

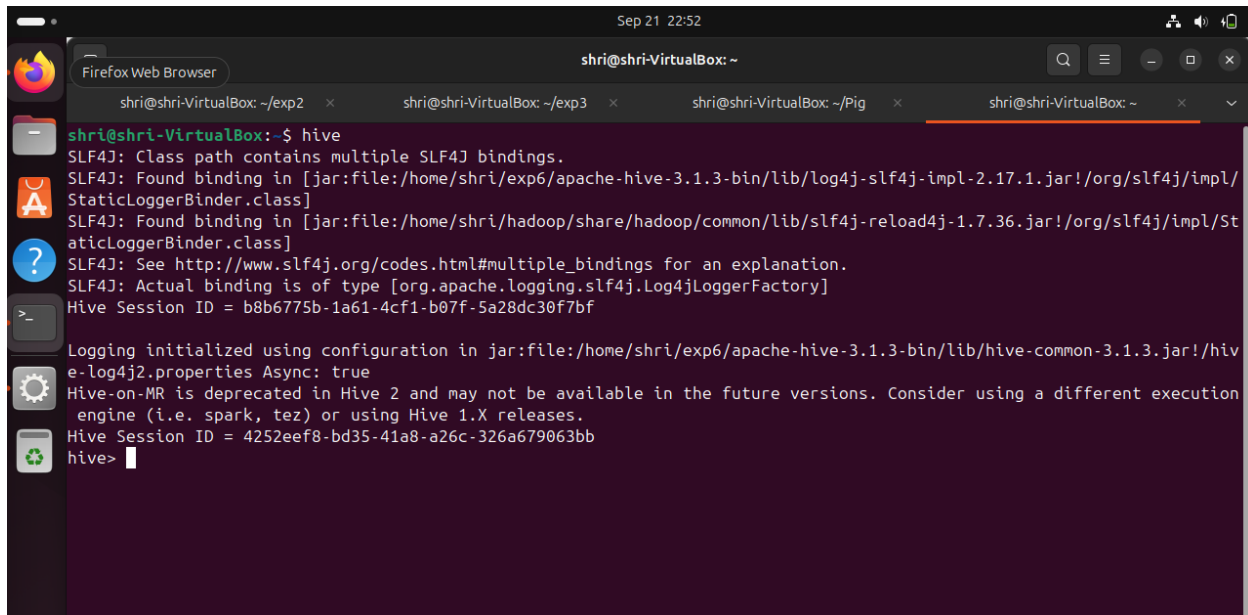
**Exp.No: 5****Install Hive and Create Tables in Hive and write queries to access the data in the table****AIM:**

To install Hive, design and test various schema models to optimize data storage and retrieval using Hive.

**PROCEDURE:**

**Step 1:** Start Hive Open a terminal and start Hive by running:

hive



```
shri@shri-VirtualBox:~$ hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/shri/exp6/apache-hive-3.1.3-bin/lib/log4j-slf4j-impl-2.17.1.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/shri/hadoop/share/hadoop/common/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.apache.logging.slf4j.Log4jLoggerFactory]
Hive Session ID = b8b6775b-1a61-4cf1-b07f-5a28dc30f7bf

Logging initialized using configuration in jar:file:/home/shri/exp6/apache-hive-3.1.3-bin/lib/hive-common-3.1.3.jar!/hive-log4j2.properties Async: true
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
Hive Session ID = 4252eef8-bd35-41a8-a26c-326a679063bb
hive>
```

**Step 2: Create a Database:** Create a new database in Hive:

hive> CREATE DATABASE sports;

```
hive> CREATE DATABASE sports;
OK
Time taken: 0.522 seconds
hive>
```

**Step 3: Use the Database:** Switch to the newly created database:

**hive> USE sports;**

```
hive> USE sports;
OK
Time taken: 0.066 seconds
hive>
```

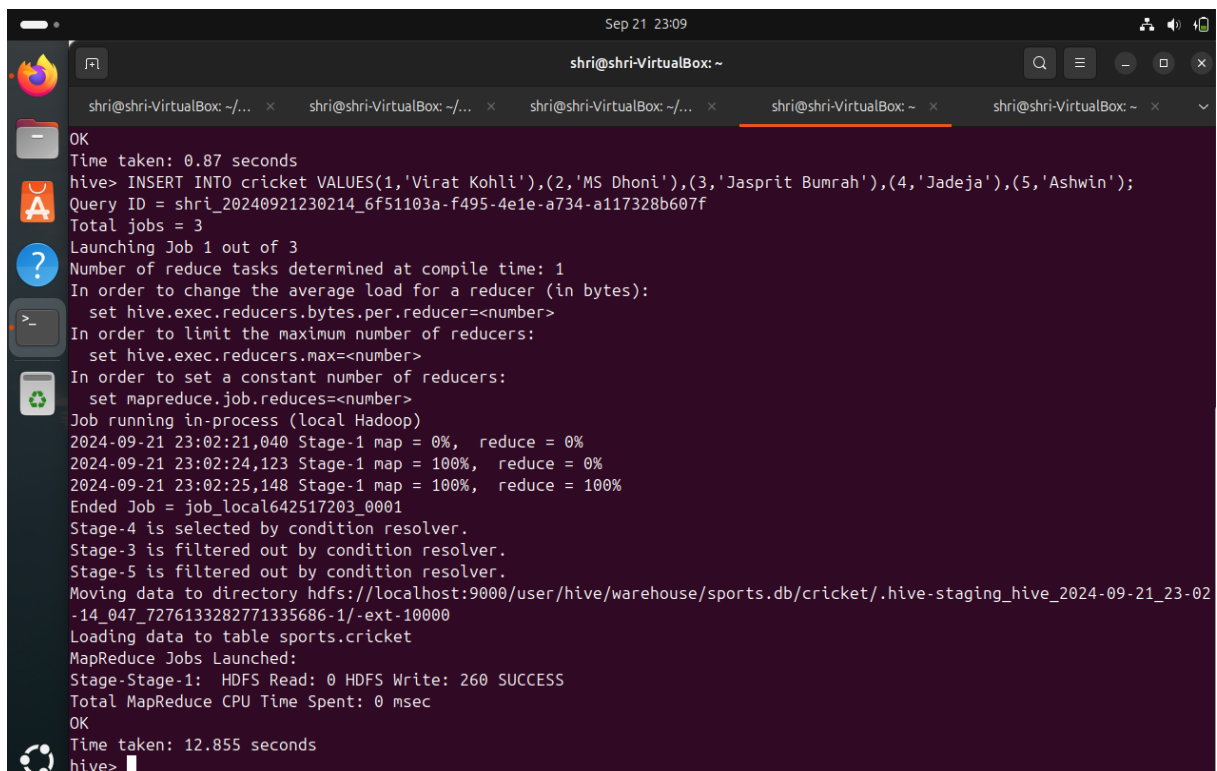
**Step 4: Create a Table:** Create a simple table in your database:

**hive> CREATE TABLE cricket( id INT, name STRING );**

```
hive> CREATE TABLE cricket(id INT , name STRING);
OK
Time taken: 0.87 seconds
hive>
```

**Step 5: Load Sample Data:** You can insert sample data into the table:

**hive> INSERT INTO cricket VALUES (1, 'Virat Kohli'),(2, 'MS Dhoni'),(3, 'Jasprit Bumrah'),(4,'Jadeja'),(5,'Ashwin');**



```
Sep 21 23:09
shri@shri-VirtualBox: ~
shri@shri-VirtualBox: ~/... x  shri@shri-VirtualBox: ~/... x  shri@shri-VirtualBox: ~/... x  shri@shri-VirtualBox: ~ x  shri@shri-VirtualBox: ~ x
OK
Time taken: 0.87 seconds
hive> INSERT INTO cricket VALUES(1,'Virat Kohli'),(2,'MS Dhoni'),(3,'Jasprit Bumrah'),(4,'Jadeja'),(5,'Ashwin');
Query ID = shri_20240921230214_6f51103a-f495-4e1e-a734-a117328b607f
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Job running in-process (local Hadoop)
2024-09-21 23:02:21,040 Stage-1 map = 0%,  reduce = 0%
2024-09-21 23:02:24,123 Stage-1 map = 100%,  reduce = 0%
2024-09-21 23:02:25,148 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local642517203_0001
Stage-4 is selected by condition resolver.
Stage-3 is filtered out by condition resolver.
Stage-5 is filtered out by condition resolver.
Moving data to directory hdfs://localhost:9000/user/hive/warehouse/sports.db/cricket/.hive-staging_hive_2024-09-21_23-02-14_047_7276133282771335686-1/-ext-10000
Loading data to table sports.cricket
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 0 HDFS Write: 260 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
Time taken: 12.855 seconds
hive>
```

**Step 6: Query Your Data:** Use SQL-like queries to retrieve data from your table:

**hive> CREATE VIEW mycricket AS SELECT id, name FROM cricket;**

```
hive> CREATE VIEW mycricket AS SELECT id,name FROM cricket;
OK
Time taken: 2.702 seconds
hive>
```

**Step 7: View the data:** To see the data in the view, you would need to query the view

**hive> SELECT \* FROM mycricket;**

```
hive> Select * FROM mycricket;
OK
1      Virat Kohli
2      MS Dhoni
3      Jasprit Bumrah
4      Jadeja
5      Ashwin
Time taken: 0.595 seconds, Fetched: 5 row(s)
```

**Step 8: Describe a Table:** You can describe the structure of a table using the DESCRIBE command:

**hive> DESCRIBE cricket;**

```
hive> DESCRIBE cricket;
OK
id          int
name        string
Time taken: 0.189 seconds, Fetched: 2 row(s)
```

**Step 9: Alter a Table:** You can alter the table structure by adding a new column:

**hive> ALTER TABLE cricket ADD COLUMNS(runs\_scored INT);**

```
hive> ALTER TABLE cricket ADD COLUMNS(runs_scored INT);
OK
Time taken: 0.384 seconds
hive> |
```

**Step 10: Quit Hive:** To exit the Hive CLI, simply type:  
**hive> quit;**

```
hive> quit;  
shri@shri-VirtualBox:~$
```

**RESULT:**

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