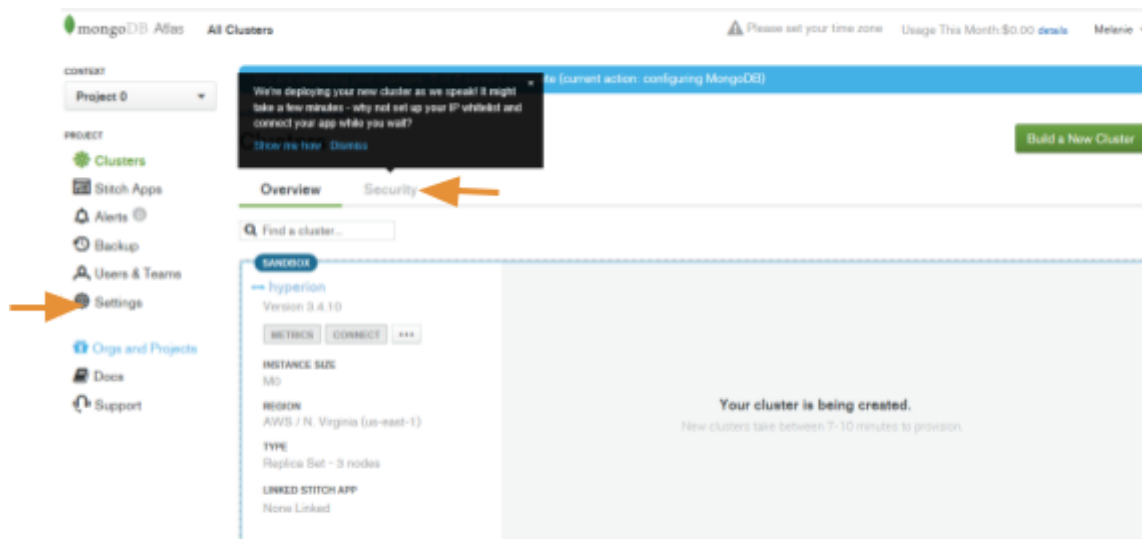
Hyperion-dev-1234

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

PREVIOUS: ADDITIONAL SETTINGS

- Click on the 'Create Cluster' button at the bottom of the page to create your cluster.

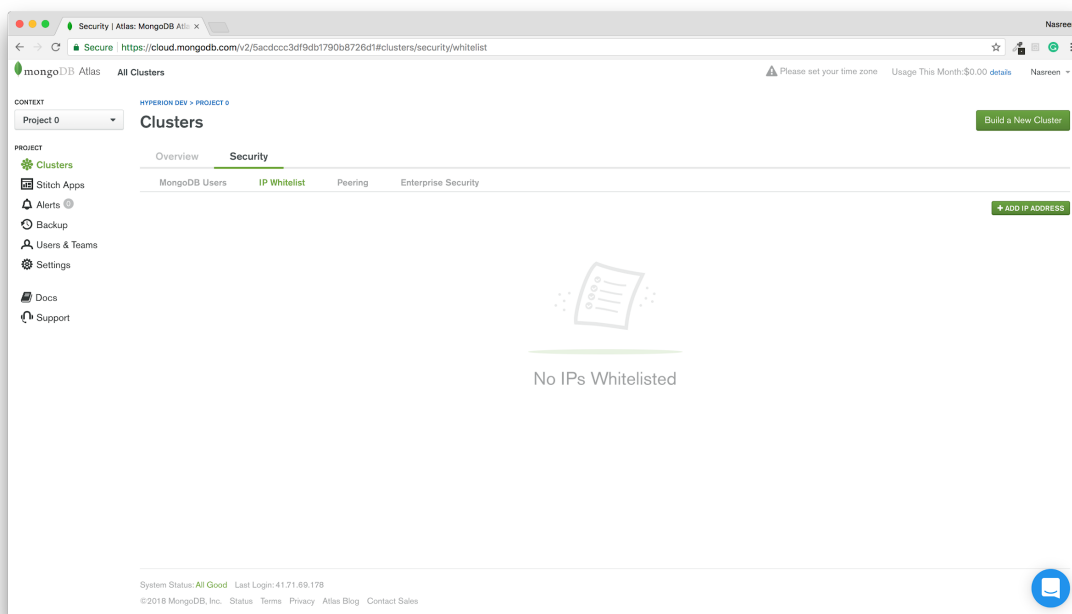
Once you have created your cluster, you will be taken to a page similar to the one shown below. You can change the name and other settings for your default project by selecting *settings* as shown in the image below.



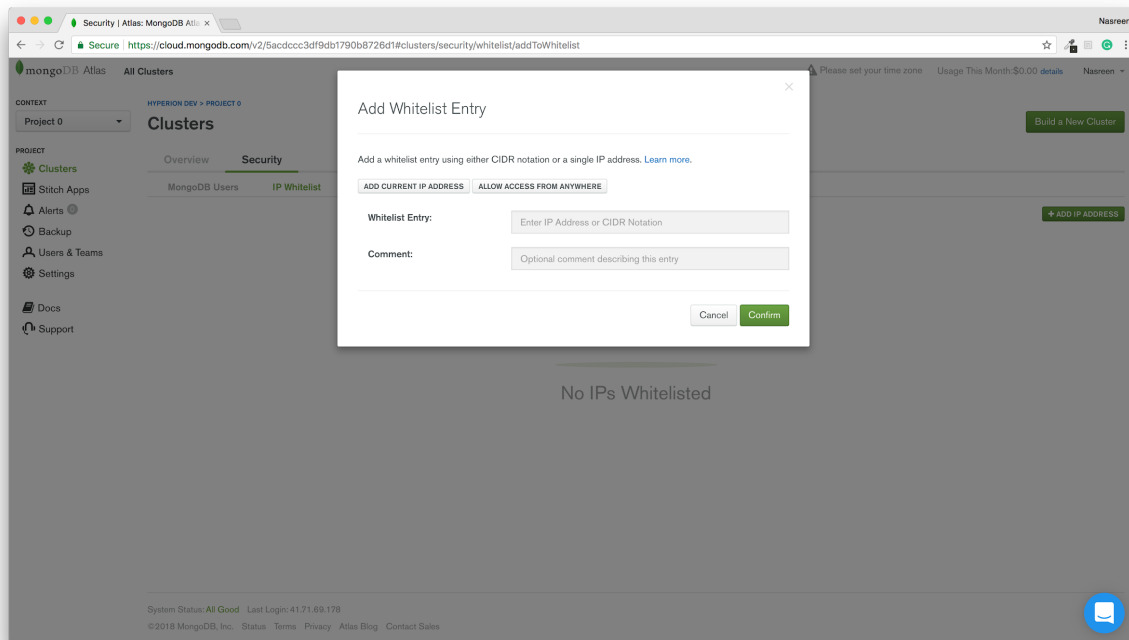
You have now successfully created a database cluster. You will now use the administrative shell, Mongo, to add a database to this cluster. Before you do this though, there are some security settings you need to tweak to make sure that you can access the Atlas cluster from your computer.

Security settings

One of the ways that MongoDB ensures security is by only allowing certain machines (IP addresses) access to your cluster. The IPs that are allowed to access your cluster are listed under the 'Security' tab in the 'IP Whitelist'. Click on the 'Security' tab and select 'IP Whitelist' from the menu, then click on the '+ Add IP Address' button.



In the 'Add Whitelist Entry' popup window, click on the 'Allow Access From Anywhere' button and then click 'Confirm'.



It is not good practice to allow all IP addresses to access your database for obvious security reasons, but for the purposes of the next few tasks, we are going to ask you to do this. The reason for this is that we are going to ask you to give an expert code reviewer access to your database and you don't have an expert code reviewer's IP address. In practice, however, it is advisable to have a limited IP whitelist.

If your machine is protected by a firewall, you also have to ensure that this doesn't block access to Atlas. Atlas servers run on **port 27017** on Amazon AWS. Check [here](#) to see if this port is blocked on your machine or not. If [this page](#) doesn't load, your firewall is probably blocking port 27017. If it is blocked, make sure to unblock it before you proceed. How this is done will depend on the firewall you are using. Google the appropriate instruction to unblock port 27017 for the firewall you are running.

From the Security tab, you can also add users and manage the rights of the users you allow to access your database.



Take note:

You should remember to configure this IP whitelist when you deploy your app. Remember that you will ultimately write code to access your database in your Express app (which you will soon learn to do). Therefore, your application server will be making requests to your database. If you

deploy your back-end app to the cloud (e.g. deploy to Heroku) you do not necessarily know what the IP address of your web server will be! This could obviously be a problem - if your web server IP address isn't added to the Atlas IP whitelist, your Express app won't be allowed to communicate with your database!

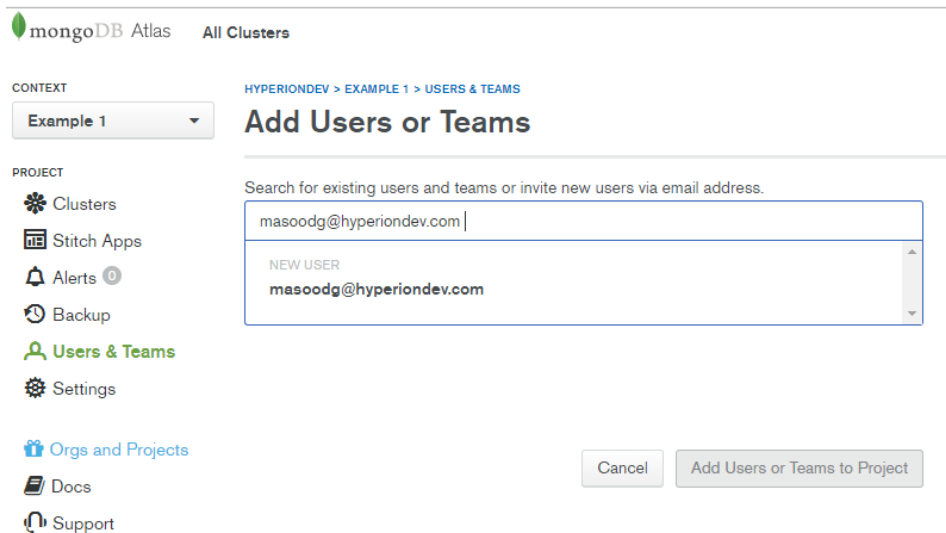
To address this problem you can create a Heroku Private Space. According to an [article by Heroku](#), "Private Spaces are dedicated environments for running dynos and certain types of add-ons within an isolated network. Access to apps in a Private Space can be controlled at the network level. Outbound requests from apps in a Private Space originate from a set of stable IP addresses, which allows you to securely communicate with IP white-listed services on-premise or on other networks."

We recommend that you read [this guide](#) by MongoDB to see how to integrate Atlas with Heroku. At the time of writing this task, this was not possible with a free Atlas account. In such a case you would have to allow access to your database from any IP address to allow your back-end deployed on Heroku to communicate with your database on MongoDB Atlas.

Although Heroku provides [add-ons](#) for accessing databases hosted by [mLab](#) (another Database as a Service for MongoDB), mLab was, at the time of writing, migrating to MongoDB Atlas. Thus the decision was made to base the content on MongoDB Atlas instead of mLab.

Manage users and teams

As the database administrator, you have to manage who is able to access your database and what they can do with your database. For your next MongoDB tasks, you are required to give an expert code reviewer access to your database. To do this, select "Users and teams" and then click on the "Add users and teams" button. You can then invite an expert code reviewer to be a user of your database by entering their email address as indicated in the image below:



ACCESS THE DATABASE ON THE CLOUD USING THE MONGO SHELL

You are now ready to access the database server you have configured on MongoDB Atlas using the Mongo shell on your local machine. Remember that Mongo is the administrative shell used to run instructions on your MongoDB server.

To connect to the database server using mongo you need a connection string that specifies everything needed for this connection. Atlas provides this connection string for you. Select “Connect” as shown below to find the connection string.

The following popup window will appear:

×

Connect to Hyperion-dev-1234

⚠ You must add a SCRAM user to this project to connect to Hyperion-dev-1234. To add a new user, navigate to the [Security tab](#).

1 Check the IP Whitelist

You will only be able to connect to your cluster from the following list of IP addresses

0.0.0.0/0 (includes your current IP address) ● Active

+ ADD ENTRY

ADD CURRENT IP ADDRESS

2 Choose a connection method:

Connect with the Mongo Shell

Mongo Shell with TLS/SSL support is required

>

Connect Your Application

Get a connection string and view driver connection examples

>

Connect with MongoDB Compass

Download Compass to explore, visualize, and manipulate your data

>

See methods to add data and diagnostics in the [Command Line Tools](#) shortcut from within your cluster.

Close

From the window above select 'Connect with the Mongo Shell'.

< BACK

×

Connect with the Mongo Shell

1 If you are not running MongoDB 3.6.3+ with TLS/SSL, click below to download the latest Mongo Shell

Windows

Mac OS X

Other Operating Systems ▾

2 Connect via the Mongo Shell

[View detailed instructions](#)

I am using shell 3.6 or later

I am using shell 3.4 or earlier


mongo "mongodb+srv://hyperion-dev-1234-5jme5.mongodb.net/test" --username <USERNAME>

COPY

You will be prompted for the password for the **USERNAME** user. Please ensure you are using a MongoDB 3.6.1+ shell with TLS/SSL.

[View your list of users](#) or [reset a password](#)

Close

 HyperionDev

Copyright © 2021 HyperionDev. All rights reserved.

The window above should then appear. Select 'I am using shell 3.6 or later', and copy and paste the connection string that appears there, into your command line interface. You should see output similar to that shown in the image below:

```
E:\>mongo "mongodb+srv://hyperion-f78fc.mongodb.net/test" --username hyperionDB
MongoDB shell version v3.6.2
Enter password:
connecting to: mongodb+srv://hyperion-f78fc.mongodb.net/test
2018-05-09T10:06:44.519+0200 I NETWORK [thread1] Starting new replica set monitor for hyperion-shard-00-00-f78fc.mongodb.net.:27017,hyperion-shard-00-01-f78fc.mongodb.net.:27017,hyperion-shard-00-02-f78fc.mongodb.net.:27017
2018-05-09T10:06:45.554+0200 I NETWORK [ReplicaSetMonitor-TaskExecutor-0] Successfully connected to hyperion-shard-00-00-f78fc.mongodb.net.:27017 (1 connections now open to hyperion-shard-00-00-f78fc.mongodb.net.:27017 with a 5 second timeout)
2018-05-09T10:06:45.568+0200 I NETWORK [thread1] Successfully connected to hyperion-shard-00-02-f78fc.mongodb.net.:27017 with a 5 second timeout)
2018-05-09T10:06:45.823+0200 I NETWORK [thread1] changing hosts to hyperion-shard-0/hyperion-shard-00-01-f78fc.mongodb.net:27017,hyperion-shard-00-02-f78fc.mongodb.net:27017,hyperion-shard-00-00-f78fc.mongodb.net.:27017,hyperion-shard-00-01-f78fc.mongodb.net.:27017,hyperion-shard-00-02-f78fc.mongodb.net.:27017
2018-05-09T10:06:46.838+0200 I NETWORK [ReplicaSetMonitor-TaskExecutor-0] Successfully connected to hyperion-shard-00-01-f78fc.mongodb.net.:27017 (1 connections now open to hyperion-shard-00-02-f78fc.mongodb.net.:27017 with a 5 second timeout)
2018-05-09T10:06:46.848+0200 I NETWORK [thread1] Successfully connected to hyperion-shard-00-00-f78fc.mongodb.net.:27017 (1 connections now open to hyperion-shard-00-00-f78fc.mongodb.net.:27017 with a 5 second timeout)
2018-05-09T10:06:48.132+0200 I NETWORK [ReplicaSetMonitor-TaskExecutor-0] Successfully connected to hyperion-shard-00-01-f78fc.mongodb.net.:27017 (1 connections now open to hyperion-shard-00-01-f78fc.mongodb.net.:27017 with a 5 second timeout)
MongoDB server version: 3.4.14
WARNING: shell and server versions do not match
MongoDB Enterprise hyperion-shard-0:PRIMARY>
```

You have connected to your MongoDB server hosted by Atlas. You are now able to use the Mongo shell to create and modify databases on your server.



Extra resource

Watch [this short video](#) to see how easily you can connect to your Atlas database with the mongo shell.

CREATE A DATABASE

Once you can access your database server (run by Atlas), you can issue instructions using mongo to change your database. We are going to create a database.

To do this type the following using the mongo shell: **use test** where *test* is the name of the database. If the database does not already exist, this instruction will create it.

```
MongoDB Enterprise hyperion-shard-0:PRIMARY> use test
switched to db test
MongoDB Enterprise hyperion-shard-0:PRIMARY>
```

MONGODB COMPASS

You may have noticed that when you installed MongoDB, MongoDB Compass was also installed. Compass allows you to interface with your database. You should be able to connect to your database using Compass too. Give it a try.

QUIT MONGO

To quit mongo, type `quit()` into the mongo shell.

Compulsory Task 1

Follow these steps:

- Install MongoDB. See appropriate detailed installation instructions on MongoDB's download centre for details.
- Create a cluster on MongoDB Atlas.
- Add an expert code reviewer as a user to your cluster on Atlas.
- Ensure that your firewall isn't blocking access to MongoDB.
- Connect to your cluster using the mongo shell.
- Make a database called "test"
- Create a document called "myMongoDB" in which you include the following:
 - A screenshot that shows that you have added an expert code reviewer as a MongoDB user to your Atlas cluster.
 - A screenshot of your command line interface that shows how you have used the Mongo shell to connect to your MongoDB Atlas cluster.
 - A screenshot of your command line interface that shows that you have successfully created a database called "test" on your MongoDB Atlas cluster.



Rate us
Share your thoughts

HyperionDev strives to provide internationally-excellent course content that helps you achieve your learning outcomes.

Think that the content of this task, or this course as a whole, can be improved or think we've done a good job?

[Click here](#) to share your thoughts anonymously.

