Finish Populating MongoDB

Estimated duration: 60 mins

Welcome to the **Finish Populating MongoDB** lab. In this lab, you will apply the skills you have already acquired in managing MongoDB databases and handling data within a Node.js environment. Using a GitHub repository, you will clone, set up, and populate a MongoDB database with gifts data, leveraging your existing knowledge in database operations and Node.js scripting. This lab is an opportunity to test your learning and showcase your ability to handle data management tasks in a full-stack development context.

Objectives

- Clone a GitHub repository template containing the necessary files and data
- Initialize and configure MongoDB in the lab environment
- Employ command-line skills to navigate and execute database operations
- Apply best practices in managing secure database connections using environment variables
- Import data into MongoDB

Prerequisite

Important security information

Welcome to the Cloud IDE for the lab. This lab environment is where all of your development will take place. It has all the tools you will need, including a MongoDB database.

Your lab environment is temporary. You will not see the same lab environment twice. So, you must push all changes to your own GitHub repository to recreate it in a new lab environment, whenever required.

Also, note that this environment is shared and, therefore, not secure. You should not store personal information, usernames, passwords, or access tokens in this environment for any purpose.

Your task

1. If you still need to generate a **GitHub Personal Access Token**, you should do so now. You can follow the instructions in this lab to generate a personal token. You will need it to push code back to your repository. Your repo should have `repo` and `write` permissions and set to expire in `60` days. When Git prompts you for a password in the Cloud IDE environment, use your Personal Access Token instead.

2. You can recreate this environment by **Initializing the Development Environment** each time.

Step 1: Clone your repository

You can skip this step if you have already cloned the repository as part of the previous lab. If the repository is not available in this lab environment, take the following steps to clone the repository.

Your task

1. Access your GitHub account and locate the repository you created from the template provided. It should be named `fullstack-capstone-project`.
2. Use Git commands to clone the repository to your local environment.
3. Confirm that the repository contents, including the gifts data, are correctly downloaded.

Hint

▼ Click here for a hint.

Utilize the `git clone <repository-url>` command.

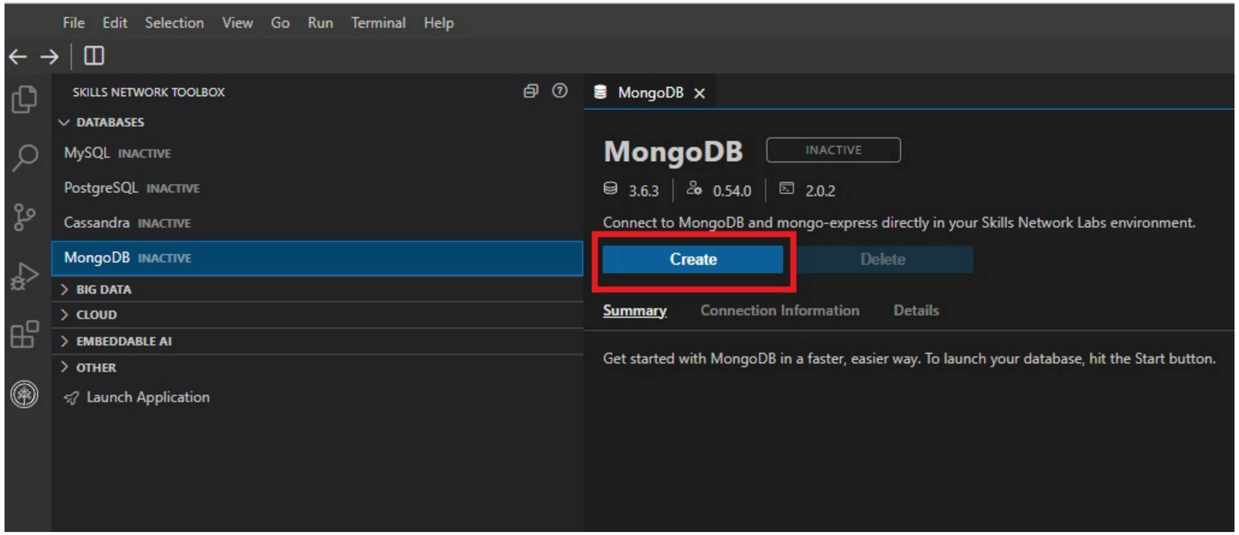
Step 2: Initialize MongoDB

Next, you will initialize an instance of MongoDB.

IMPORTANT! You must take a screenshot of the terminal output as directed in the last task of this step, Initialize MongoDB. Save it in an easy to fine location. You will need this screenshot for your final project.

Tasks

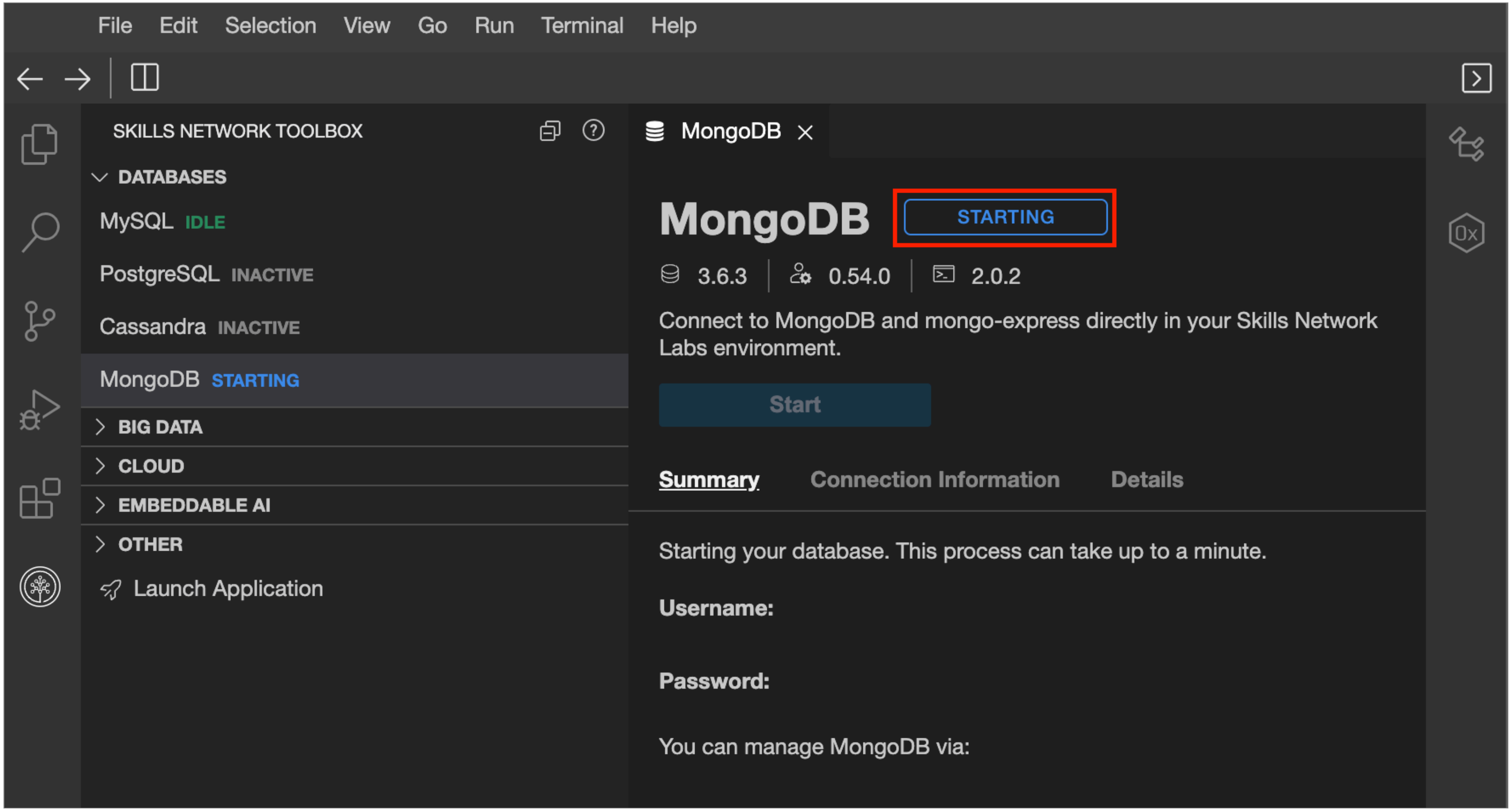
1. Open the `Skills Network Toolbox`
2. Open the `Databases` category and select `MongoDB`. Click `Create` to initialize the database service.



Connect to MongoDB and mongo-express directly in your Skills Network Labs environment.

Get started with MongoDB in a faster, easier way. To launch your database, hit the Start button.

3. You should see the database starting. This process can take a couple of minutes.



Connect to MongoDB and mongo-express directly in your Skills Network Labs environment.

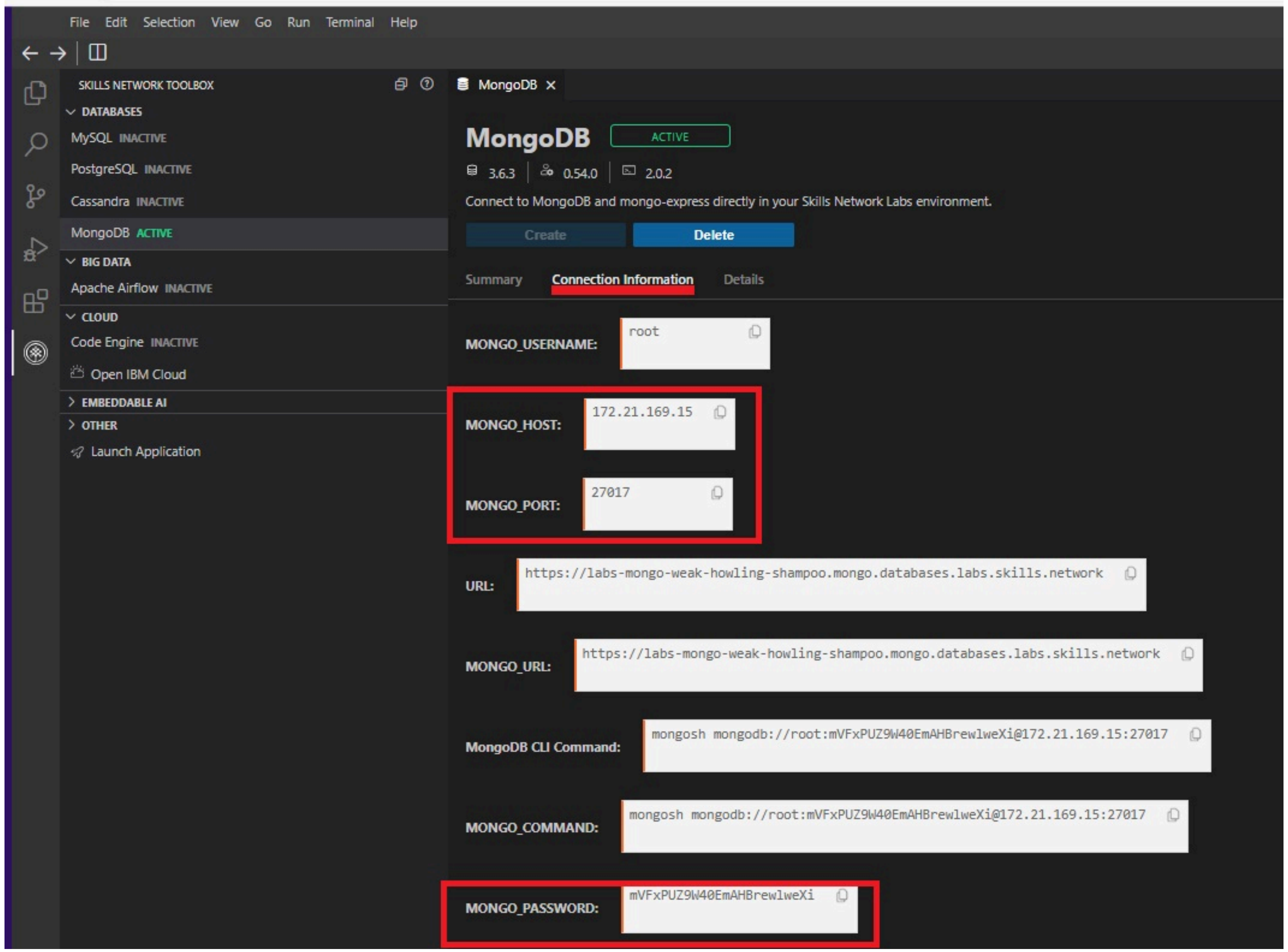
Starting your database. This process can take up to a minute.

Username:

Password:

You can manage MongoDB via:

4. Copy the **MONGO_HOST**,**MONGO_PORT** ,**MONGO_PASSWORD** which is generated under connection information tab.You will need this information in the next step of this lab. Since the lab environment is temporary, the password and Host Id might be different the next time you start MongoDB database service.



MongoDB ACTIVE

3.6.3 | 0.54.0 | 2.0.2

Connect to MongoDB and mongo-express directly in your Skills Network Labs environment.

MONGO_USERNAME: root

MONGO_HOST: 172.21.169.15

MONGO_PORT: 27017

URL: https://labs-mongo-weak-howling-shampoo.mongo.databases.labs.skills.network

MONGO_URL: https://labs-mongo-weak-howling-shampoo.mongo.databases.labs.skills.network

MongoDB CLI Command: mongosh mongodb://root:mVFxPUZ9W40EmAHBrewlweXi@172.21.169.15:27017

MONGO_COMMAND: mongosh mongodb://root:mVFxPUZ9W40EmAHBrewlweXi@172.21.169.15:27017

MONGO_PASSWORD: mVFxPUZ9W40EmAHBrewlweXi

Step 3: Import gifts data

The next step is to import the gifts data provided to you in the `giftlink-backend/util/import-mongo/gifts.json` file. MongoDB is a document-based NoSQL database where data is organized into collections and documents, providing a flexible schema. Each document can store varied data structures, making MongoDB particularly well-suited for handling large volumes of unstructured or semi-structured data. In this capstone project, you will work with `gifts` that have the following attributes.

- id
- name
- category
- condition
- posted_by
- zipcode
- date_added
- age_days
- age_years
- description
- image

A sample gift document is given below.

```
{
```

```
_id: ObjectId('65c166b57361698a5cdd82cc'),
    id: '875',
    name: 'Lamp',
    category: 'Kitchen',
    condition: 'New',
    posted_by: '26872',
    zipcode: '94805',
    date_added: 1667520000,
    age_days: 400,
    age_years: 1.1,
    description: "A charming lamp that's been lighting up my reading nook. It casts a soft, inviting glow, perfect for those who love to curl up with a good book. It's seen some of my favorite chapters and is ready to illuminate yours!",
    image: '/images/lamp.jpeg'
}
```

Open the `gifts.json` file and look at the sample data provided to you. You don't need to make any changes to this file.

Open **gifts.json** in IDE

Tasks

1. Open a terminal with `Terminal` -> `New Terminal` if one isn't open already.
2. Change to the `import-mongo` directory.

   ```
   cd /home/project/fullstack-capstone-project/giftlink-backend/util/import-mongo
   ```

3. Duplicate the `.env.sample` file, naming the new file `.env`. This file contains the username and password for MongoDB. Fill out the password and Host Id from the previous step where you started the database service. The username should remain `root`.

   ```
   MONGO_URL=mongodb://root:password_of_mongodb_instance@localhost:27017
   ```

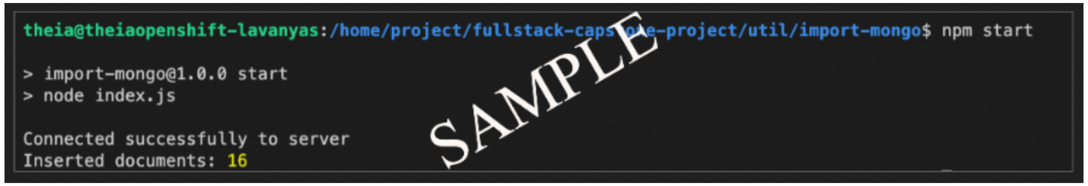
4. Install the required npm packages by executing the `npm install` command in the terminal.

   ```
   npm install
   ```

5. Run the Node program by executing the `npm start` command in the terminal.

```
npm start
```

6. If all goes well, you should see the following message in the terminal:



```
theia@theiaopenshift-lavanyas:/home/project/fullstack-capstone-project/util/import-mongo$ npm start

> import-mongo@1.0.0 start
> node index.js

Connected successfully to server
Inserted documents: 16
```

The line, `Inserted documents: 16` confirms you inserted 16 gifts into the database.

7. Take a screenshot of the terminal output and save it in an easy to fine location. You will need this screenshot for your final project.

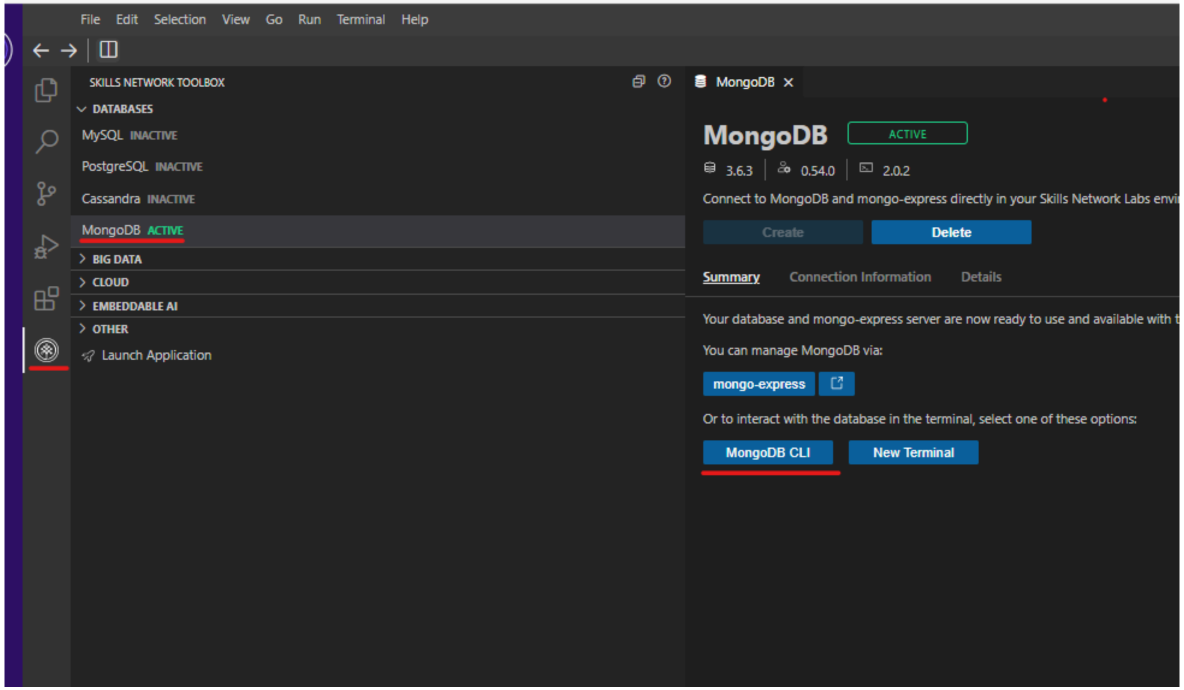
Step 4: Validate gifts data import

Having successfully imported your sample gifts data into MongoDB, next you will validate your database contains the correct data. In this task, you'll use *mongosh*, the official MongoDB shell, an interactive JavaScript interface to MongoDB.

Mongosh will query and update data as well as perform administrative operations. With its user-friendly interface and powerful capabilities, mongosh is an essential tool for developers working with MongoDB databases.

Tasks

1. Open the mongosh terminal by using the MongoDB tab. You can open it again if you closed it.



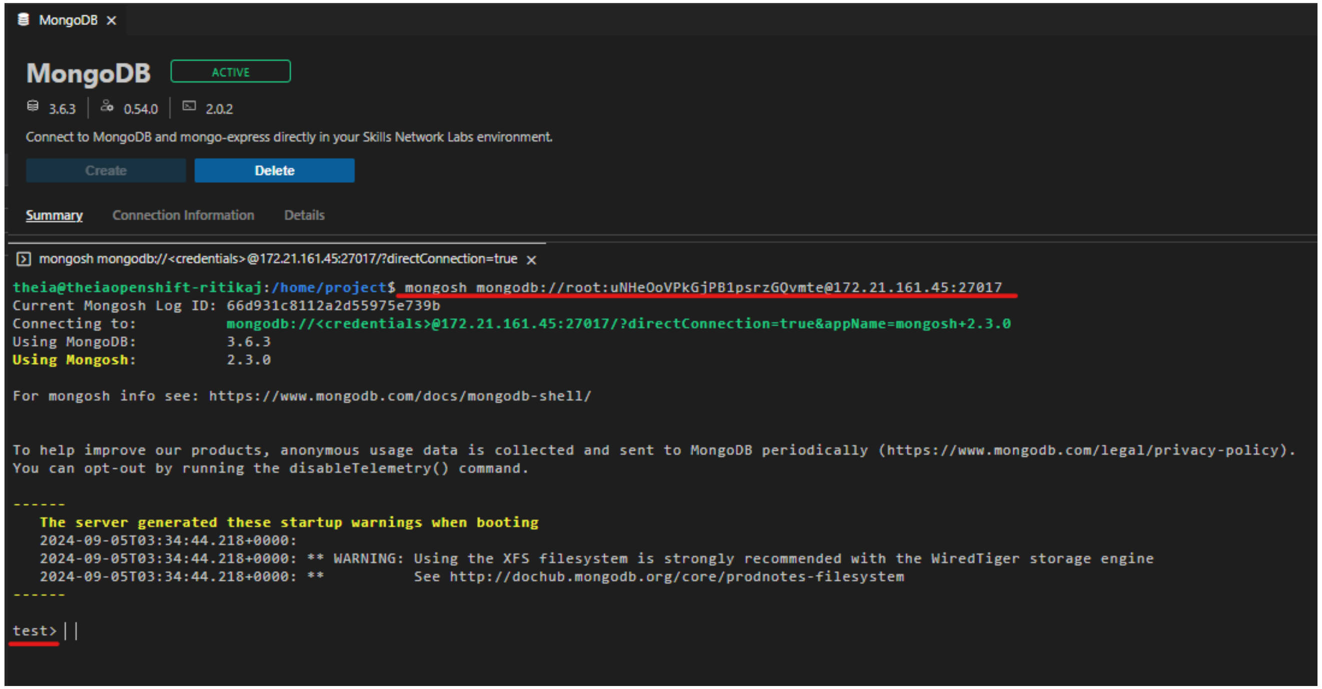
Connect to MongoDB and mongo-express directly in your Skills Network Labs envi

Your database and mongo-express server are now ready to use and available with t

You can manage MongoDB via:

Or to interact with the database in the terminal, select one of these options:

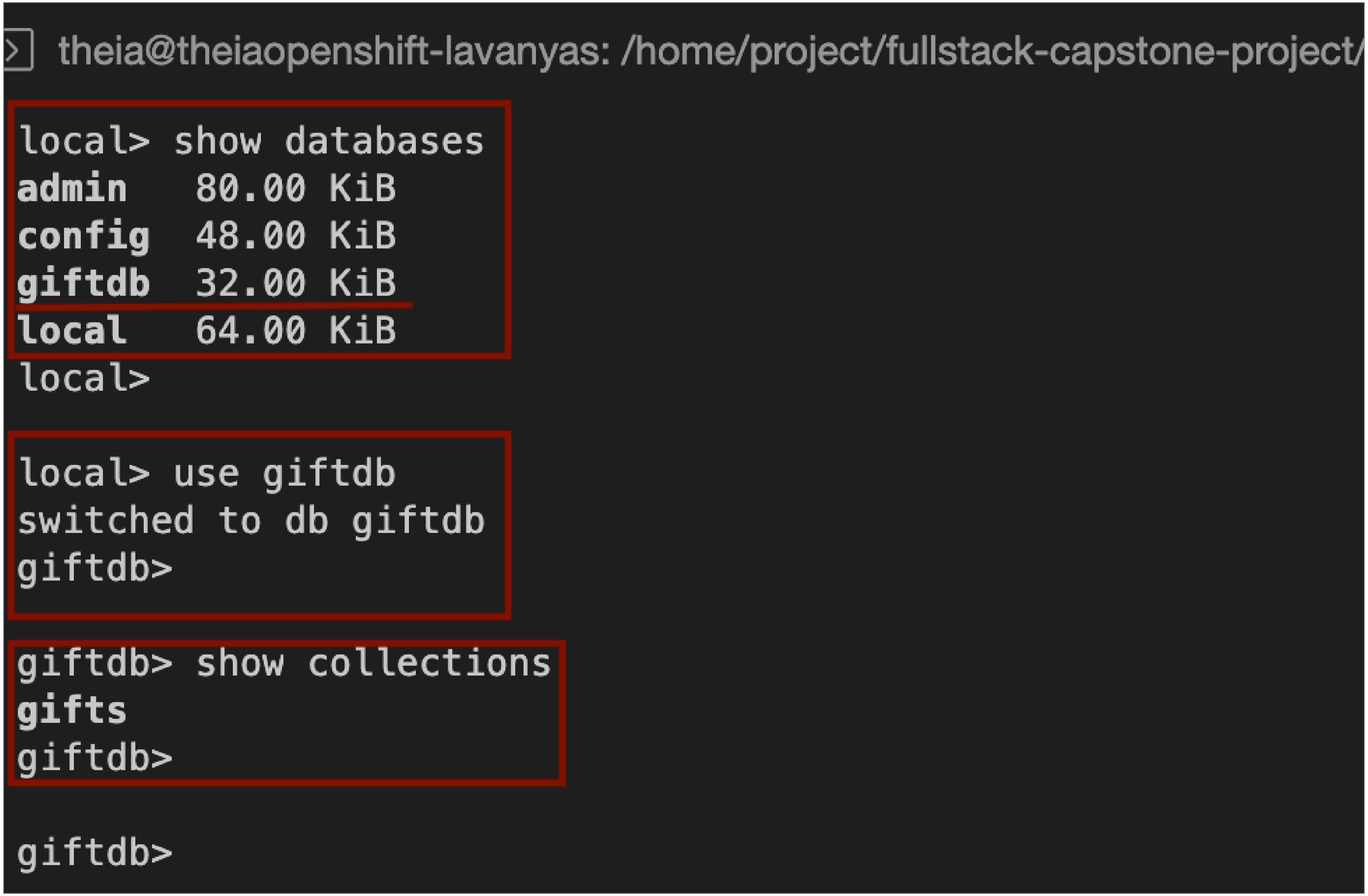
2. This will automatically open a terminal and start `Mongosh` for you.



Connect to MongoDB and mongo-express directly in your Skills Network Labs environment.

```
theia@theiaopenshift-ritikaj:/home/project$ mongosh mongodb://root:uNHeOoVPkGjPB1psrzGQvmte@172.21.161.45:27017
Current Mongosh Log ID: 66d931c8112a2d55975e739b
Connecting to:          mongodb://<credentials>@172.21.161.45:27017/?directConnection=true&appName=mongosh+2.3.0
Using MongoDB:          3.6.3
Using Mongosh:          2.3.0

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


To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.

------
   The server generated these startup warnings when booting
   2024-09-05T03:34:44.218+0000:
   2024-09-05T03:34:44.218+0000: ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine
   2024-09-05T03:34:44.218+0000: **          See http://dochub.mongodb.org/core/prodnotes-filesystem
------

test> ||
```

Try the `show databases` command to see all the databases in the system. You should see several databases, including `giftdb`. Use the `use giftdb` command to switch to that database, and then execute `show collections` to display the gifts collection created by the Node program.



```
theia@theiaopenshift-lavanyas: /home/project/fullstack-capstone-project/

local> show databases
admin    80.00 KiB
config   48.00 KiB
giftdb   32.00 KiB
local    64.00 KiB
local>

local> use giftdb
switched to db giftdb
giftdb>

giftdb> show collections
gifts
giftdb>

giftdb>
```

3. Your turn now. Count the number of documents in the `gifts` collection. The result should be 16. Take a screenshot of the terminal showing the command and the result and save it as `fsc-mongosh-count.png`.

4. Execute a second command to get the two documents from the `gifts` collection. Save a screenshot of the terminal showing the command and the result as `fsc-mongosh-find.png`.

5. Find the gift with the id `429`. Save the screenshot of the teminal showing the command and the result as `fsc-mongosh-429.png`. The output should look as follows:

```
[
    {
    _id: ObjectId('65c166b57361698a5cdd82da'),
    id: '429',
    name: 'Side Table',
    category: 'Living Room',
    condition: 'Older',
    posted_by: '23789',
    zipcode: '94102',
    date_added: 1664928900,
    age_days: 1100,
    age_years: 3.01,
    description: "This side table has been a convenient spot for lamps and books. It's still functional but has some scratches. Selling it as part of a living room makeover.",
    image: '/images/side-table.jpeg'
  }
]
```

Hints

Count the number of gifts in the collection:

▼ Click here for the answer

Use the `countDocuments()` method.

Retrieve the first two gifts from the collection:

▼ Click here for the answer

Use the `find()` method with the `limit()` method.

Find a specific gift in the collection:

▼ Click here for the answer

Use the `find()` method and pass in the query `{ "id": "429" }` as a parameter.

Conclusion

Congratulations on completing the `Populate MongoDB` lab. By completing this lab, you effectively utilized your existing knowledge in database management and Node.js to perform a full-stack development task. This exercise reinforced your ability to integrate back-end technologies and handle data operations, solidifying your skills in building dynamic and efficient web applications.

Next steps

You should spend some more time trying other commands you remember from the course on MongoDB. This environment is a good place to practice your skills.