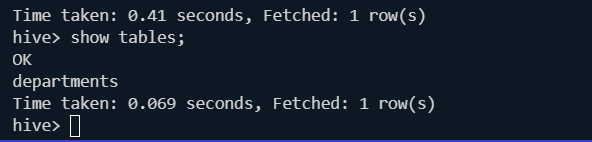
Hive command:

Hive

Show databases;



Show tables



Create table:

create database retail\_db;

use retail\_db;

create table if not exists orders(

    order\_id INT,

    order\_date STRING,

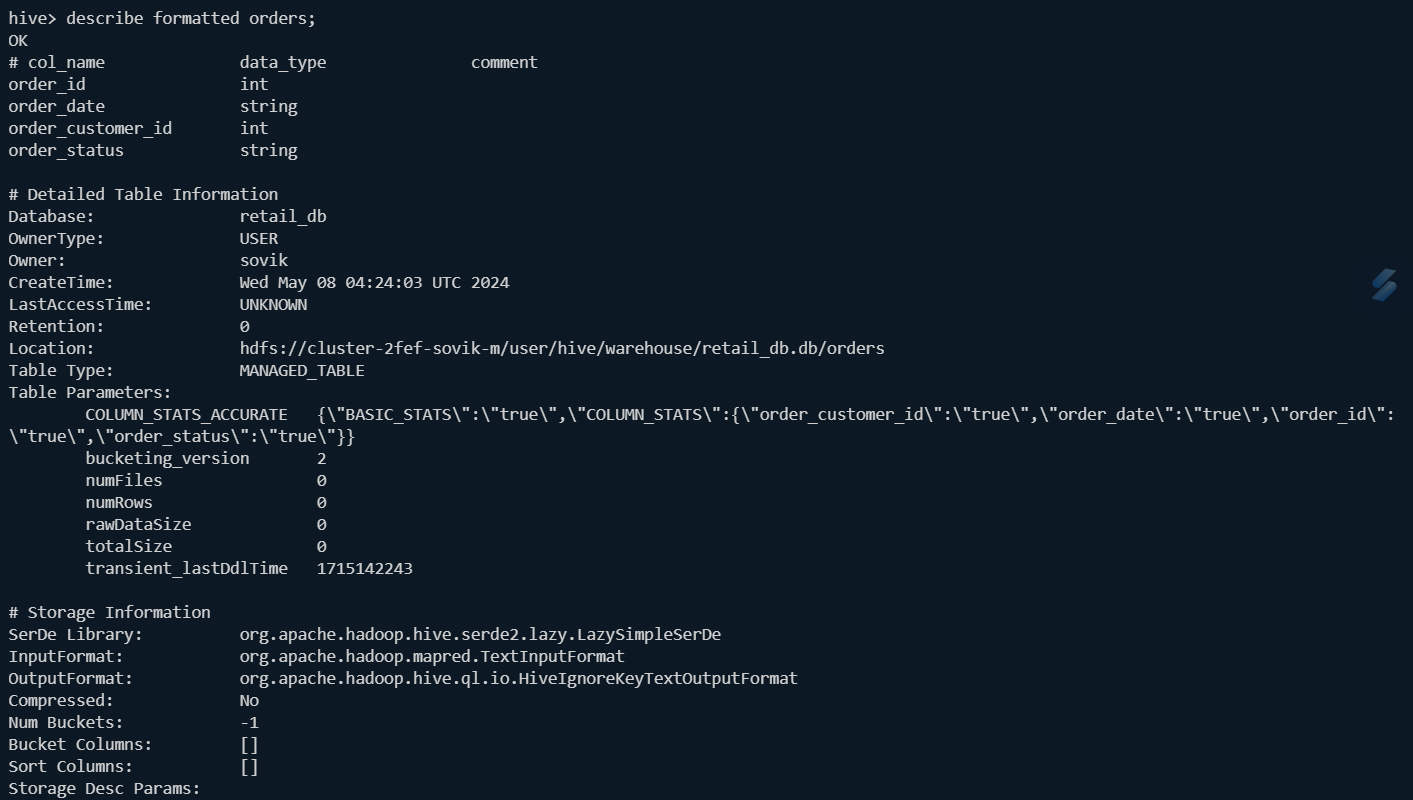
    order\_customer\_id INT,

    order\_status STRING

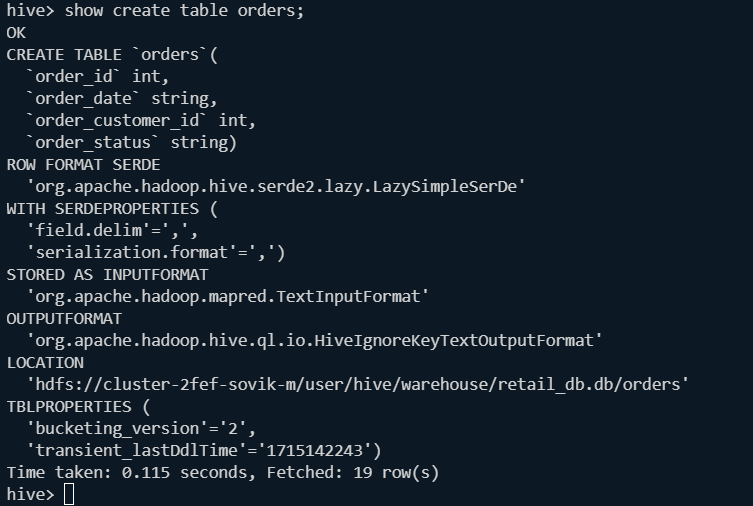
) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' ;

Check:

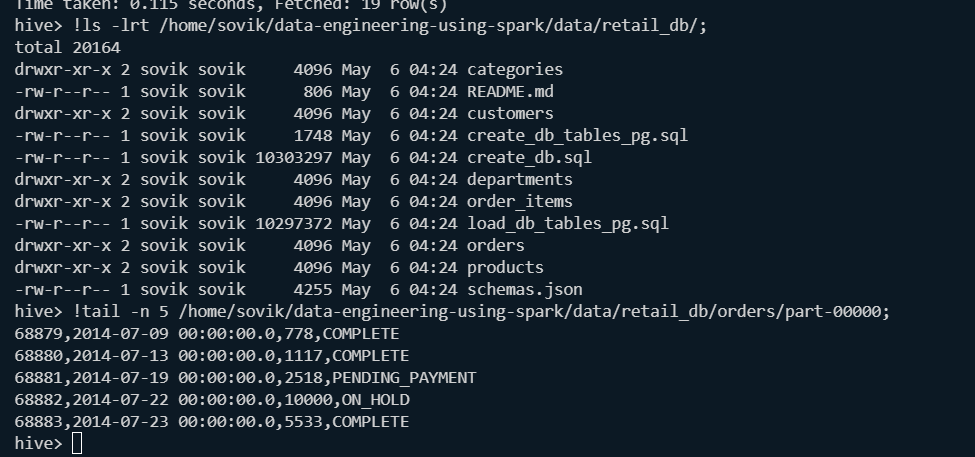
Describe formatted <table\_name>;



Show the create table statement:



Checking path from where we are going to fetch the data:



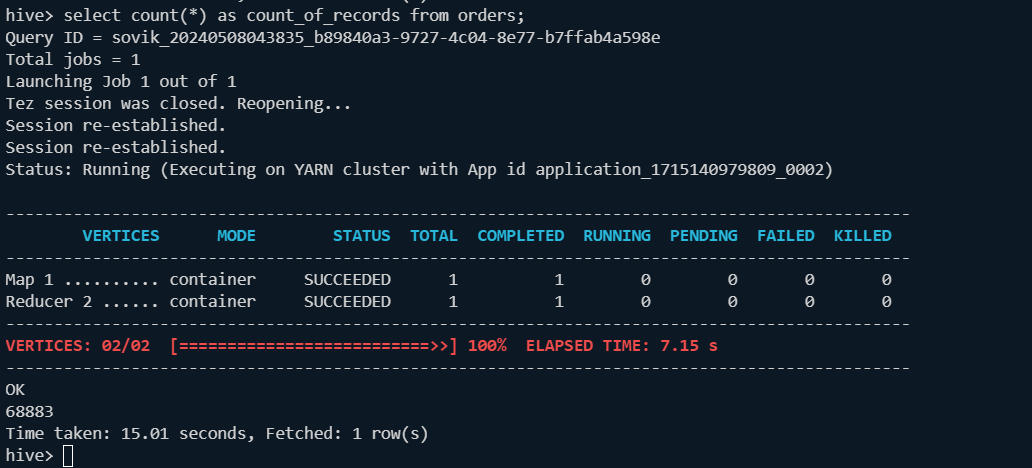
Load data into the table:

LOAD DATA LOCAL INPATH '/home/sovik/data-engineering-using-spark/data/retail\_db/orders/' into table orders;

Validate the table:



See if map reduce is working internally:



All Scripts:

create table if not exists orders(

    order\_id INT,

    order\_date STRING,

    order\_customer\_id INT,

    order\_status STRING

) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' ;

#!ls -lrt "path";

DESCRIBE formatted <table\_name>;

LOAD DATA LOCAL INPATH '/home/sovik/data-engineering-using-spark/data/retail\_db/orders/' into table orders;

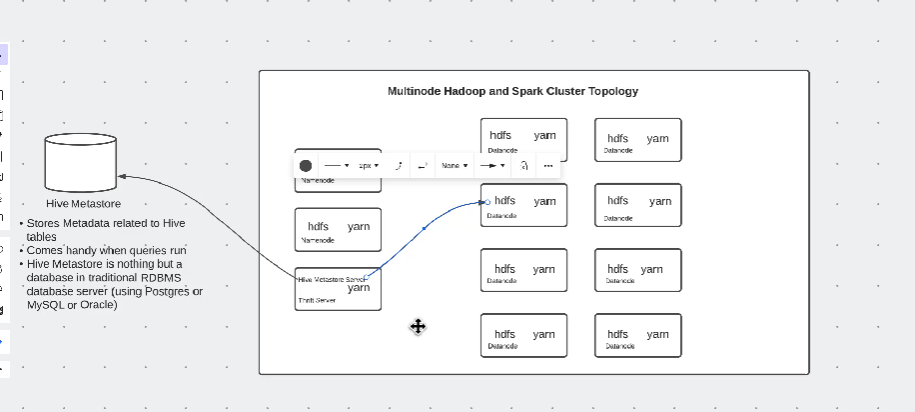
 select \* from orders LIMIT 20;

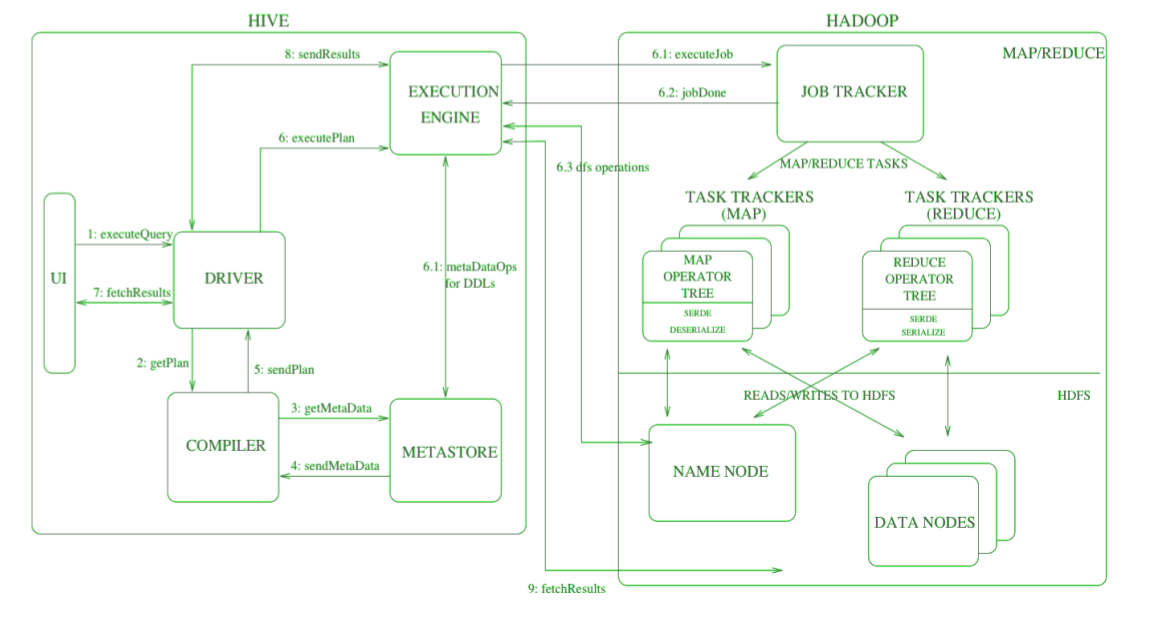
select count(\*) as count\_of\_records from orders;

show tables;

select current\_database();

**Hive Architecture:**





**Beeline:**

Beeline is a command-line interface (CLI) tool for interacting with Apache Hive, which is a data warehouse system built on top of Hadoop. Beeline serves as a thin client that communicates with a Hive server using the HiveServer2 JDBC interface. It allows you to run queries, execute Hive scripts, and interact with Hive tables and data from the command line or from within scripts.

Here are some key features and characteristics of Beeline:

1. **JDBC Client**: Beeline connects to the HiveServer2 using the JDBC protocol, which is a standard way to connect Java applications to databases. This makes it more flexible and compatible with other JDBC clients.
2. **Interactive Shell**: Beeline provides an interactive shell where you can execute HiveQL queries and commands. It allows you to connect to a Hive server, run queries, view results, and perform other data management tasks.
3. **Remote Connection**: Unlike the older Hive CLI, which connected directly to Hive, Beeline connects remotely to the Hive server. This design aligns better with modern security practices and allows for better scalability.
4. **Command-Line Tool**: Beeline is a command-line tool that you can use for scripting and automation. You can write scripts that use Beeline to execute Hive queries and perform other tasks.
5. **Security**: Beeline supports various authentication mechanisms such as Kerberos and SSL for secure connections to Hive servers.
6. **Configurable**: Beeline has options and settings that you can configure according to your requirements, such as connection URL, user credentials, and other connection properties.

Here’s how you might use Beeline:

* **Starting Beeline**: To start Beeline, you can usually run the **beeline** command in the terminal. This may vary depending on how you installed Hive.
* **Connecting to Hive**: You need to specify the connection URL (e.g., **jdbc:hive2://hostname:port/**) and optionally provide user credentials. For example:

shell

!connect jdbc:hive2://localhost:10000/default user password!connect jdbc:hive2://localhost:10000/default user password

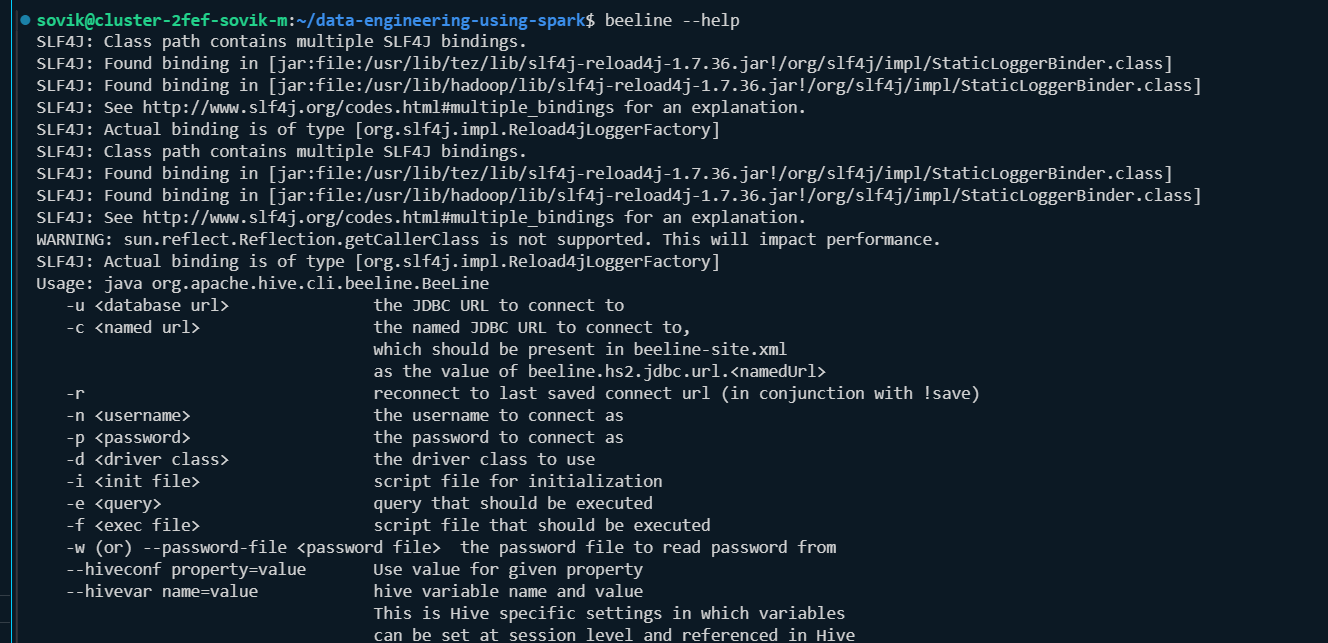
* **Running Queries**: Once connected, you can run HiveQL queries using the command-line prompt. For example:

shell

SELECT \* FROM my\_table;SELECT \* FROM my\_table;

* **Exiting Beeline**: You can exit the Beeline session with the command **!quit**.

Overall, Beeline is a flexible and powerful tool for interacting with Hive and can be a useful tool in your data processing and management workflows.

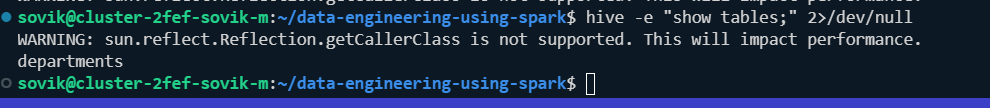


**How to run multiple hive command:**

**hive -e "USE retail\_db;show tables;"**



discarding any error output (such as potential warnings) to prevent it from being displayed by using /dev/null



To run the commands from scripts:

Script to run:

create database if not exists retail\_db;

use retail\_db;

create table if not exists orders(

    order\_id INT,

    order\_date STRING,

    order\_customer\_id INT,

    order\_status STRING

) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' ;

# !ls -lrt "path";

# DESCRIBE formatted <table\_name>;

LOAD DATA LOCAL INPATH '/home/sovik/data-engineering-using-spark/data/retail\_db/orders/' into table orders;

select \* from orders LIMIT 20;

# select count(\*) as count\_of\_records from orders;

# show tables;

# select current\_database();

# DROP table orders;

Command:

sovik@cluster-2fef-sovik-m:~/data-engineering-using-spark$ **chmod 777 hive\_scripts/create\_tables.hql**

sovik@cluster-2fef-sovik-m:~/data-engineering-using-spark$ **hive -f hive\_scripts/\*.hql**

SLF4J: Class path contains multiple SLF4J bindings.

SLF4J: Found binding in [jar:file:/usr/lib/tez/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: Found binding in [jar:file:/usr/lib/hadoop/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: See http://www.slf4j.org/codes.html#multiple\_bindings for an explanation.

SLF4J: Actual binding is of type [org.slf4j.impl.Reload4jLoggerFactory]

SLF4J: Class path contains multiple SLF4J bindings.

SLF4J: Found binding in [jar:file:/usr/lib/tez/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: Found binding in [jar:file:/usr/lib/hadoop/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: See http://www.slf4j.org/codes.html#multiple\_bindings for an explanation.

WARNING: sun.reflect.Reflection.getCallerClass is not supported. This will impact performance.

SLF4J: Actual binding is of type [org.slf4j.impl.Reload4jLoggerFactory]

Hive Session ID = b5f5da96-dc56-4f5e-ae68-64ba15ef5c9b

Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j2.properties Async: true

WARNING: An illegal reflective access operation has occurred

WARNING: Illegal reflective access by org.apache.hadoop.hive.common.StringInternUtils (file:/usr/lib/hive/lib/hive-common-3.1.3.jar) to field java.net.URI.string

WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.hive.common.StringInternUtils

WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations

WARNING: All illegal access operations will be denied in a future release

Hive Session ID = 7f0e9831-98e4-4546-9c54-fdd415a11655

OK

Time taken: 0.645 seconds

OK

Time taken: 0.037 seconds

OK

Time taken: 0.302 seconds

Loading data to table retail\_db.orders

OK

Time taken: 0.32 seconds

OK

1 2013-07-25 00:00:00.0 11599 CLOSED

2 2013-07-25 00:00:00.0 256 PENDING\_PAYMENT

3 2013-07-25 00:00:00.0 12111 COMPLETE

4 2013-07-25 00:00:00.0 8827 CLOSED

5 2013-07-25 00:00:00.0 11318 COMPLETE

6 2013-07-25 00:00:00.0 7130 COMPLETE

7 2013-07-25 00:00:00.0 4530 COMPLETE

8 2013-07-25 00:00:00.0 2911 PROCESSING

9 2013-07-25 00:00:00.0 5657 PENDING\_PAYMENT

10 2013-07-25 00:00:00.0 5648 PENDING\_PAYMENT

11 2013-07-25 00:00:00.0 918 PAYMENT\_REVIEW

12 2013-07-25 00:00:00.0 1837 CLOSED

13 2013-07-25 00:00:00.0 9149 PENDING\_PAYMENT

14 2013-07-25 00:00:00.0 9842 PROCESSING

15 2013-07-25 00:00:00.0 2568 COMPLETE

16 2013-07-25 00:00:00.0 7276 PENDING\_PAYMENT

17 2013-07-25 00:00:00.0 2667 COMPLETE

18 2013-07-25 00:00:00.0 1205 CLOSED

19 2013-07-25 00:00:00.0 9488 PENDING\_PAYMENT

20 2013-07-25 00:00:00.0 9198 PROCESSING

Time taken: 1.644 seconds, Fetched: 20 row(s)

sovik@cluster-2fef-sovik-m:~/data-engineering-using-spark$

**Now drop the table:**

**ovik@cluster-2fef-sovik-m:~/data-engineering-using-spark$ *hive -e "USE retail\_db;drop table orders;show tables;"***

SLF4J: Class path contains multiple SLF4J bindings.

SLF4J: Found binding in [jar:file:/usr/lib/tez/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: Found binding in [jar:file:/usr/lib/hadoop/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: See http://www.slf4j.org/codes.html#multiple\_bindings for an explanation.

SLF4J: Actual binding is of type [org.slf4j.impl.Reload4jLoggerFactory]

SLF4J: Class path contains multiple SLF4J bindings.

SLF4J: Found binding in [jar:file:/usr/lib/tez/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: Found binding in [jar:file:/usr/lib/hadoop/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: See http://www.slf4j.org/codes.html#multiple\_bindings for an explanation.

WARNING: sun.reflect.Reflection.getCallerClass is not supported. This will impact performance.

SLF4J: Actual binding is of type [org.slf4j.impl.Reload4jLoggerFactory]

Hive Session ID = f21dd2e1-3421-4fb1-bd39-ddba68897ba8

Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j2.properties Async: true

WARNING: An illegal reflective access operation has occurred

WARNING: Illegal reflective access by org.apache.hadoop.hive.common.StringInternUtils (file:/usr/lib/hive/lib/hive-common-3.1.3.jar) to field java.net.URI.string

WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.hive.common.StringInternUtils

WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations

WARNING: All illegal access operations will be denied in a future release

Hive Session ID = ea066df6-8a08-4b99-adaf-ccc2aa0705e7

OK

Time taken: 0.711 seconds

OK

Time taken: 0.16 seconds

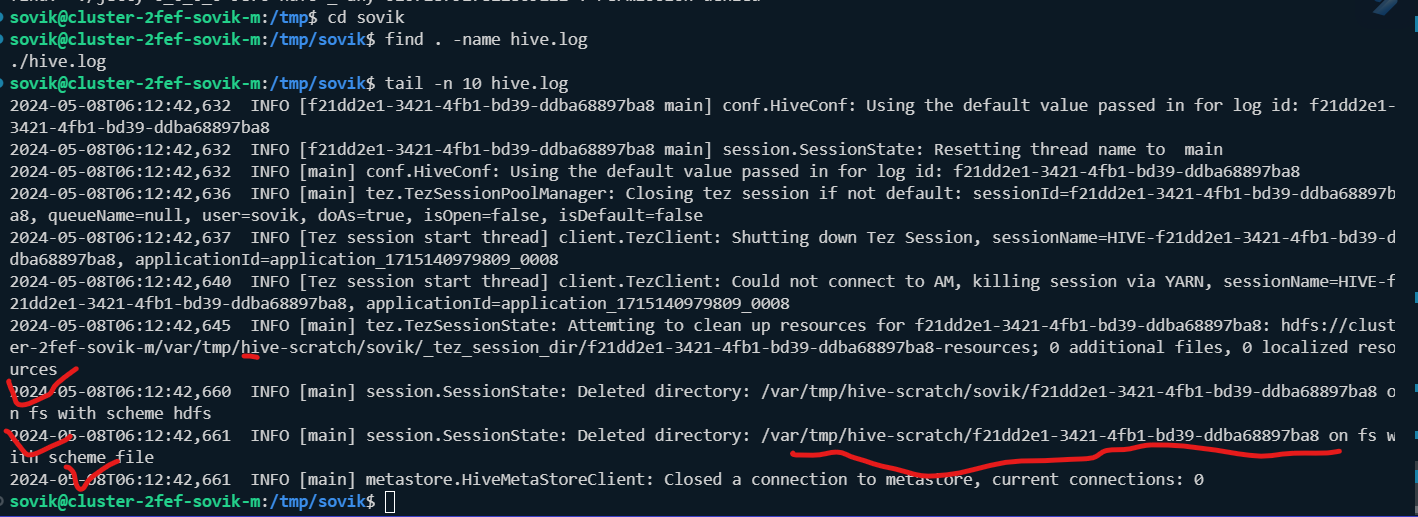
OK

Time taken: 0.167 seconds

sovik@cluster-2fef-sovik-m:~/data-engineering-using-spark$

**Find logs:**

**/tmp/Sovik/**

****

**Environment variable in hive:**

**SET;**

**Check** *hive.metastore.warehouse.dir:*

**sovik@cluster-2fef-sovik-m:~/data-engineering-using-spark$ hive -e "SET" | grep hive.metastore.warehouse**

SLF4J: Class path contains multiple SLF4J bindings.

SLF4J: Found binding in [jar:file:/usr/lib/tez/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: Found binding in [jar:file:/usr/lib/hadoop/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: See http://www.slf4j.org/codes.html#multiple\_bindings for an explanation.

SLF4J: Actual binding is of type [org.slf4j.impl.Reload4jLoggerFactory]

SLF4J: Class path contains multiple SLF4J bindings.

SLF4J: Found binding in [jar:file:/usr/lib/tez/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: Found binding in [jar:file:/usr/lib/hadoop/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: See http://www.slf4j.org/codes.html#multiple\_bindings for an explanation.

SLF4J: Actual binding is of type [org.slf4j.impl.Reload4jLoggerFactory]

Hive Session ID = fac8cd0e-915f-4434-8092-e3089552b124

Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j2.properties Async: true

WARNING: An illegal reflective access operation has occurred

WARNING: Illegal reflective access by org.apache.hadoop.hive.common.StringInternUtils (file:/usr/lib/hive/lib/hive-common-3.1.3.jar) to field java.net.URI.string

WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.hive.common.StringInternUtils

WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations

WARNING: All illegal access operations will be denied in a future release

Hive Session ID = 9407c4d1-0792-45e3-bba1-5c988dbd79c7

**hive.metastore.warehouse.dir=/user/hive/warehouse**