

Cassandra CRUD Operations



Estimated time needed: 15 minutes

Objectives

After completing this lab, you will be able to:

- Insert data into a table with an INSERT command
- Read data from a table by querying from it
- Update and delete data from the table based on specific criteria

About This SN Labs Cloud IDE

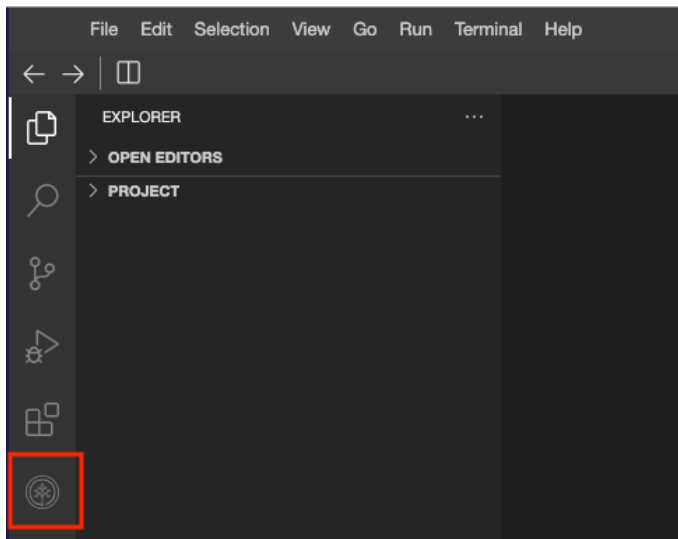
This Skills Network Labs Cloud IDE provides a hands-on environment for course and project related labs. It utilizes Theia, an open-source IDE (Integrated Development Environment) platform, that can be run on desktop or on the cloud. To complete this lab, we will be using the Cloud IDE based on Theia and Cassandra running in a Docker container.

Important Notice about this lab environment

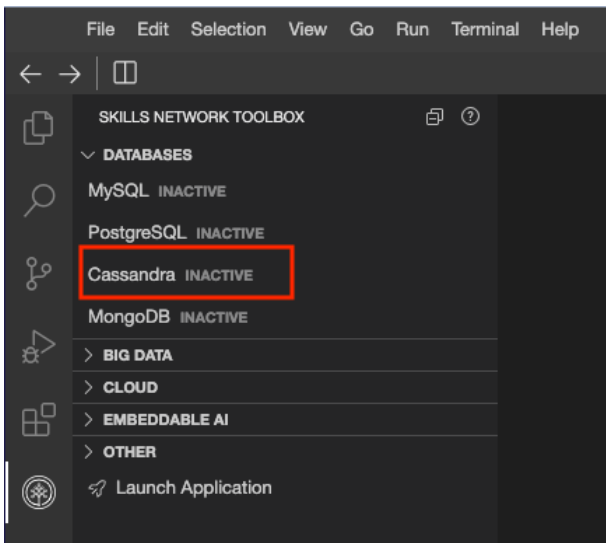
Please be aware that sessions for this lab environment are not persisted. Every time you connect to this lab, a new environment is created for you. Any data you may have saved in the earlier session would get lost. Plan to complete these labs in a single session, to avoid losing your data.

Set-up: Start Cassandra

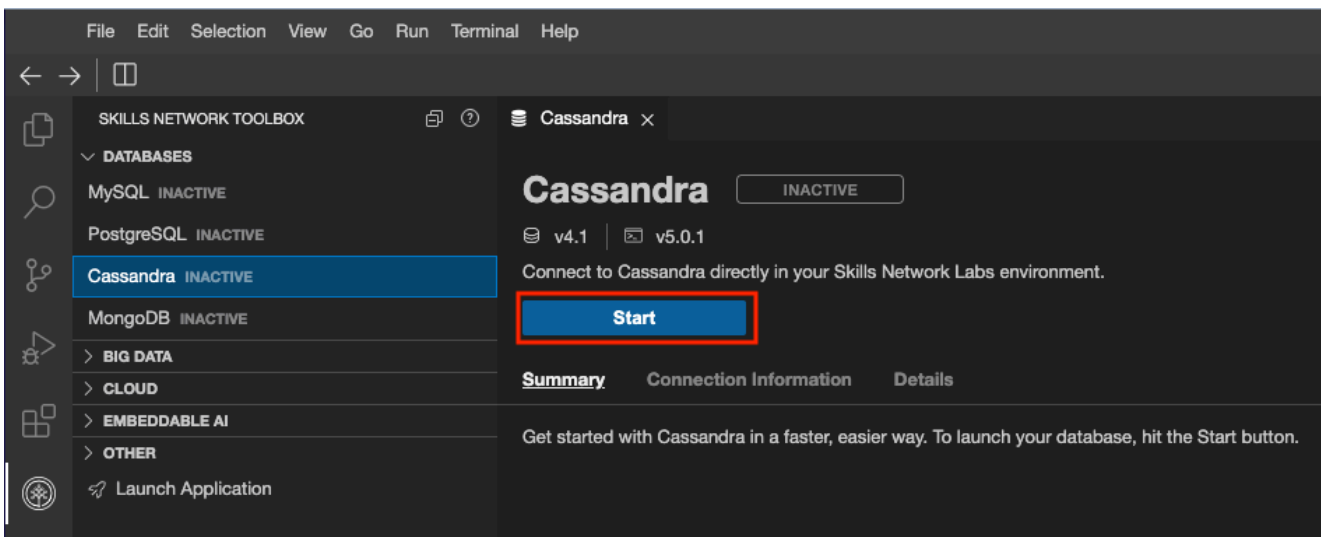
Navigate to Skills Network Toolbox.



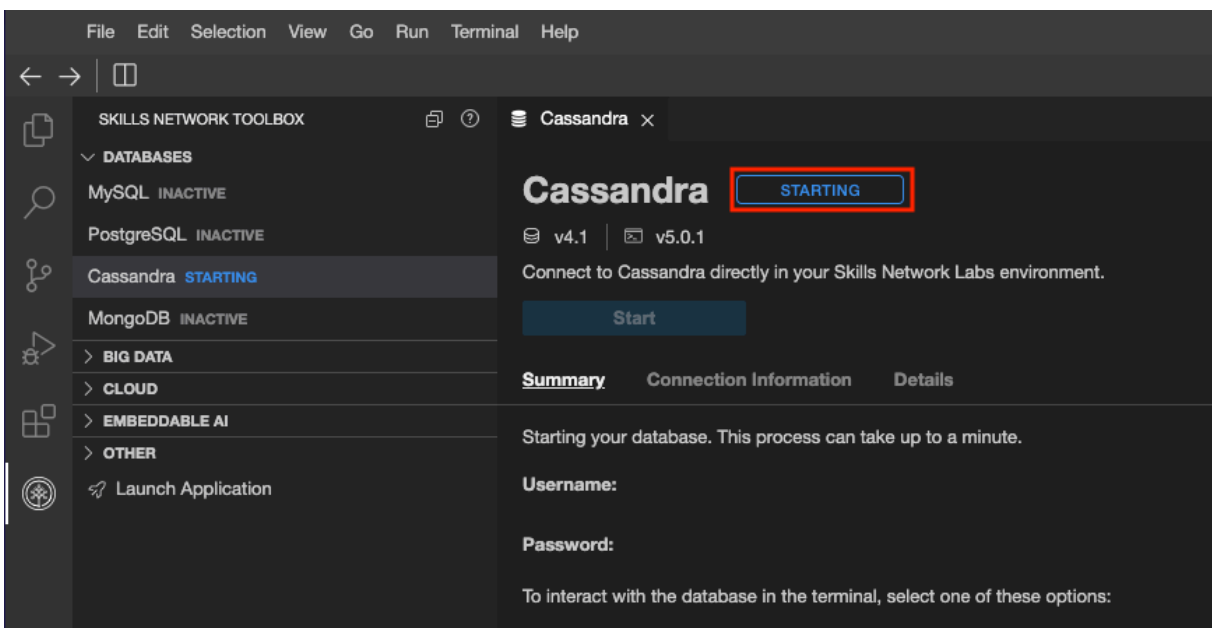
You will notice Cassandra listed there, but inactive. Which means the database is not available to use.



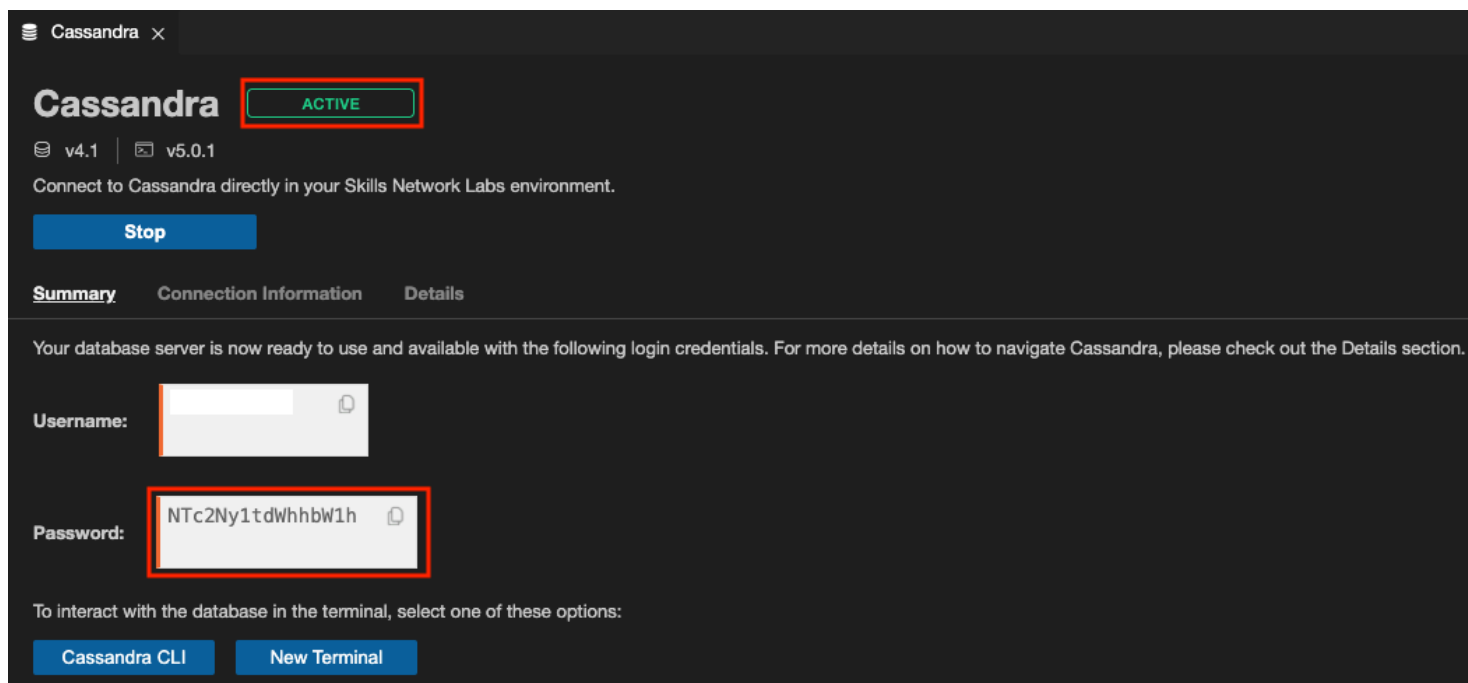
Once you click on it, you will see more details about it and a button to start it.



Clicking on the start button will run a background process to configure and start your Cassandra server.



Shortly after that, your server is ready for use. This deployment has access control enabled and Cassandra enforces authentication. So, take note of the password as you will need it to login as cassandra user.



Cassandra ACTIVE

v4.1 | v5.0.1

Connect to Cassandra directly in your Skills Network Labs environment.

Stop

Summary | Connection Information | Details

Your database server is now ready to use and available with the following login credentials. For more details on how to navigate Cassandra, please check out the Details section.

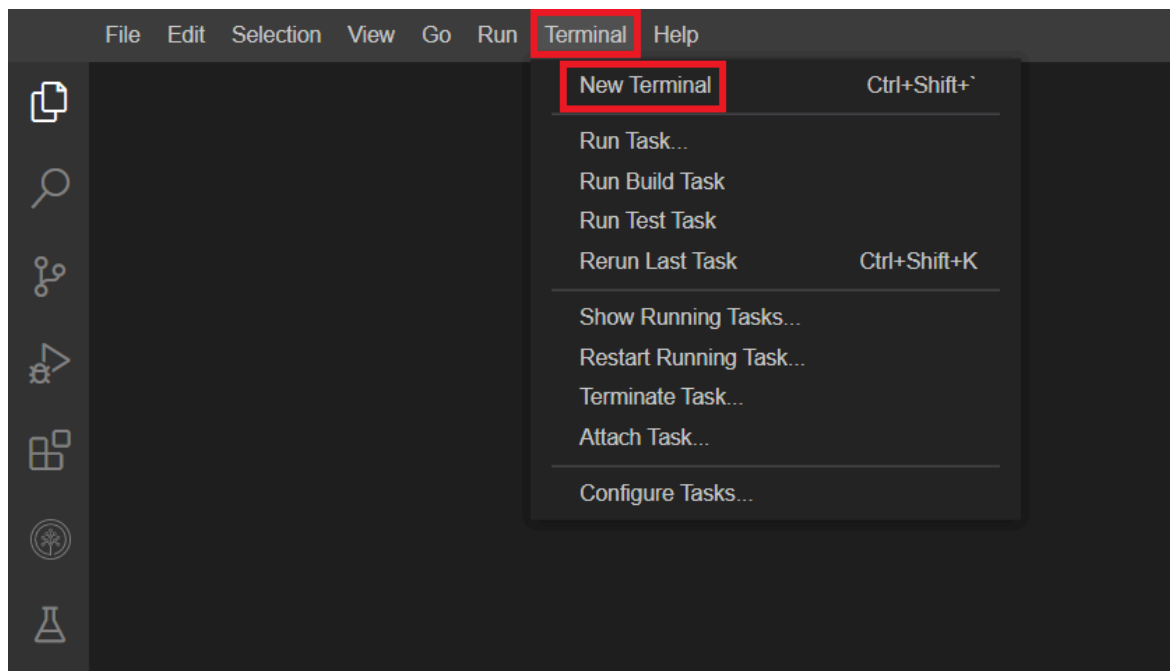
Username: [Redacted]

Password: NTc2Ny1tdWhhbW1h [Redacted]

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

Cassandra CLI | **New Terminal**

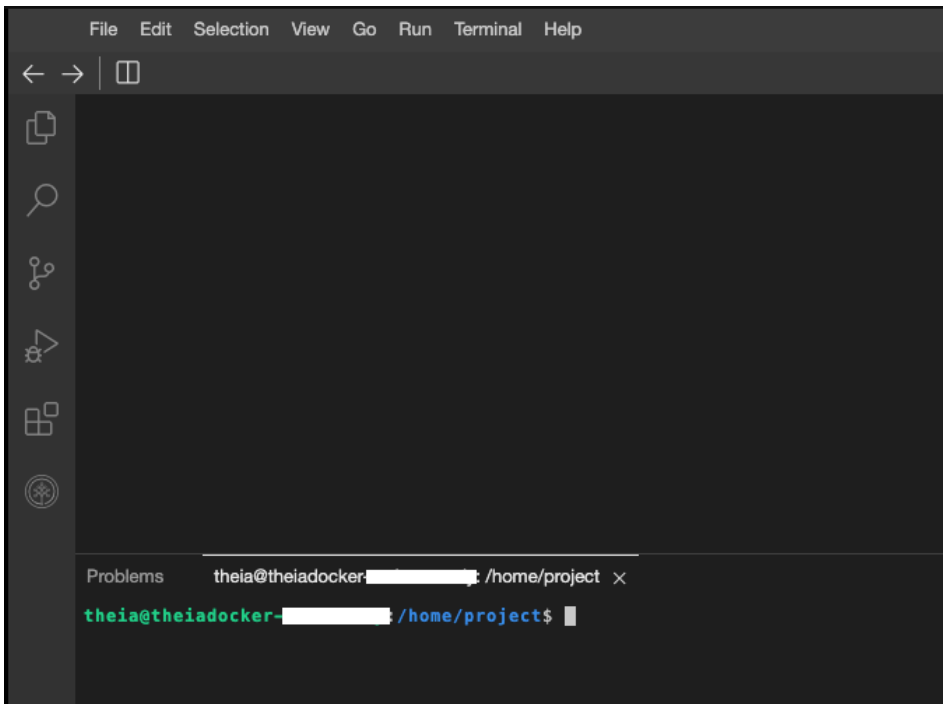
You can now either open terminal and enter details yourself.



File | Edit | Selection | View | Go | Run | **Terminal** | Help

- New Terminal** (Ctrl+Shift+')
- Run Task...
- Run Build Task
- Run Test Task
- Rerun Last Task (Ctrl+Shift+K)
- Show Running Tasks...
- Restart Running Task...
- Terminate Task...
- Attach Task...
- Configure Tasks...

This will open a new terminal at the bottom of the screen as in the image below.



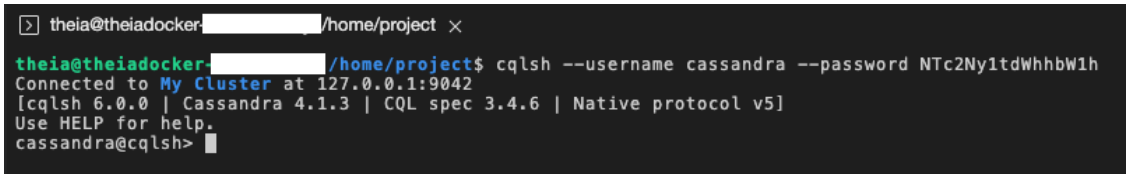
Run the below command on the newly opened terminal. (You can copy the code by clicking on the little copy button on the bottom right of the codeblock below and then paste it, wherever you wish)

1. 1

1. `cqlsh --username cassandra --password PASSWORD`

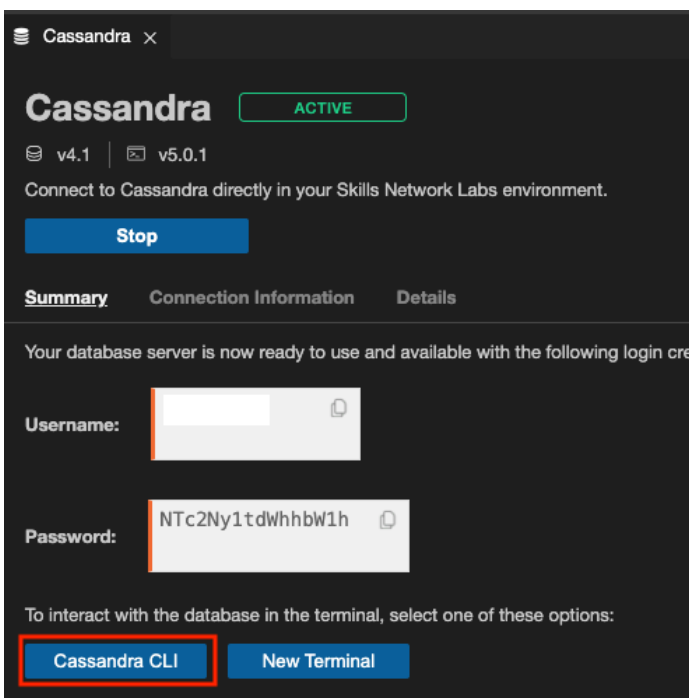
Copied!

Executed!



The command contains the username and password to connect to Cassandra server. Your output could be different from the one shown above. Copy the command given to you, and keep it handy. You will need it in the next step.

Or you can simply click on Cassandra CLI which does that for you.



Exercise 1 - Create Keyspace and Table

Create training keyspace

Create a keyspace named `training` using `SimpleStrategy` and `replication_factor` of 3.

- [Click here for Hint](#)
- [Click here for Solution](#)

Create movies table

Create a table named `movies` with columns:

- `movie_id` is an integer and is the primary key.
- `movie_name` is a text column.
- `year_of_release` is an integer.

- [Click here for Hint](#)
- [Click here for Solution](#)

Exercise 2 - Insert data into a table

Let's insert a row into the table `movies`.

On `cqlsh` run the below command.

```
1. 1
2. 2
3. 3

1. INSERT into movies(
2. movie_id, movie_name, year_of_release)
3. VALUES (1, 'Toy Story', 1995);
```

Copied!

Verify that the data is saved.

```
1. 1

1. select * from movies;
```

Copied!

Insert the below movies into the table.

movie_id	movie_name	year_of_release
2	Jumanji	1995
3	Heat	1995
4	Scream	1995
5	Fargo	1996

- [Click here for Solution](#)

Exercise 3 - Read data from a table

In the previous exercise you have inserted some data into the table named `movies`.

Let's query the data in the `movies` table.

Query all rows.

```
1. 1

1. select * from movies;
```

Copied!

Query the movie name where `movie_id` is 1.

```
1. 1

1. select movie_name from movies where movie_id = 1;
```

Copied!

Exercise 4 - Update data in a table

The `movie_id` for *Scream* is 4. It was released in 1996 and not 1995.

Here is how you modify it.

- 1.
- 2.
- 3.

```
1. UPDATE movies
2. SET year_of_release = 1996
3. WHERE movie_id = 4;
```

Copied!

Verify that the update was successful.

- 1.

```
1. select * from movies where movie_id = 4;
```

Copied!

Exercise 5 - Delete data from a table

Delete the movie with the `movie_id` 5.

- 1.
- 2.

```
1. DELETE from movies
2. WHERE movie_id = 5;
```

Copied!

Verify using the below command.

- 1.

```
1. select * from movies;
```

Copied!

Practice exercises

1. Problem: Insert the below movie into the `movies` table.

```
movie_id movie_name year_of_release
6        Twister    1997
```

- [Click here for Hint](#)
- [Click here for Solution](#)

2. Problem: Modify the release year of *Twister* to 1996.

- [Click here for Hint](#)
- [Click here for Solution](#)

3. Problem: Delete the movie *Twister*.

- [Click here for Hint](#)
- [Click here for Solution](#)

4. Problem: Drop the `training` keyspace.

- [Click here for Hint](#)
- [Click here for Solution](#)

Summary

In this lab, you have gained an understanding of CRUD operations in Cassandra.

Author(s)

[Muhammad Yahya](#)

Changelog

Date	Version	Changed by	Change Description
2024-01-10	0.5	Mercedes Schneider	QA pass with edits
2023-10-09	0.4	Muhammad Yahya	Refreshed

(C) IBM Corporation 2023. All rights reserved.