MongoDB Atlas Setup Guide

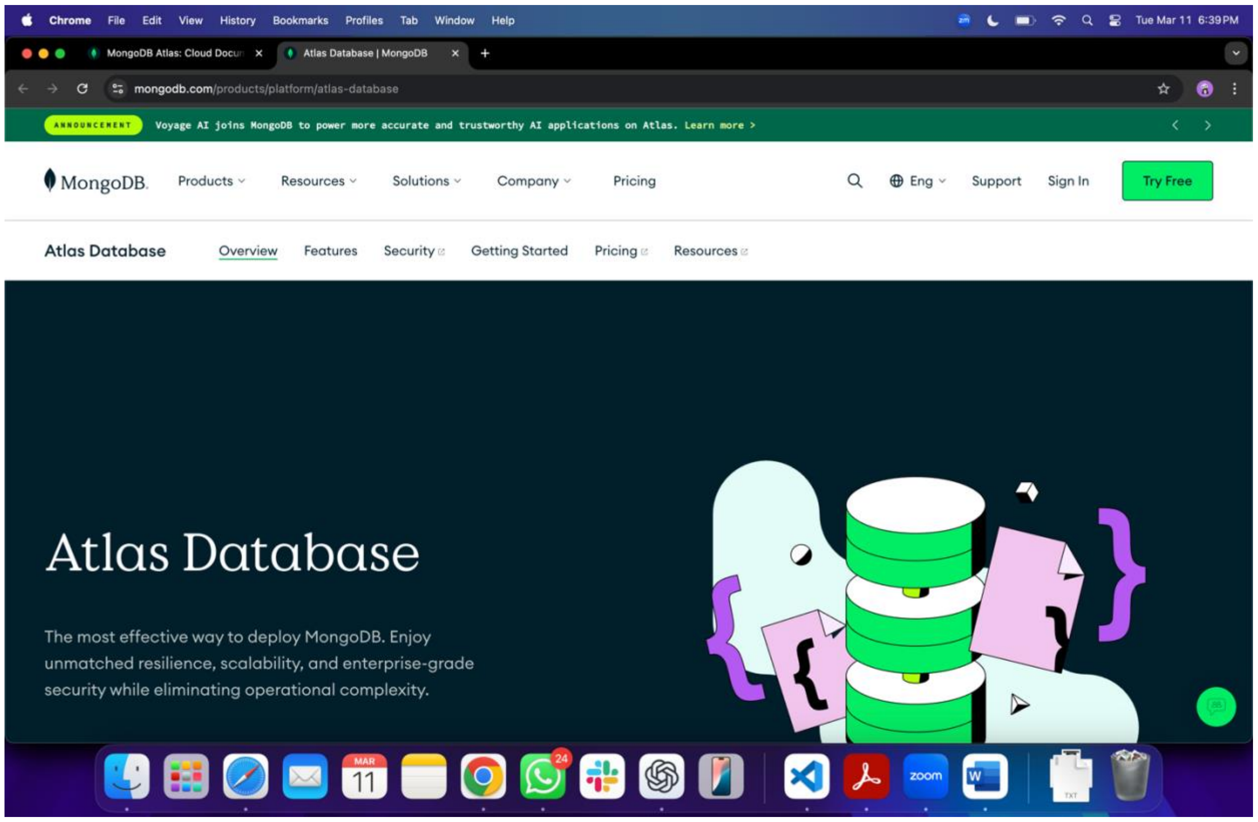
Step 0: Installing Python on Your Local Machine

If you haven't installed Python on your computer, please follow the guide below:

1. Installing Python on macOS using Homebrew (Recommended to avoid interfering with the system's default Python).
    - Install Homebrew on macOS.
    - Install Python using Homebrew.
2. Installing Python on Windows.
3. Using a Code Editor (e.g., VS Code)
    - Installation guide for macOS
    - Installation guide for Windows
4. Creating a Virtual Environment
5. Installing Python Libraries using pip

Step 1. Sign Up for MongoDB Atlas

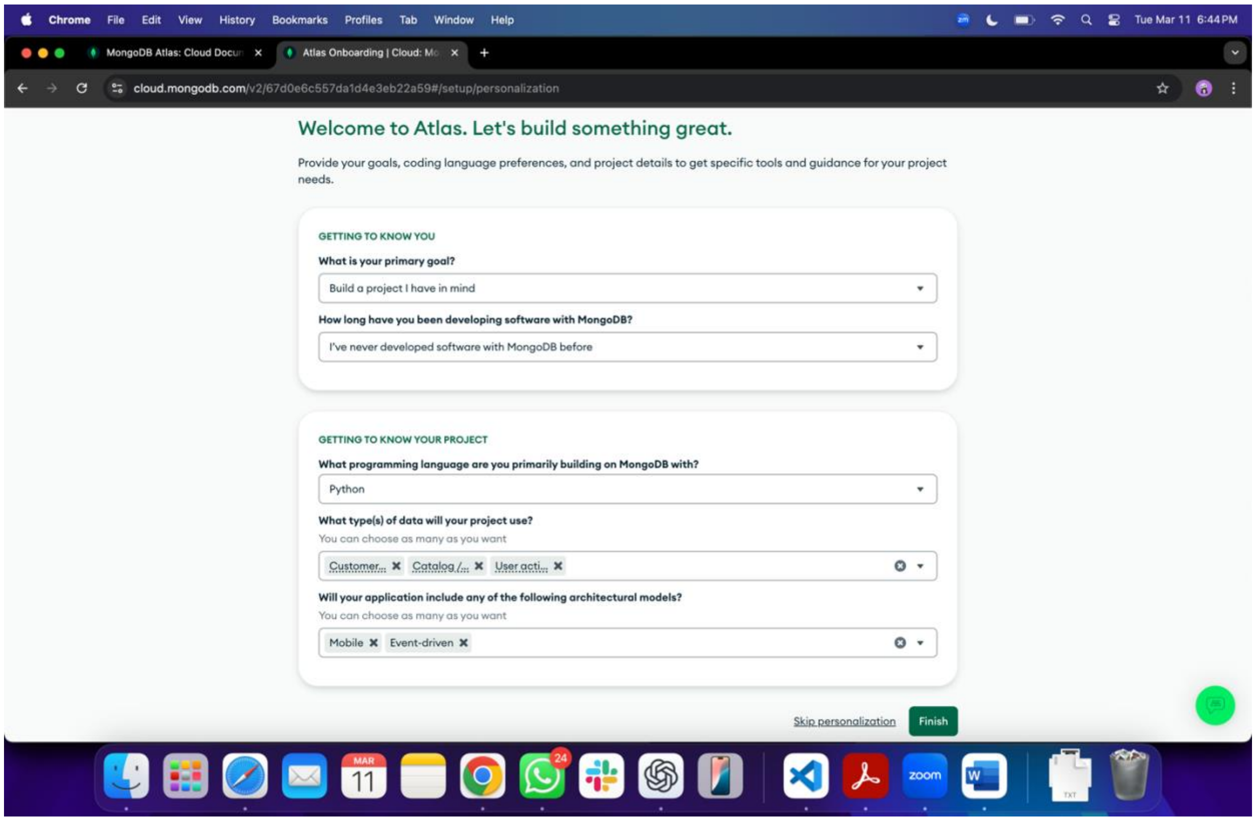
1. Go to MongoDB Atlas.


Chrome File Edit View History Bookmarks Profiles Tab Window Help
Tue Mar 11 6:39PM
MongoDB Atlas: Cloud Docu
Atlas Database | MongoDB
mongodb.com/products/platform/atlas-database
ANNOUNCEMENT Voyage AI joins MongoDB to power more accurate and trustworthy AI applications on Atlas. Learn more >
MongoDB.
Products
Resources
Solutions
Company
Pricing
Eng
Support
Sign In
Try Free
Atlas Database
Overview
Features
Security
Getting Started
Pricing
Resources

Atlas Database

The most effective way to deploy MongoDB. Enjoy unmatched resilience, scalability, and enterprise-grade security while eliminating operational complexity.

2. Click Get Started and Sign Up or Log In.
3. (Optional): Get Free MongoDB Credits with GitHub Student Developer Pack
    - Visit [MongoDB for Students](#).
    - Sign in with your GitHub account.
4. Accept the terms and conditions.
5. Complete the optional personalization steps.


Welcome to Atlas. Let's build something great.

Provide your goals, coding language preferences, and project details to get specific tools and guidance for your project needs.

GETTING TO KNOW YOU

What is your primary goal?
Build a project I have in mind

How long have you been developing software with MongoDB?
I've never developed software with MongoDB before

GETTING TO KNOW YOUR PROJECT

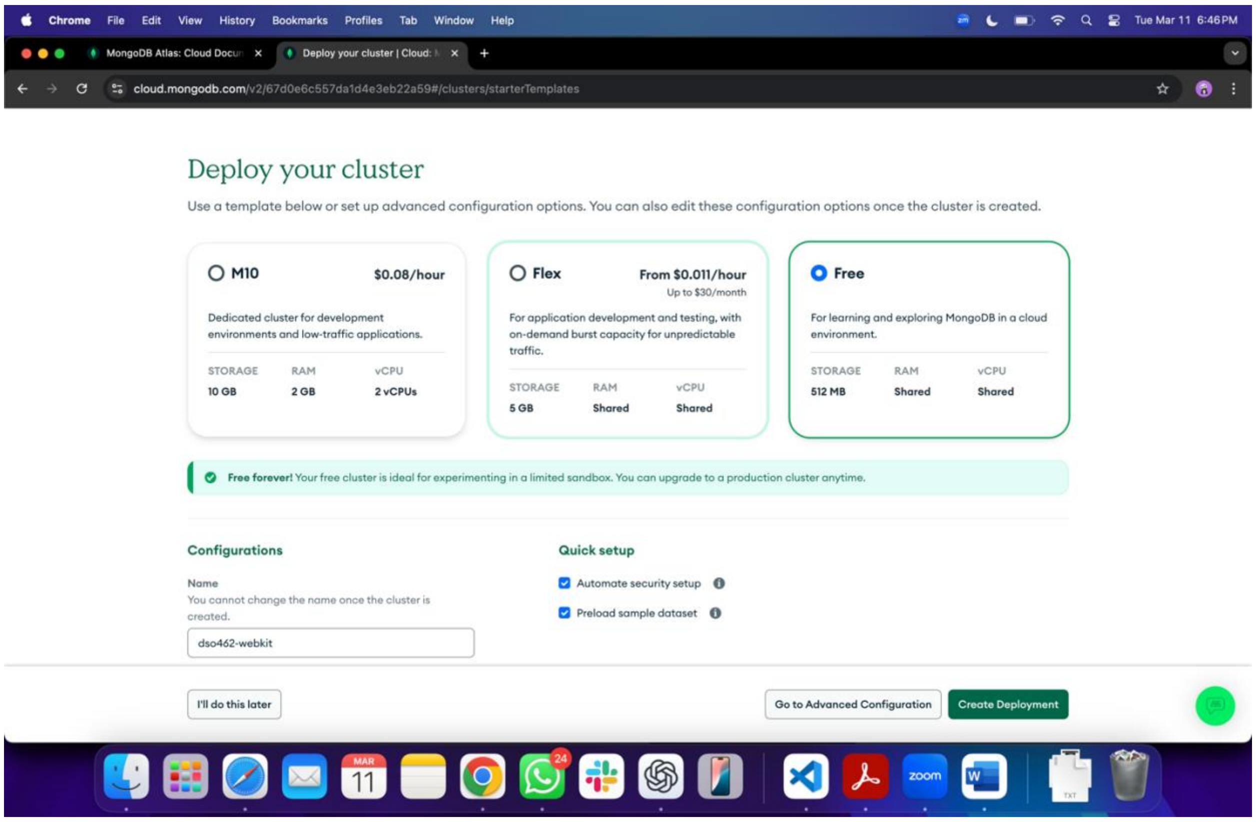
What programming language are you primarily building on MongoDB with?
Python

What type(s) of data will your project use?
You can choose as many as you want
Customer... Catalog /... User acti...

Will your application include any of the following architectural models?
You can choose as many as you want
Mobile Event-driven

Step 2. Create a MongoDB Cluster

1. After logging in, you will create a cluster.
2. Choose the Free Tier Cluster (*shared cluster, free for development*).
3. Cluster Configuration:
    - Cluster Name: Choose any descriptive name (*e.g., "DSO462-WebKit"*).
    - Cloud Provider: Select from AWS, Google Cloud, or Azure. *I chose* Google Cloud (personal preference).
    - Region: Choose a location close to your target users. *I chose* America (for lower latency if the majority of users are in the U.S.).
4. Click Create Deployment (*this may take a few minutes*).



cloud.mongodb.com/v2/67d0e6c557da1d4e3eb22a59#/clusters/starterTemplates

Deploy your cluster

Use a template below or set up advanced configuration options. You can also edit these configuration options once the cluster is created.

○ M10	$0.08/hour		
Dedicated cluster for development environments and low-traffic applications.			
STORAGE	RAM	vCPU	
10 GB	2 GB	2 vCPUs	

○ Flex	From $0.011/hour Up to $30/month		
For application development and testing, with on-demand burst capacity for unpredictable traffic.			
STORAGE	RAM	vCPU	
5 GB	Shared	Shared	

● Free			
For learning and exploring MongoDB in a cloud environment.			
STORAGE	RAM	vCPU	
512 MB	Shared	Shared	

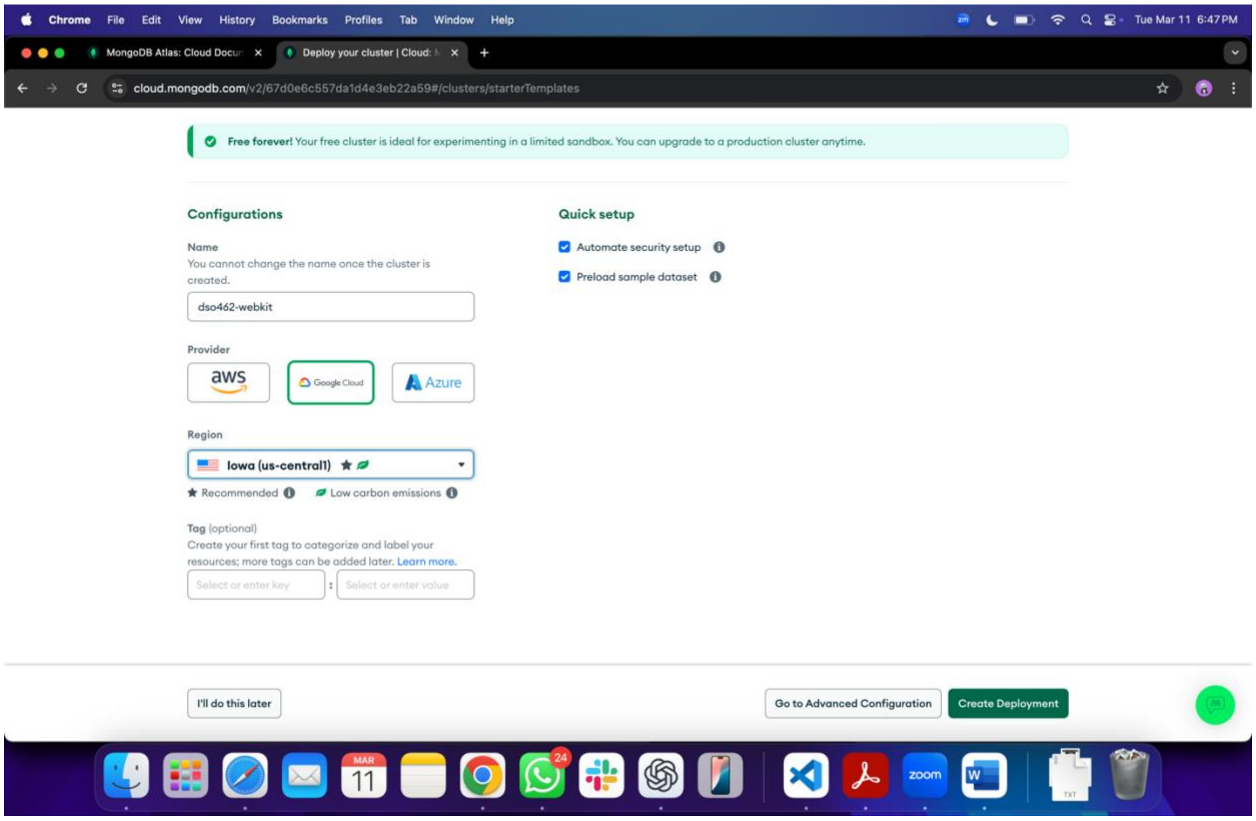
Free forever! Your free cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

Configurations

Name
You cannot change the name once the cluster is created.
dso462-webkit

Quick setup

☑ Automate security setup
☑ Preload sample dataset



cloud.mongodb.com/v2/67d0e6c557da1d4e3eb22a59#/clusters/starterTemplates

Free forever! Your free cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

Configurations

Name
You cannot change the name once the cluster is created.
dso462-webkit

Provider
aws [✓ Google Cloud] Azure

Region
Iowa (us-central1)

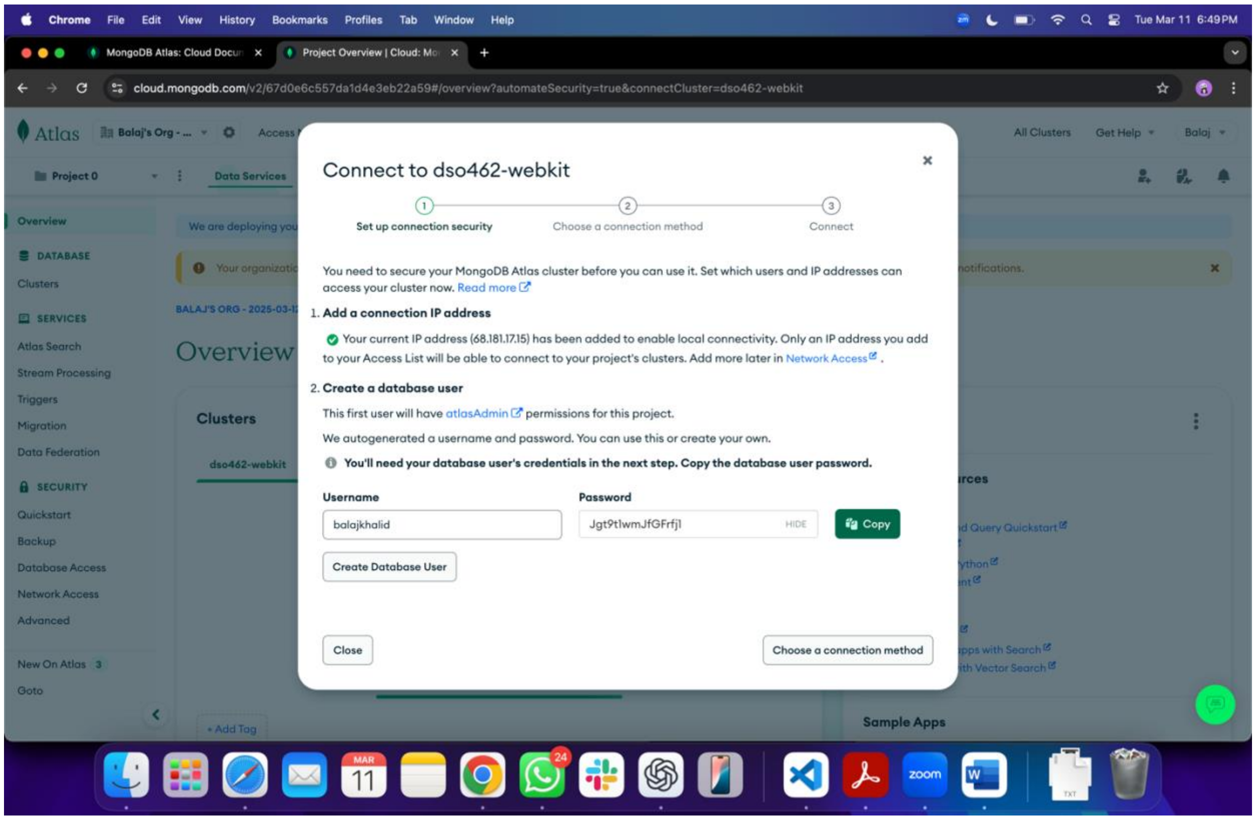
★ Recommended Low carbon emissions

Tag (optional)
Create your first tag to categorize and label your resources; more tags can be added later. Learn more.

Quick setup

☑ Automate security setup
☑ Preload sample dataset

Step 3. Connecting to Database



Connect to dso462-webkit

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

1. **Add a connection IP address**

   Your current IP address (68.181.17.15) has been added to enable local connectivity. Only an IP address you add to your Access List will be able to connect to your project's clusters. Add more later in Network Access .

2. **Create a database user**

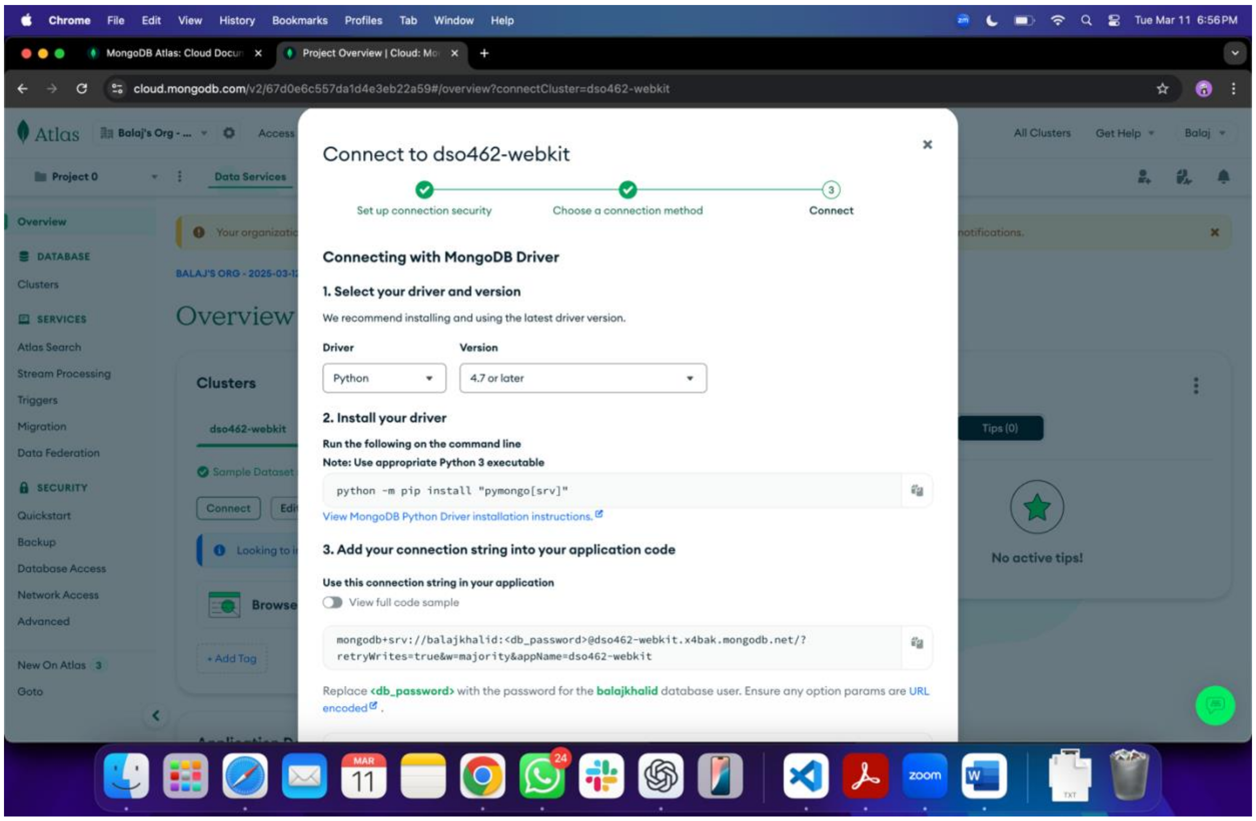
   This first user will have atlasAdmin permissions for this project.

   We autogenerated a username and password. You can use this or create your own.

   You'll need your database user's credentials in the next step. Copy the database user password.

   Username: balajkhalid
   Password: Jgt9tlwmJfGFrfj1

1. Once your cluster is created you will be guided to connect to your app.
2. By default, your IP will be whitelisted (allowed access).
3. Click Create Database Users and create an initial user (you can create more later).
4. Next click on Chose a connection method.
5. Choose Connection Method → Drivers.
6. Select Python as the language.
7. Choose Version 4.7 or later.
8. Install the pymongo library to interact with MongoDB. Use command ```python -m pip install "pymongo[srv]"```
9. In your project folder, create a .env file.
10. Add connection string: ```MONGO_URI=mongodb+srv://<username>:<password>@yourcluster.mongodb.net/?retryWrites=true&w=majority&appName=<appName>```



Connect to dso462-webkit

Set up connection security — Choose a connection method — Connect

Connecting with MongoDB Driver

1. Select your driver and version

We recommend installing and using the latest driver version.

Driver	Version
Python	4.7 or later

2. Install your driver

Run the following on the command line
Note: Use appropriate Python 3 executable

```
python -m pip install "pymongo[srv]"
```

View MongoDB Python Driver installation instructions.

3. Add your connection string into your application code

Use this connection string in your application

View full code sample

```
mongodb+srv://balajkhalid:<db_password>@dso462-webkit.x4bak.mongodb.net/?
retryWrites=true&w=majority&appName=dso462-webkit
```

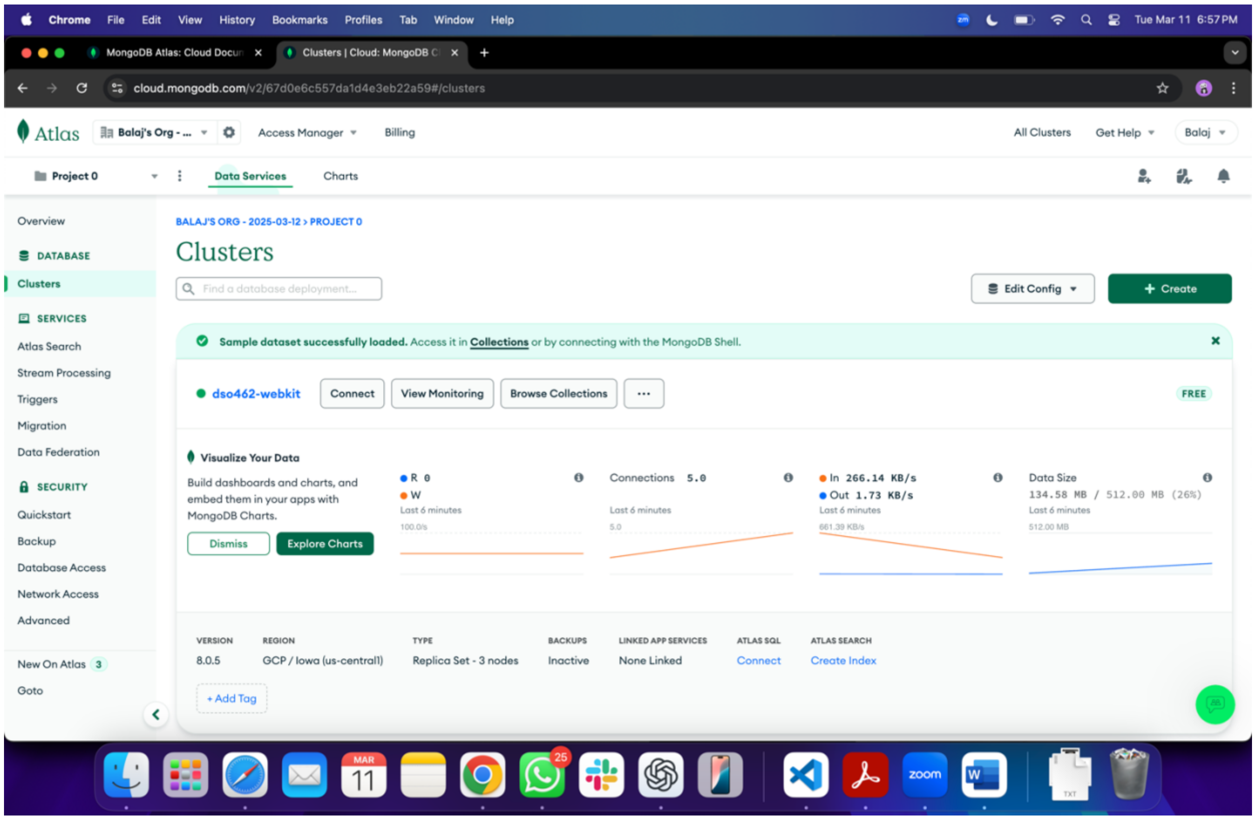
Replace **<db_password>** with the password for the **balajkhalid** database user. Ensure any option params are URL encoded .

Step 4. Test MongoDB Connection

1. Use the file test.py bundled with initial webkit code on github.
2. Run the script using command ```python3 test.py```
3. Expected output: “Pinged your deployment. You successfully connected to MongoDB!”

Now you have a functional MongoDB Database.

Step 5. Creating and Managing Databases & Collections



cloud.mongodb.com/v2/67d0e6c557da1d4e3eb22a59#/clusters

BALAJ'S ORG - 2025-03-12 > PROJECT 0

Clusters

Sample dataset successfully loaded. Access it in Collections or by connecting with the MongoDB Shell.

dso462-webkit

Visualize Your Data

Build dashboards and charts, and embed them in your apps with MongoDB Charts.

R 0
W
Last 6 minutes

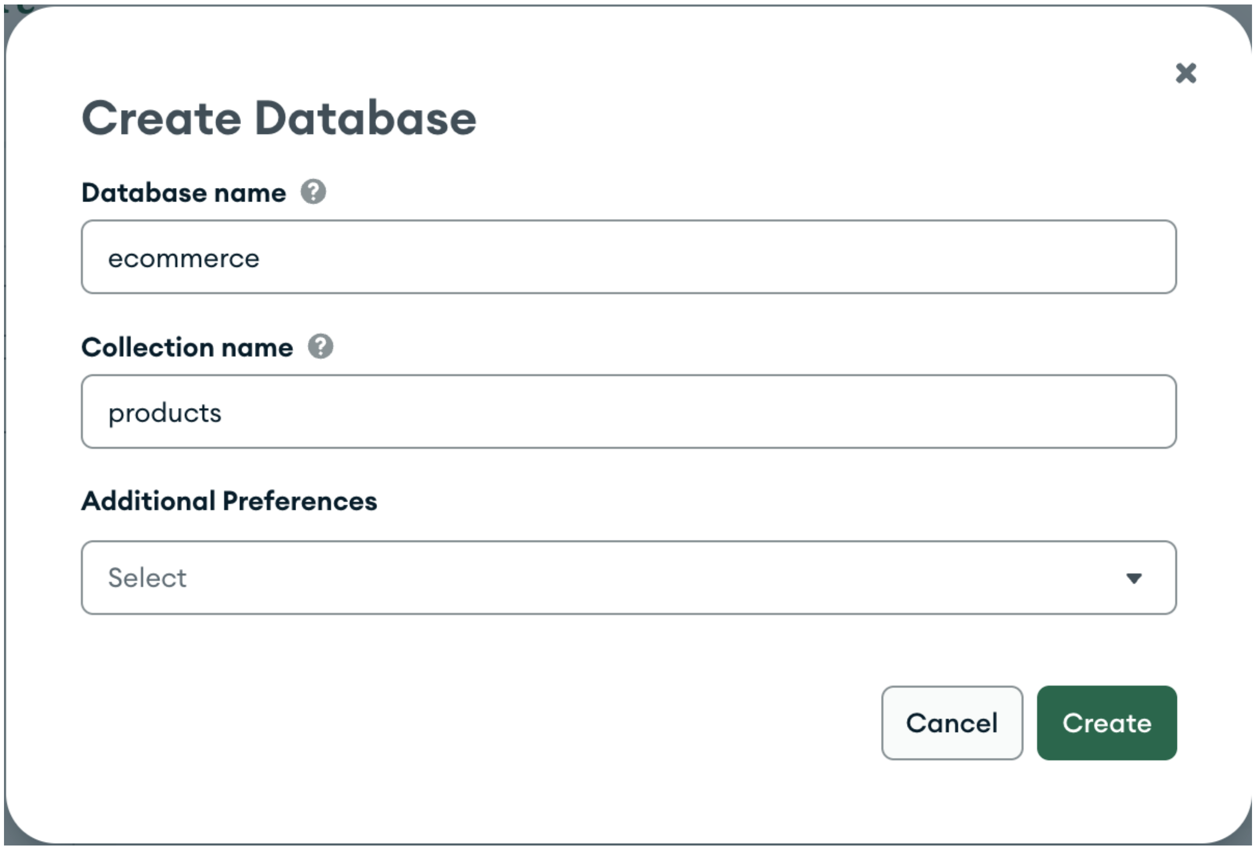
Connections 5.0
Last 6 minutes

In 266.14 KB/s
Out 1.73 KB/s
Last 6 minutes

Data Size
134.58 MB / 512.00 MB (26%)
Last 6 minutes

VERSION	REGION	TYPE	BACKUPS	LINKED APP SERVICES	ATLAS SQL	ATLAS SEARCH
8.0.5	GCP / Iowa (us-central1)	Replica Set - 3 nodes	Inactive	None Linked	Connect	Create Index

1. Go to Clusters → Browse Collections.
2. Click Create Database.
3. Enter:
    a. Database Name: ecommerce (sample database name)
    b. Collection Name: products (sample collection)
4. Click Create.



Understanding Database Structure

- Database → Similar to a schema in SQL.
- Collection → Equivalent to a table in SQL.
- Document → Similar to a row (stored in JSON format).

Step 6. Querying Data (Basic CRUD Operations)

6.1. Insert Data

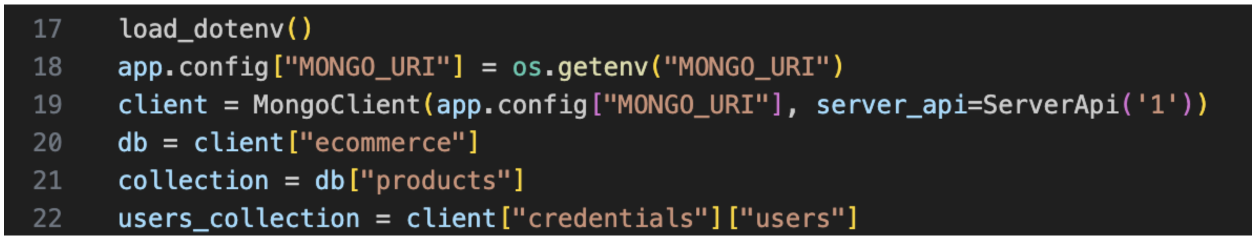
1. Refer to file insert_data.py bundled with initial webkit code on github.
2. Run the script using command ```python3 insert_data.py```

6.2. Retrieve Data

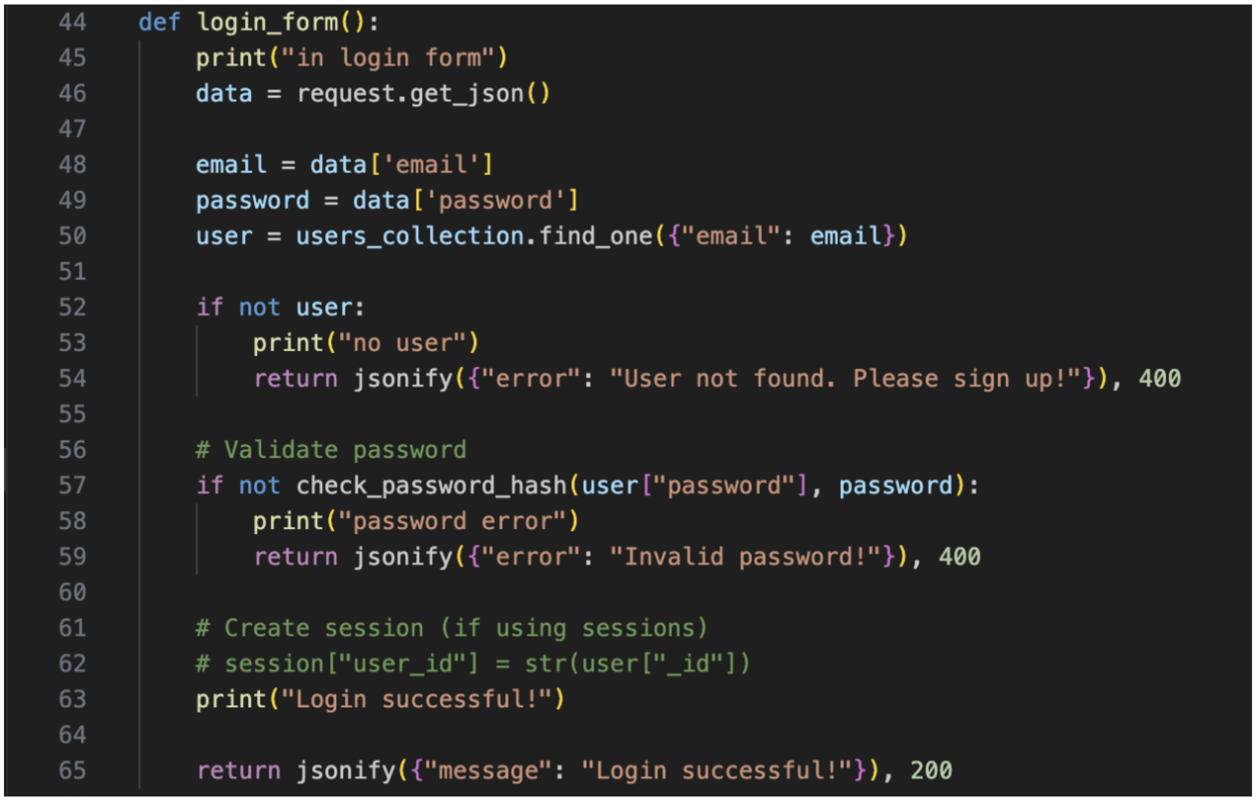
The following code refers to app.py bundled with initial webkit code on github.

1. In line 17, we are loading the .env file.
2. In line 18, we are assigning the URI to variable ```app.config["MONGO_URI"]
3. In line 19, we are forming a connection with MongoDB databse.
4. In line 22, we are using the client object created in line 19 to access database followed by table.

5. In line 50, we are using user_collections object defined in line 22, to query the database.



```
17  load_dotenv()
18  app.config["MONGO_URI"] = os.getenv("MONGO_URI")
19  client = MongoClient(app.config["MONGO_URI"], server_api=ServerApi('1'))
20  db = client["ecommerce"]
21  collection = db["products"]
22  users_collection = client["credentials"]["users"]
```

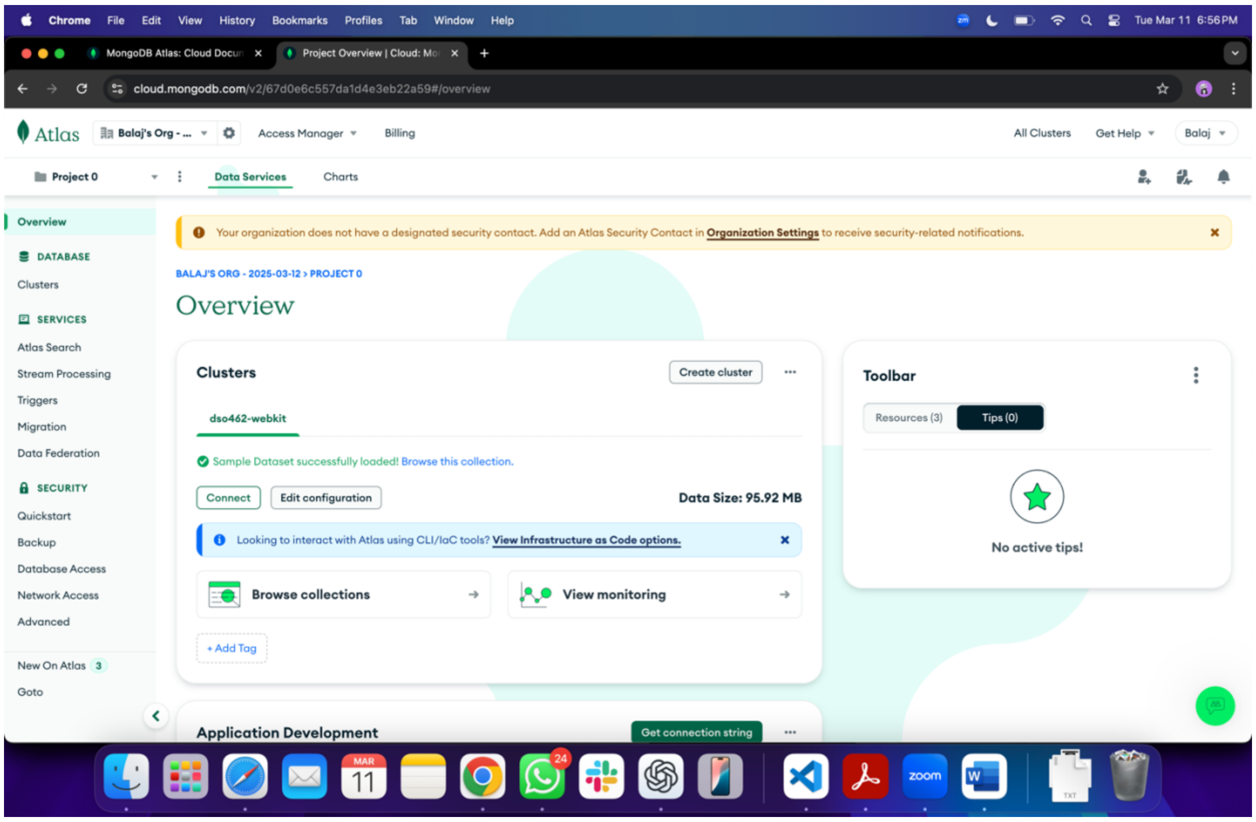


```
44  def login_form():
45      print("in login form")
46      data = request.get_json()
47  
48      email = data['email']
49      password = data['password']
50      user = users_collection.find_one({"email": email})
51  
52      if not user:
53          print("no user")
54          return jsonify({"error": "User not found. Please sign up!"}), 400
55  
56      # Validate password
57      if not check_password_hash(user["password"], password):
58          print("password error")
59          return jsonify({"error": "Invalid password!"}), 400
60  
61      # Create session (if using sessions)
62      # session["user_id"] = str(user["_id"])
63      print("Login successful!")
64  
65      return jsonify({"message": "Login successful!"}), 200
```

Additional Setup

Create Database Users (Authentication & Security)

- MongoDB requires database users for authentication.
- These users are different from your MongoDB Atlas login.



cloud.mongodb.com/v2/67d0e6c557da1d4e3eb22a59#/overview

Your organization does not have a designated security contact. Add an Atlas Security Contact in Organization Settings to receive security-related notifications.

Sample Dataset successfully loaded! Browse this collection.

Data Size: 95.92 MB

Looking to interact with Atlas using CLI/IaC tools? View Infrastructure as Code options.

No active tips!

Create the Additional Database User

1. Go to Database Access (left sidebar).
2. Click Create Database User.
3. Set a Username & Password.
4. Choose Authentication Method:
   a. (Default, Recommended) → Works for most applications.
5. Set Access Level:
   a. Read and Write → If the user needs to add or modify data.
   b. Read-Only → For limited access.
6. Click Create User.

Whitelist IP Addresses (Security & Team Collaboration)

To restrict access to trusted sources, MongoDB Atlas uses IP whitelisting.

Whitelist Teammates' IPs

1. In Network Access, click Add IP Address.
2. Enter the IP addresses of teammates who need access.
3. Set Access Level:
   - "Allow only for a specific period" (temporary access).
   - "Allow indefinitely" (permanent access for teammates).

Whitelist All IPs (Not Recommended for Production)

If you're working on an early-stage project and want to allow all IPs:

1. Go to Network Access.
2. Add 0.0.0.0/0 → This allows access from any IP.
3. ⚠️ Warning: This is insecure and should only be used for quick testing.

Managing User Access & Team Collaboration

Invite Team Members

1. Go to Projects → Access Manager.
2. Click Invite Members.
3. Assign roles (*e.g., Read-Only, Admin*).

Set Up Different Database Users for Different Parts of Your App

1. Go to Projects → Database Access.
2. Click Add New Database User.
3. Assign different permissions based on team needs.

Deploying the Application

When deploying your app:

- Ensure environment variables are set in .env.
- Whitelist the server's IP (add to Network Access).
- Use read-only users for production databases.