# Exp5: Installation of Hive on Ubuntu

**Aim:**

To Download and install Hive, Understanding Startup scripts, Configuration files.

# Procedure:

**Step 1: Download and extract it**

Download the Apache hive and extract it use tar, the commands given below:

*$wgethttps://downloads.apache.org/hive/hive-3.1.2/apache-hive-3.1.2-bin.tar.gz*

*$ tar –xvf apache-hive-3.1.2-bin.tar.gz*

# Step 2: Place different configuration properties in Apache Hive

In this step, we are going to do two things

o Placing Hive Home path in bashrc file

$*nano .bashrc And append the below lines in it*



2. Exporting **Hadoop path in Hive-config.sh** (To communicate with the Hadoop eco system we are defining Hadoop Home path in hive config field) **Open the hive- config.sh as shown in below**

*$cd apache-hive-3.1.2-bin/bin*

*$cp hive-env.sh.template hive-env.sh*

*$nano hive-env.sh*

*Append the below commands on it*

*export HADOOP\_HOME=/home/Hadoop/Hadoop*

*export HIVE\_CONF\_DIR=/home/Hadoop/apache-hive-3.1.2/conf*



# Step 3: Install mysql

1. Install mysql in Ubuntu by running this command:

*$sudo apt update*

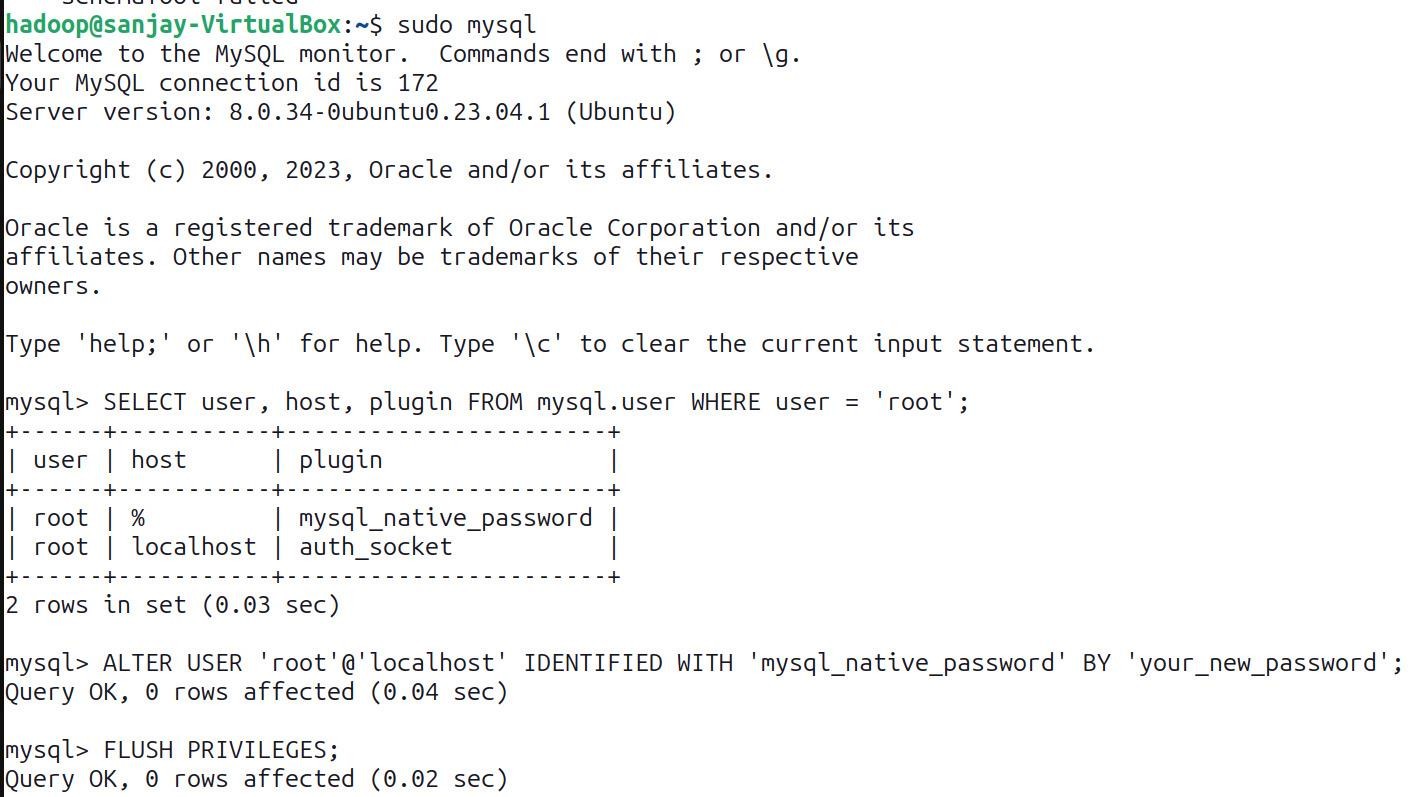
*$sudo apt install mysql-server*

1. *Alter username and password for MySQLby running below commands:*

*$sudomysql*

Pops command line interface for MySQLand run the below SQL queries to change username and set password

*mysql> SELECT user, host, plugin FROM mysql.user WHERE user = 'root';*



*mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH 'mysql\_native\_password' BY 'your\_new\_password';*

*mysql> FLUSH PRIVILEGES;*

# Step 4:Config hive-site.xml

Config the hive-site.xml by appending this xml code and change the username and password according to your MySQL.

*$cd apache-hive-3.1.2-bin/bin*

*$cp hive-default.xml.template hive-site.xml*

*$nano hive-site.xml Append these lines into it*

*Replace root as your username of MySQL Replaceyour\_new\_password as with your password of MySQL*

*<configuration>*

*<property>*

*<name>javax.jdo.option.ConnectionURL</name>*

*<value>jdbc:mysql://localhost/metastore?createDatabaseIfNotExist=true</value>*

*</property>*

*<property>*

*<name>javax.jdo.option.ConnectionDriverName</name>*

*<value>com.mysql.cj.jdbc.Driver</value>*

*</property>*

*<property>*

*<name>javax.jdo.option.ConnectionUserName</name>*

*<value>root</value>*

*</property>*

*<property>*

*<name>javax.jdo.option.ConnectionPassword</name>*

*<value>your\_new\_password</value>*

*</property>*

*<property>*

*<name>datanucleus.autoCreateSchema</name>*

*<value>true</value>*

*</property>*

*<property>*

*<name>datanucleus.fixedDatastore</name>*

*<value>true</value>*

*</property>*

*<property>*

*<name>datanucleus.autoCreateTables</name>*

*<value>True</value>*

*</property>*

*</configuration>*

# Step 5: Setup MySQL java connector:

*First, you'll need to download the MySQL Connector/J, which is the JDBC driver for MySQL. You can download it from the below link* [https://drive.google.com/file/d/1QFhB7Kvcat7a4LzDRe6GcmZva1yAxKz-](https://drive.google.com/file/d/1QFhB7Kvcat7a4LzDRe6GcmZva1yAxKz-/view?usp=drive_link)

[/view?usp=drive\_link](https://drive.google.com/file/d/1QFhB7Kvcat7a4LzDRe6GcmZva1yAxKz-/view?usp=drive_link)

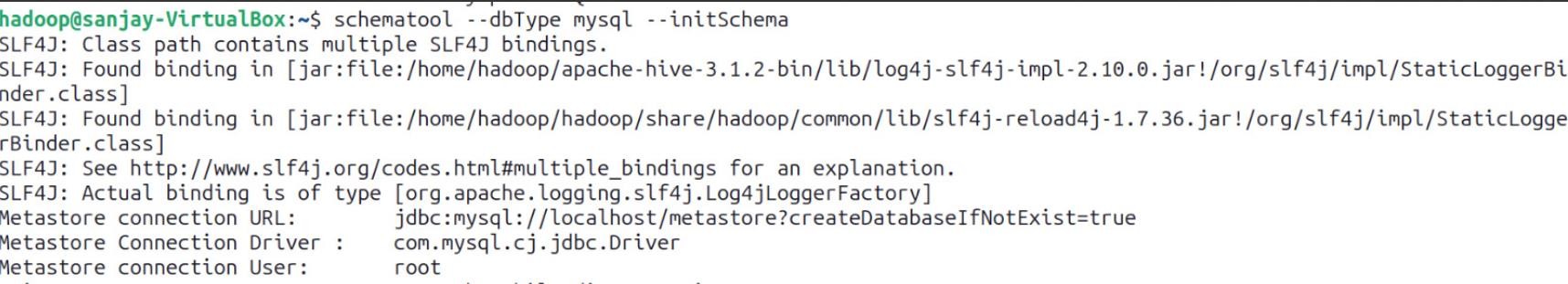
Copy the downloaded MySQL Connector/J JAR file to the Hive library directory. By default, the Hive library directory is usually located at*/path/to/apache-hive-3.1.2/lib/*on Ubuntu. Use the following command to copy the JAR file:

*$sudo cp /path/to/mysql-connector-java-8.0.15.jar /path/to/apache-hive-3.1.2/lib/ Replace /path/to/ with the actual path to the JAR file.*

# Step 6:Initialize the Hive Metastore Schema:

*Run the following command to initialize the Hive metastore schema:*

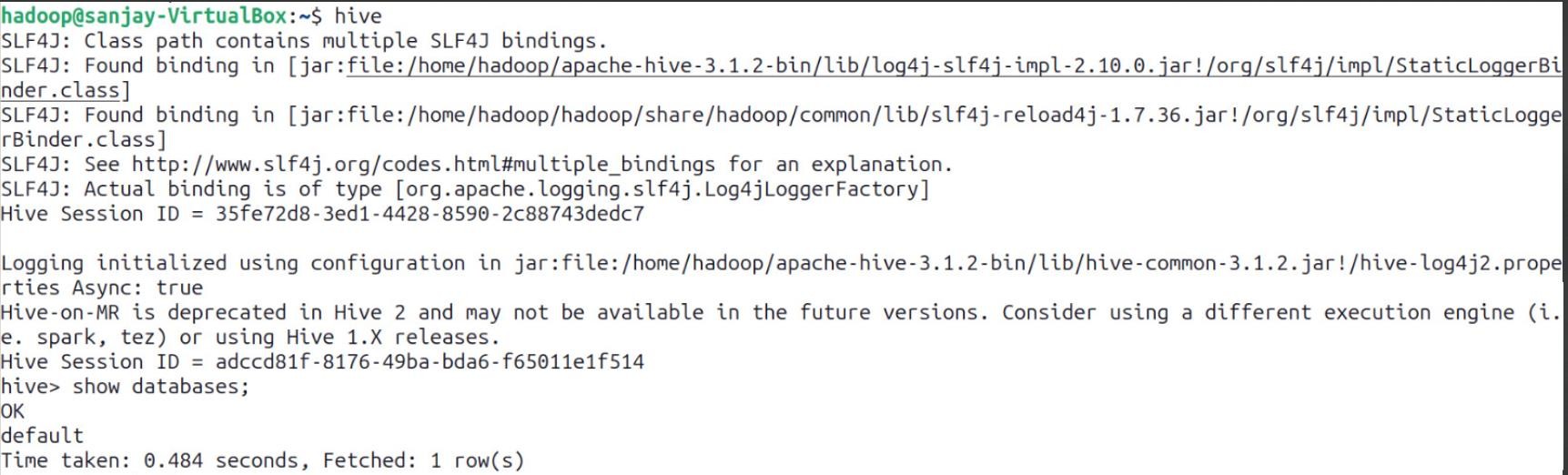
*$$HIVE\_HOME/bin/schematool -initSchema -dbTypemysql*



# Step 7: Start hive:

You can test Hive by running the Hive shell: Copy code hive You should be able to run Hive queries, and metadata will be stored in your MySQL database.

*$hive*



# Result:

Thus, the Apache Hive installation is completed successfully on Ubuntu.

# Exp5a: Design and test various schema models to optimize data storage and retrieval Using Hive.

**Aim:**

To Design and test various schema models to optimize data storage and retrieval Using Hbase.

# Procedure:

**Step 1: Start Hive**

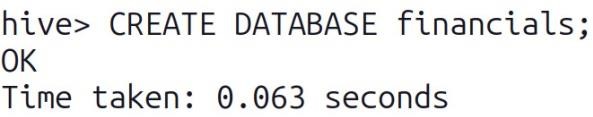
Open a terminal and start Hive by running:

$hive

# Step 2: Create a Database

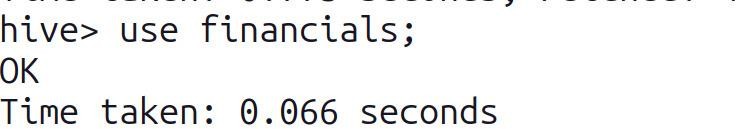
Create a new database in Hive:

hive>CREATE DATABASE financials;



## Step 3: Use the Database:

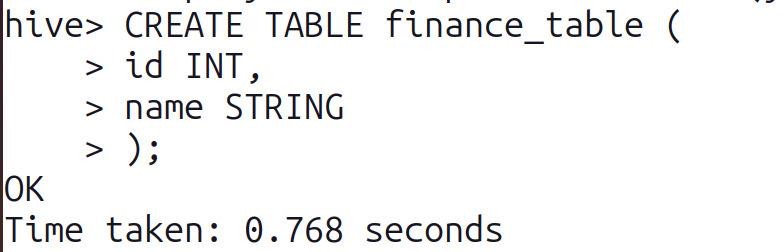
*Switch to the newly created database:*

*hive>use financials;*

## Step 4: Create a Table:

*Create a simple table in your database:*

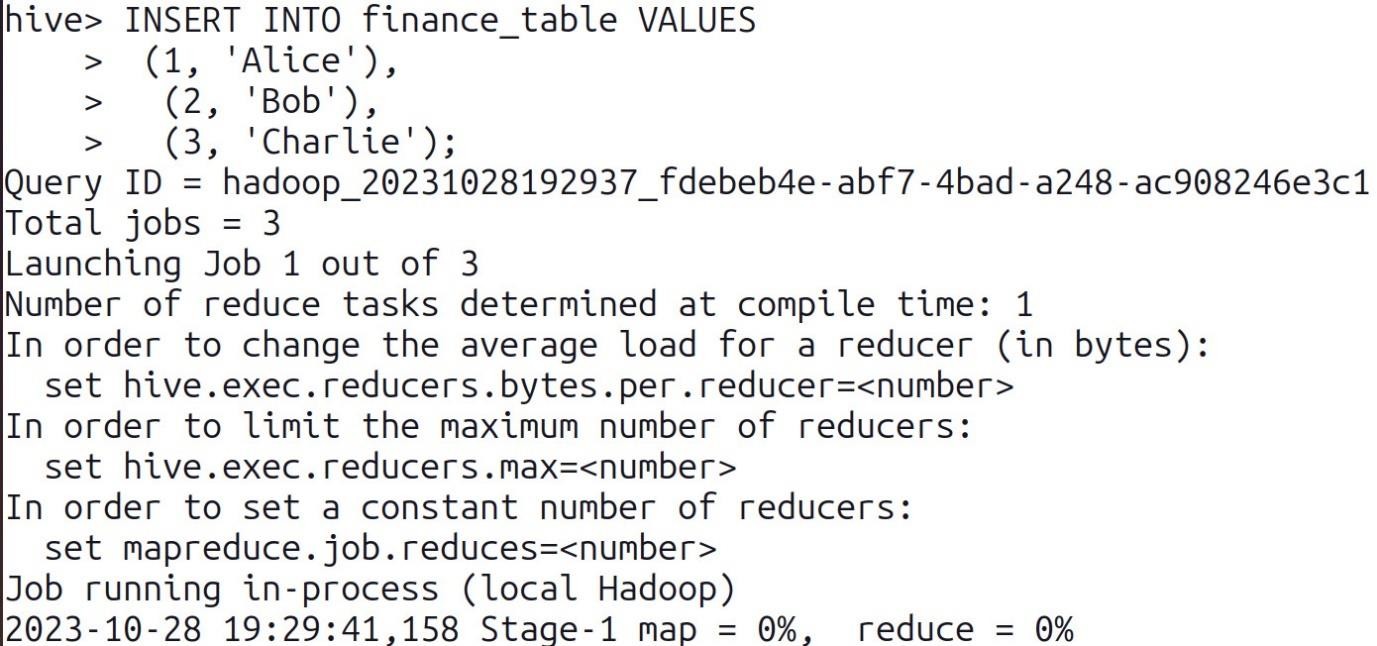
*hive>CREATE TABLE finance\_table( id INT, name STRING );*



## Step 5: Load Sample Data:

*You can insert sample data into the table:*

*hive>INSERT INTO finance\_tableVALUES (1, 'Alice'), (2, 'Bob'), (3, 'Charlie');*



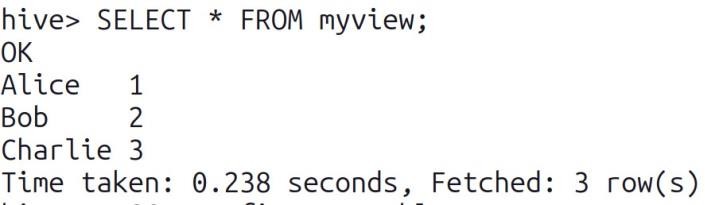
## Step 6: Query Your Data

*Use SQL-like queries to retrieve data from your table:*

*hive>CREATE VIEW myview AS SELECT name, id FROM finance\_table;*

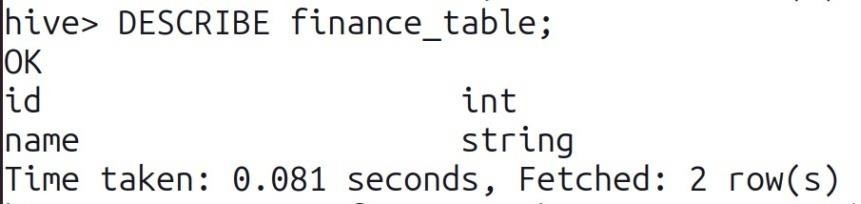
## Step 7: View the data:

*To see the data in the view, you would need to query the view hive>SELECT\*FROM myview;*



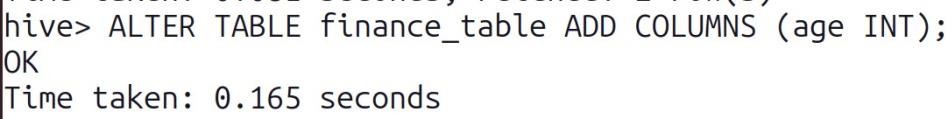
## Step 8: Describe a Table:

*You can describe the structure of a table using the DESCRIBE command: hive>DESCRIBE finance\_table;*



## Step 9: Alter a Table:

*You can alter the table structure by adding a new column: hive>ALTER TABLE finance\_table ADD COLUMNS (age INT);*

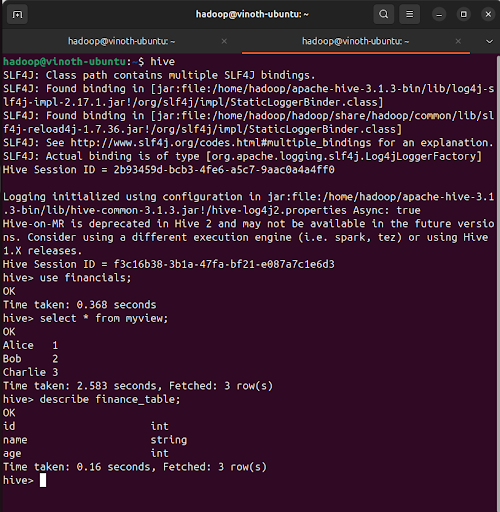


## Step 10: Quit Hive:

*To exit the Hive CLI, simply type: hive>quit;*



**Output:**

****

# Result:

Thus, the usage of various commands in Hive has been successfully completed.