Getting Started with Hive

Estimated time needed: 20 minutes

In this lab you will explore Apache Hive, a distributed, fault-tolerant data warehouse system that enables analytics at a massive scale. You will be creating a table and running SQL commands on it.

Learning Objectives

At the end of this lab, you will be able to:

- Create a table in Hive
- Add data to the table using file
- Add data to the table using `insert`
- Query the data in the table using SQL commands

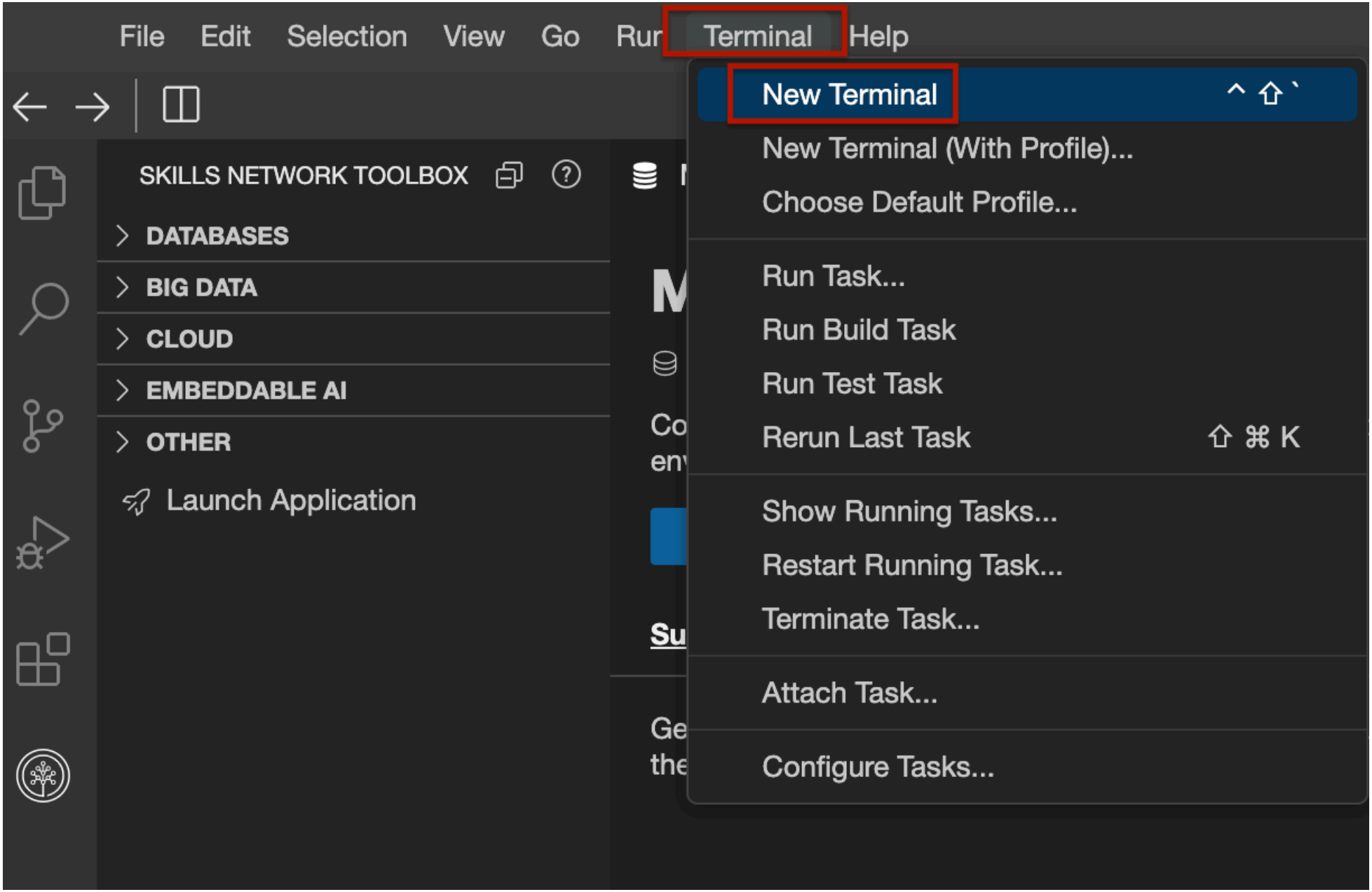
Prerequisites

- You should comfortable working with the Linux terminal
- Prior knowledge of SQL will be helpful

  While all the terminal commands can be copy pasted and run, it is highly recommended for you to type the commands for better learning.

Step 1: Get a copy of the CSV file

1. You will run the commands in the terminal. If you don't have a terminal open, open a new terminal, by clicking on `Terminal` and choosing `New Terminal` from the submenu.



2. Create a directory named `data` under `/home/project` by running the following command.

```
1
```

```
mkdir /home/project/data
```

Copied!

3. Change to the `/home/project/data` directory.

```
1
```

```
cd /home/project/data
```

Copied!

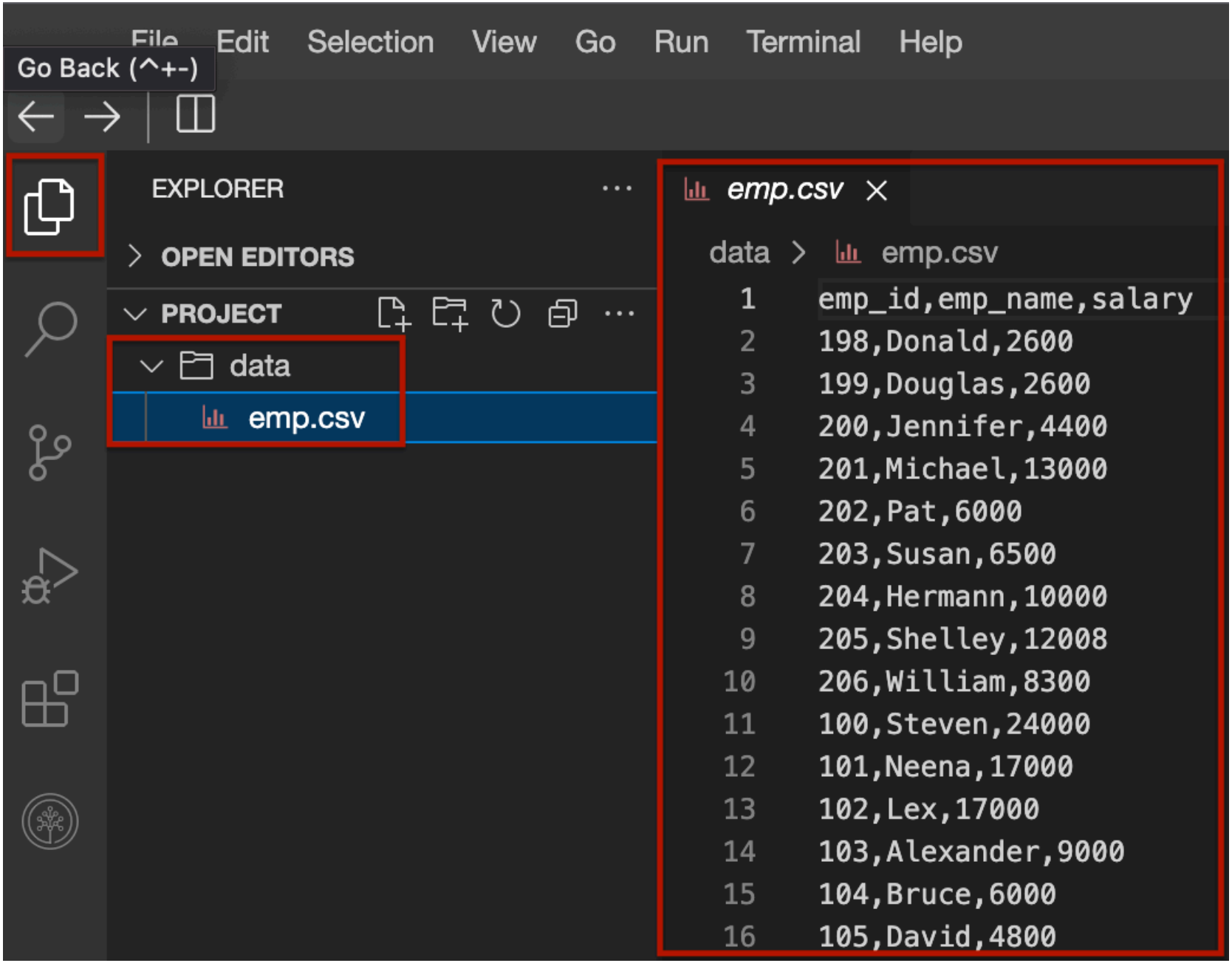
4. Run the following command to get the `emp.csv`, a data file with Employee data, in a comma-separated file which you will use later to infuse data into the table you create.

```
1
```

```
wget https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-BD0225EN-SkillsNetwork/data/emp.csv
```

Copied!

5. Open the file in editor and view the file.



Step 2: Setup Hive and Bee

1. You will use the hive from the docker hub for this lab. Pull the hive image into your system by running the following command.

```
1
```

```
docker pull apache/hive:4.0.0-alpha-1
```

Copied! Executed!

This will take a few seconds, depending on the speed of your internet connection.

2. Now, you will run the hive server on port `10002`. You will name the server instance `myhiveserver`. We will mount the local `data` folder in the hive server as `hive_custom_data`. This would mean that the whole `data` folder that you created locally, along with anything you add in the data folder, is copied into the container under the directory `hive_custom_data`.

```
1
```

```
docker run -d -p 10000:10000 -p 10002:10002 --env SERVICE_NAME=hiveserver2 -v /home/project/data:/hive_custom_data --name myhiveserver apache/hive:4.0.0-alpha-1
```

Copied! Executed!

3. You can open and take a look at the Hive server with the GUI. Click the button to open the HiveServer2 GUI.

HiveServer2 GUI

4. Now run the following command, which allows you to access `beeline`. This is a SQL cli where you can create, modify, delete table, and access data in the table.

```
1
```

```
docker exec -it myhiveserver beeline -u 'jdbc:hive2://localhost:10000/'
```

Copied! Executed!

Step 3: Create table, add and view data

1. To create a new table `Employee` with three columns as in the csv you downloaded - em_id, emp_name and salary, run the following command.

```
1
```

```
create table Employee(emp_id string, emp_name string, salary  int)  row format delimited fields terminated by ',' ;
```

Copied!

You may notice that there is an explicit mention for the fields delimited by `,` just as in the csv file.

2. Run the following command to check if the table is created.

```
1
```

```
show tables;
```

Copied!

This should list the Employee table that you just created.

3. Now load the data into the table from the csv file by running the following command.

```
1
```

```
1. LOAD DATA INPATH '/hive_custom_data/emp.csv' INTO TABLE Employee;
```

Copied!

3. Run the following command to list all the rows from the table to check if the data has been loaded from the CSV.

```
1. 1
```

```
1. SELECT * FROM employee;
```

Copied!

4. You can view the details of the commands and the outcome in the HiveServer2 GUI.

HiveServer2 GUI

5. To quit from the beehive prompt in the terminal, press `ctrl+D`.

Hive internally uses MapReduce to process and analyze data. When you execute a Hive query, it generates MapReduce jobs that run on the Hadoop cluster.

Conclusion

In this lab you created a table in hive, added data to the table from csv and listed the data contained in the table.

Next Steps

You can explore more SQL commands with table and see how it works.

Author(s)

Lavanya T S