Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

Department of Computer Science and Engineering (Data Science)

Name : Rishabh Singhvi **Sap id : 60009210206** **Div : D22**

Subject: Big Data Engineering (DJ19DSL604)

AY: 2023-24

Experiment 6

(Data Warehouse)

Aim: Implement data warehousing using HIVE.

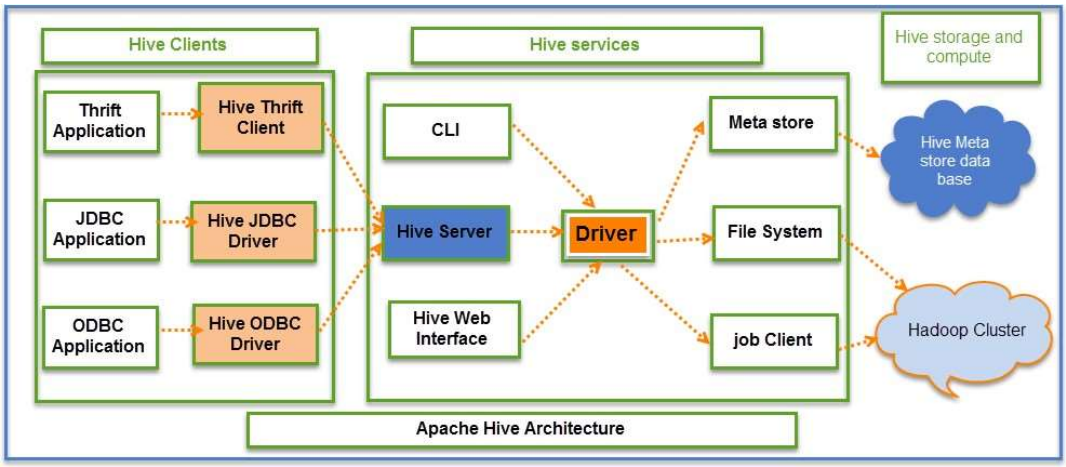
Theory:

Introduction to HIVE

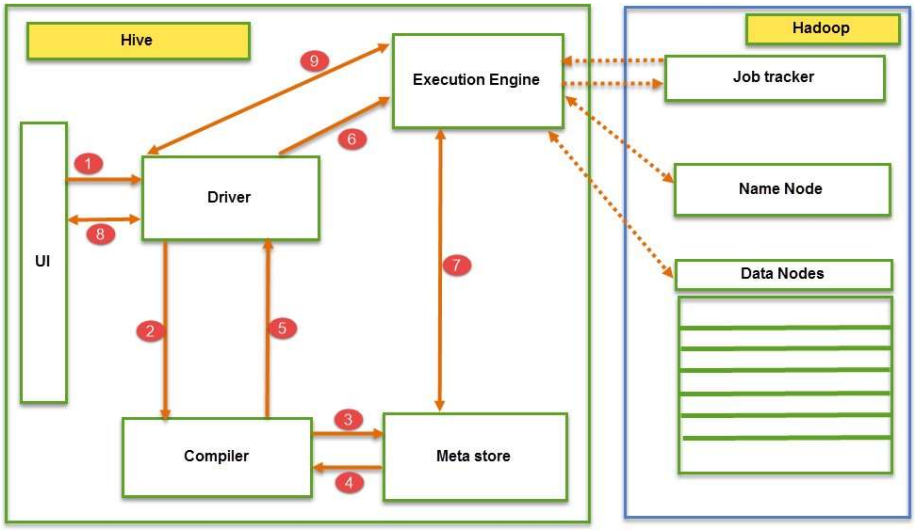
Hive as an ETL and data warehousing tool on top of Hadoop ecosystem provides functionalities like Data modeling, Data manipulation, Data processing and Data querying. Data Extraction in Hive means the creation of tables in Hive and loading structured and semi structured data as well as querying data based on the requirements.

For batch processing, we are going to write custom defined scripts using a custom map and reduce scripts using a scripting language. It provides SQL like environment and support for easy querying.

HIVE Architecture



Job execution flow:



Different modes of Hive:

Hive can operate in two modes depending on the size of data nodes in Hadoop.

These modes are,

Local mode

- **Map reduce mode** When to use Local mode:

- If the Hadoop installed under pseudo mode with having one data node we use Hive in this mode If the data size is smaller in term of limited to single local machine, we can use this mode Processing will be very fast on smaller data sets present in the local machine.

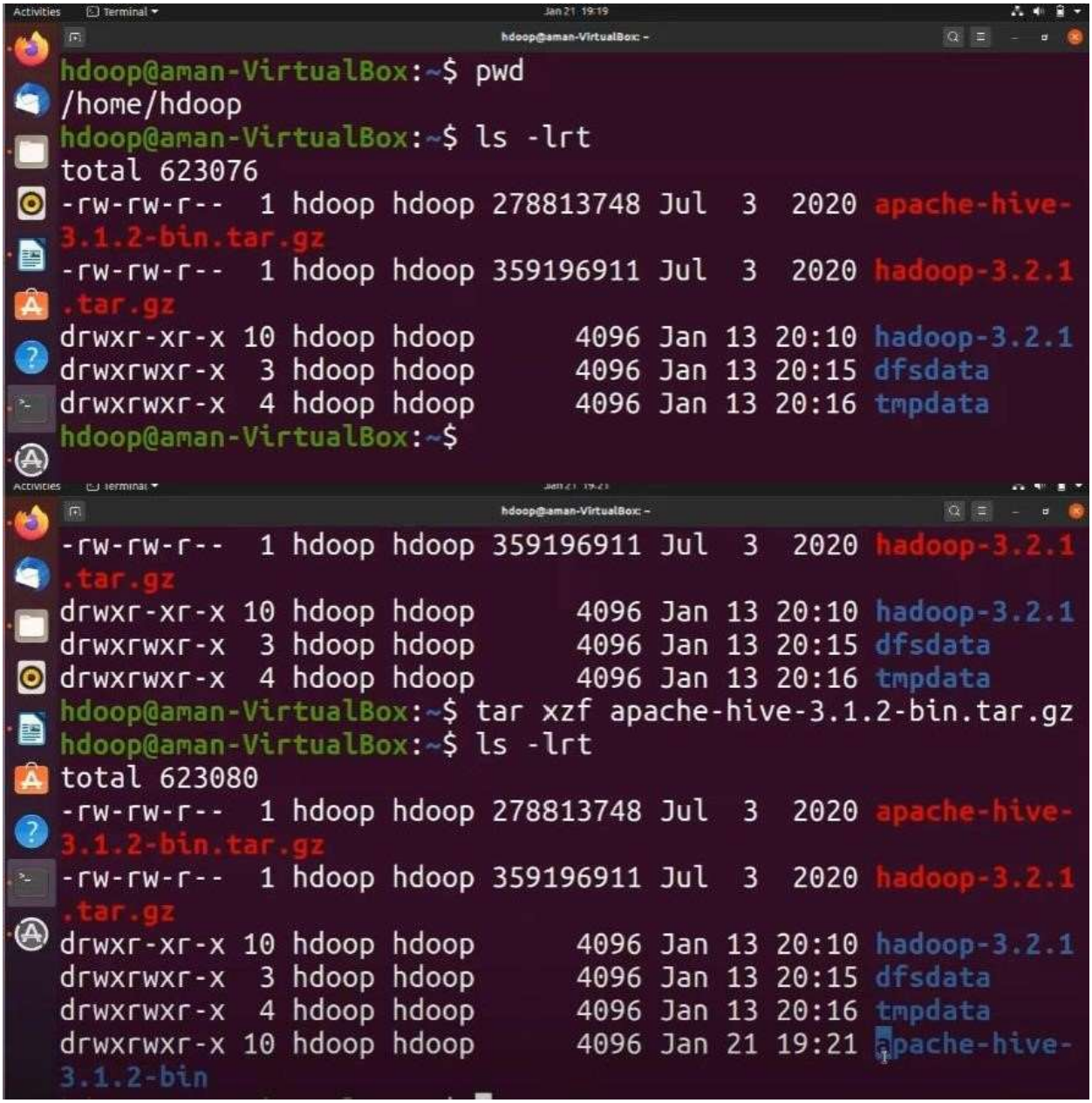
When to use Map reduce mode:

- If Hadoop is having multiple data nodes and data is distributed across different node we use Hive in this mode
- It will perform on large amount of data sets and query going to execute in parallel way Processing of large data sets with better performance can be achieved through this mode

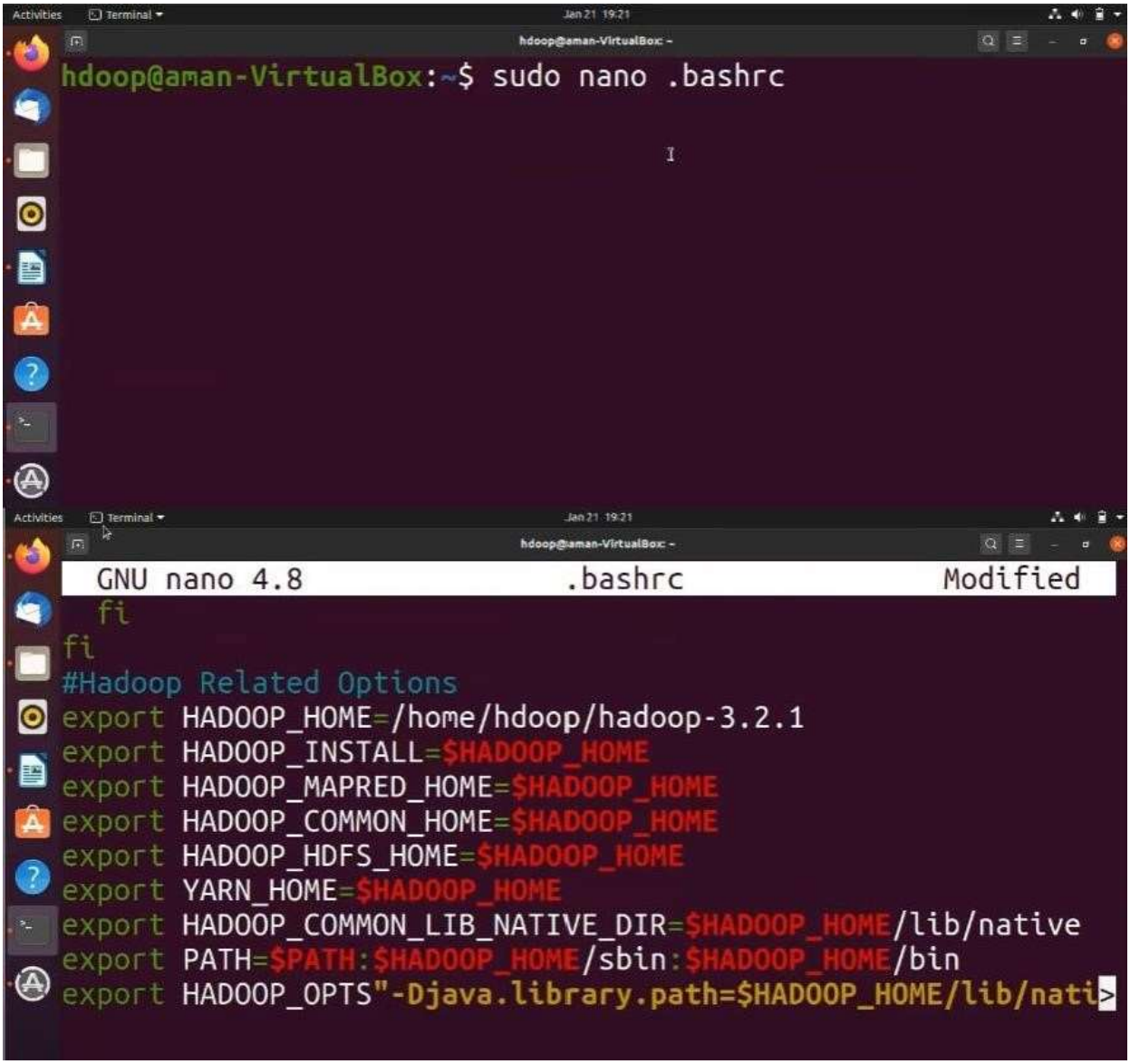
Lab Assignment:

1. Installation of HIVE.

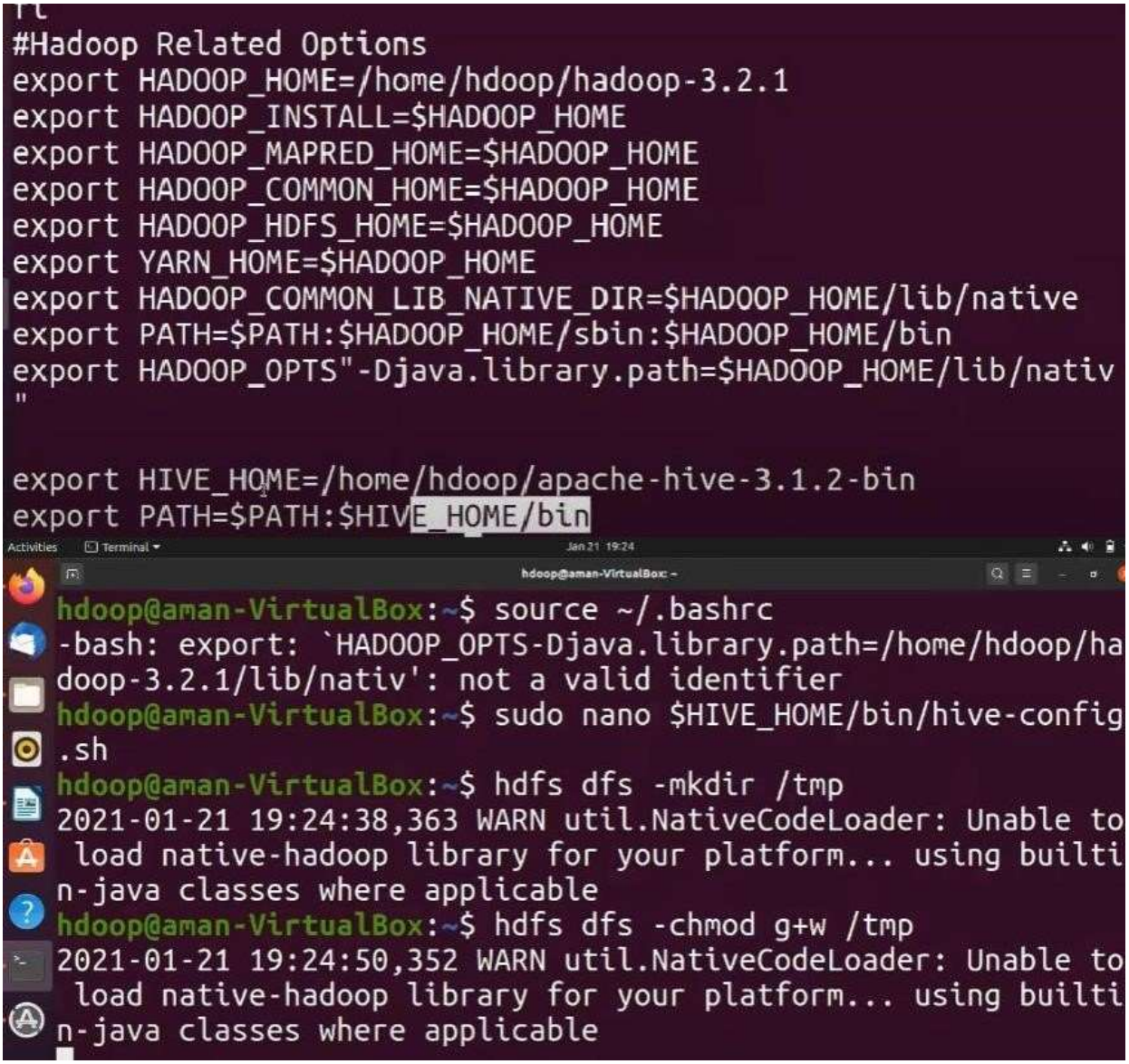
Department of Computer Science and Engineering (Data Science)

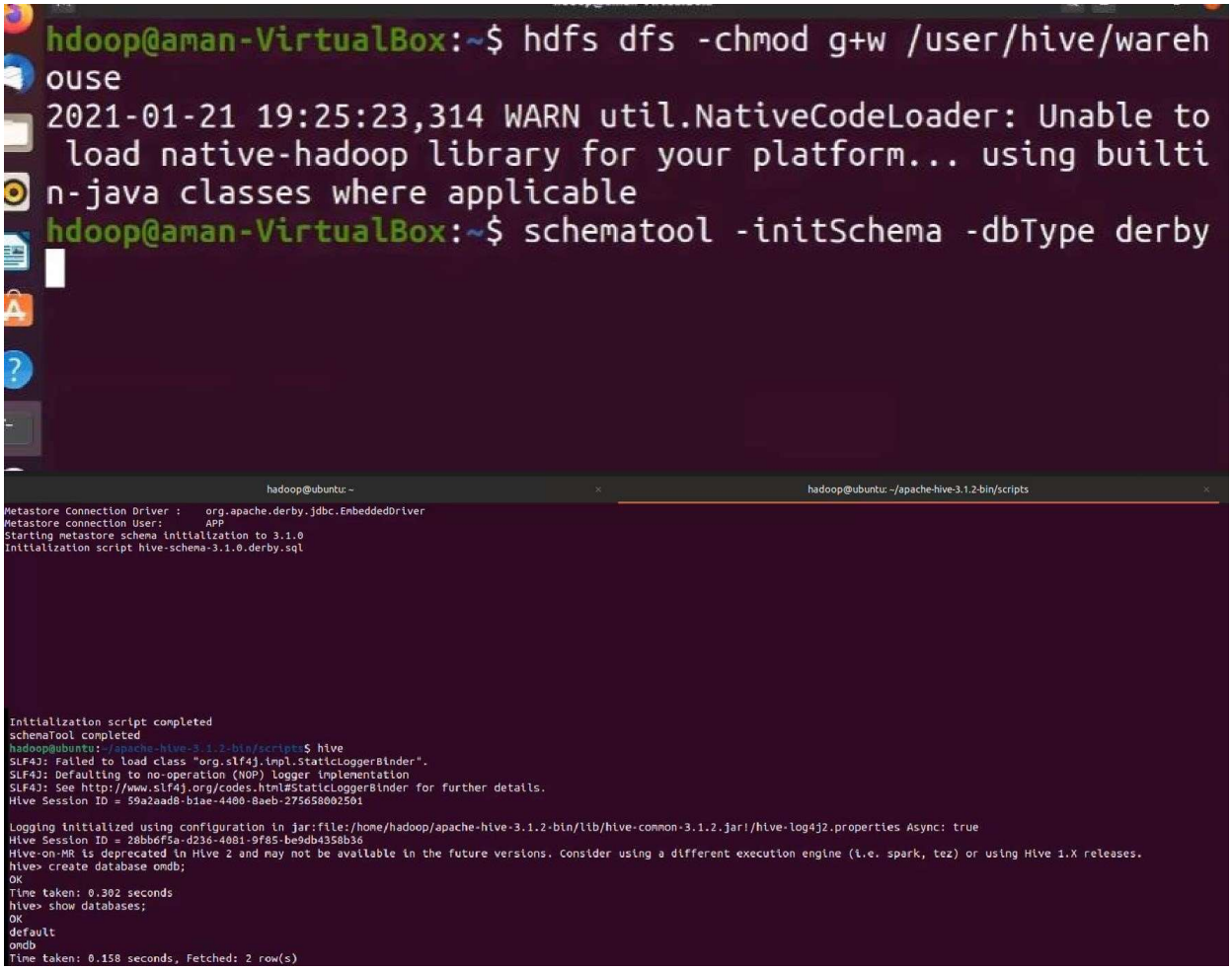


Department of Computer Science and Engineering (Data Science)



Department of Computer Science and Engineering (Data Science)





2. Implement the following SQL queries in HIVE on any database:
    a. Create Database
    b. Order by Query
    c. Group by Query
    d. Sort By
    e. Cluster By
    f. Distribute By

Department of Computer Science and Engineering (Data Science)

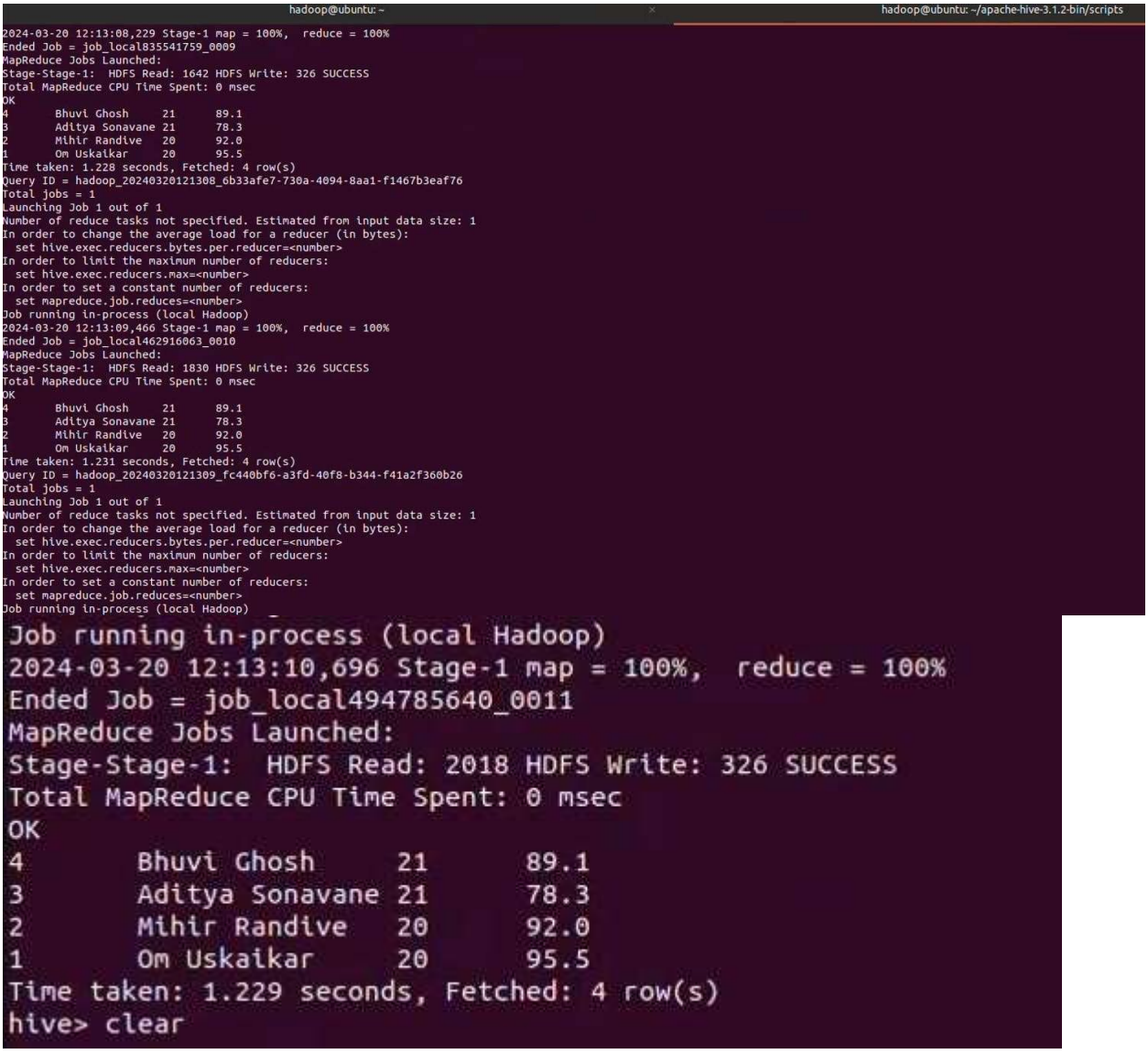
```
hive> use omdb;
OK
Time taken: 0.022 seconds
hive> CREATE TABLE IF NOT EXISTS students (
    >   student_id INT,
    >   student_name STRING,
    >   age INT,
    >   marks DOUBLE
    > )
    > ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
    > LINES TERMINATED BY '\n'
    > STORED AS textfile;
OK
Time taken: 0.597 seconds
hive> INSERT INTO students VALUES (1, 'Om Uskaikar', 20, 95.5),
    >                         (2, 'Mihir Randive', 20, 92.0),
    >                         (3, 'Aditya Sonavane', 21, 78.3),
    >                         (4, 'Bhuvi Ghosh', 21, 89.1);
Query ID = hadoop_20240320120955_9f0286ba-9a75-499f-ae0d-75be68d3790c
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)
2024-03-20 12:10:00,184 Stage-1 map = 0%,  reduce = 0%
2024-03-20 12:10:01,263 Stage-1 map = 100%,  reduce = 0%
2024-03-20 12:10:02,272 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local363716093_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/omdb.db/students/.hive-staging_hive_2024-03-20_12-09-55_972_6509308937409880813-1/-ext-10000
Loading data to table omdb.students
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 0 HDFS Write: 326 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
Time taken: 8.377 seconds
Moving data to directory hdfs://localhost:9000/user/hive/warehouse/omdb.db/students/.hive-staging_hive_2024-03-20_12-09-55_972_6509308937409880813-1/-ext-10000
Loading data to table omdb.students
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 0 HDFS Write: 326 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
Time taken: 8.377 seconds
hive> SELECT * FROM students ORDER BY marks DESC;
Query ID = hadoop_20240320121045_a2edbb2b-6d28-46d5-9aef-d4817dac7ed1
Total jobs = 1
Launching Job 1 out of 1
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)
2024-03-20 12:10:46,405 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local1212386302_0002
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 326 HDFS Write: 326 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
1       Om Uskaikar     20      95.5
2       Mihir Randive   20      92.0
4       Bhuvi Ghosh     21      89.1
3       Aditya Sonavane 21      78.3
Time taken: 1.278 seconds, Fetched: 4 row(s)
hive> SELECT age, COUNT(*) AS student_count FROM students GROUP BY age;  -- Groups by age and counts students
    >
    > SELECT age, COUNT(*) AS student_count FROM students GROUP BY age;
Query ID = hadoop_20240320121126_216688bc-85fb-46ac-bb4b-dc28d9353c5c
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 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)
```

Department of Computer Science and Engineering (Data Science)

```
Number of reduce tasks not specified. Estimated from input data size: 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)
2024-03-20 12:11:28,173 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local897478557_0003
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 514 HDFS Write: 326 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
20      2
21      2
Time taken: 1.379 seconds, Fetched: 2 row(s)
Query ID = hadoop_20240320121128_e48f7c87-8f07-4038-947e-d871b459635f
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 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)
2024-03-20 12:11:29,455 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local1118419812_0004
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 702 HDFS Write: 326 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
20      2
21      2
Time taken: 1.274 seconds, Fetched: 2 row(s)
hive> SELECT * FROM students CLUSTER BY age;  -- Distributes and sorts by age
    > SELECT * FROM students CLUSTER BY age;  -- Distributes and sorts by age
    >
    > SELECT * FROM students CLUSTER BY age;
Query ID = hadoop_20240320121210_5c556a46-2a0e-42bb-a9e7-be26a1a49298
Total jobs = 1
Launching Job 1 out of 1
```

```
hadoop@ubuntu: ~                                    hadoop@ubuntu: ~/apache-hive-3.1.2-bi
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 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)
2024-03-20 12:11:29,455 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local1118419812_0004
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 702 HDFS Write: 326 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
20      2
21      2
Time taken: 1.274 seconds, Fetched: 2 row(s)
hive> SELECT * FROM students CLUSTER BY age;  -- Distributes and sorts by age
    > SELECT * FROM students CLUSTER BY age;  -- Distributes and sorts by age
    >
    > SELECT * FROM students CLUSTER BY age;
Query ID = hadoop_20240320121210_5c556a46-2a0e-42bb-a9e7-be26a1a49298
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 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)
2024-03-20 12:12:11,987 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local138346866_0005
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 890 HDFS Write: 326 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
2       Mihir Randive   20      92.0
1       Om Uskaikar     20      95.5
4       Bhuvi Ghosh     21      89.1
3       Aditya Sonavane 21      78.3
Time taken: 1.259 seconds, Fetched: 4 row(s)
```

Department of Computer Science and Engineering (Data Science)



Department of Computer Science and Engineering (Data Science) Name :

```
hive> SELECT * FROM students ORDER BY marks DESC;
Query ID = hadoop_20240320121436_3b9dc475-d84d-4158-a41e-f674bf21dee1
Total jobs = 1
Launching Job 1 out of 1
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)
2024-03-20 12:14:37,435 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local1666408747_0012
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 2206 HDFS Write: 326 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
1       Om Uskaikar     20      95.5
2       Mihir Randive   20      92.0
4       Bhuvi Ghosh     21      89.1
3       Aditya Sonavane 21      78.3
Time taken: 1.277 seconds, Fetched: 4 row(s)
```

Department of Computer Science and Engineering (Data Science) Name :

```
2024-03-20 12:14:37,435 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local1666408747_0012
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 2206 HDFS Write: 326 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
1       Om Uskaikar     20      95.5
2       Mihir Randive   20      92.0
4       Bhuvi Ghosh     21      89.1
3       Aditya Sonavane 21      78.3
Time taken: 1.277 seconds, Fetched: 4 row(s)
hive> SELECT age, COUNT(*) AS student_count FROM students GROUP BY age;
Query ID = hadoop_20240320121446_142f1e85-c2f4-499b-a7a1-d93648cd3736
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 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)
2024-03-20 12:14:47,532 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local730891649_0013
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 2394 HDFS Write: 326 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
20      2
21      2
Time taken: 1.276 seconds, Fetched: 2 row(s)
hive> SELECT * FROM students CLUSTER BY age;
Query ID = hadoop_20240320121455_12f3f5ed-f782-4159-a4ff-9ab1d81ca416
Total jobs = 1
```

Department of Computer Science and Engineering (Data Science) Name :

```
hive> SELECT * FROM students CLUSTER BY age;
Query ID = hadoop_20240320121455_12f3f5ed-f782-4159-a4ff-9ab1d81ca416
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 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)
2024-03-20 12:14:56,442 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local753181872_0014
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 2582 HDFS Write: 326 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
2       Mihir Randive   20      92.0
1       Om Uskaikar     20      95.5
4       Bhuvi Ghosh     21      89.1
3       Aditya Sonavane 21      78.3
Time taken: 1.226 seconds, Fetched: 4 row(s)
hive> SELECT * FROM students DISTRIBUTE BY age;
Query ID = hadoop_20240320121503_8bfa3fbe-755e-4259-b903-c5f8288b38b6
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 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)
2024-03-20 12:15:04,243 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local832007827_0015
```

```
hive> SELECT * FROM students CLUSTER BY age;
Query ID = hadoop_20240320121455_12f3f5ed-f782-4159-a4ff-9ab1d81ca416
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 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)
2024-03-20 12:14:56,442 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local753181872_0014
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 2582 HDFS Write: 326 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
2       Mihir Randive   20      92.0
1       Om Uskaikar     20      95.5
4       Bhuvi Ghosh     21      89.1
3       Aditya Sonavane 21      78.3
Time taken: 1.226 seconds, Fetched: 4 row(s)
hive> SELECT * FROM students DISTRIBUTE BY age;
Query ID = hadoop_20240320121503_8bfa3fbe-755e-4259-b903-c5f8288b38b6
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 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)
2024-03-20 12:15:04,243 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local832007827_0015
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 2770 HDFS Write: 326 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
4       Bhuvi Ghosh     21      89.1
3       Aditya Sonavane 21      78.3
2       Mihir Randive   20      92.0
1       Om Uskaikar     20      95.5
Time taken: 1.148 seconds, Fetched: 4 row(s)
hive>
```

Working with HIVE ETL:

g. Structured Data using Hive.
h. Semi structured data using Hive (XML, JSON).

```
hive> create table json1(str string);
OK
Time taken: 0.2 seconds
hive> LOAD DATA LOCAL INPATH '/home/hadoop/employee.json' INTO TABLE json1;
Loading data to table default.json1
OK
Time taken: 0.853 seconds
hive> select * from json1;
OK
{"Id": 1, "Name": "OM", "Age": 30, "Address": "123 Main St", "Salary": 50000.0, "Department": "IT"},
{"Id": 2, "Name": "Mihir", "Age": 35, "Address": "456 Elm St", "Salary": 60000.0, "Department": "HR"},
{"Id": 3, "Name": "Bhuvi", "Age": 40, "Address": "789 Oak St", "Salary": 70000.0, "Department": "Finance"},
{"Id": 4, "Name": "Vishma", "Age": 25, "Address": "567 Pine St", "Salary": 55000.0, "Department": "Marketing"},
{"Id": 5, "Name": "Atharv", "Age": 28, "Address": "890 Maple St", "Salary": 52000.0, "Department": "Sales"},
{"Id": 6, "Name": "Yash", "Age": 32, "Address": "901 Cedar St", "Salary": 58000.0, "Department": "IT"},
{"Id": 7, "Name": "Hiya", "Age": 29, "Address": "234 Oak St", "Salary": 54000.0, "Department": "Finance"},
{"Id": 8, "Name": "Anuradha", "Age": 37, "Address": "345 Pine St", "Salary": 62000.0, "Department": "HR"}
Time taken: 0.084 seconds, Fetched: 8 row(s)
hive> SELECT get_json_object(str, '$.Id') AS Id, get_json_object(str, '$.Name') AS Name,get_json_object(str, '$.Age') AS Age,get_json_object(str, '$.Address') AS Address,get_json_object(str, '$.Salary')
AS Salary,get_json_object(str, '$.Department') AS Department FROM json;
FAILED: SemanticException [Error 10004]: Line 1:23 Invalid table alias or column reference 'str': (possible column names are: json)
hive> SELECT get_json_object(str, '$.Id') AS Id, get_json_object(str, '$.Name') AS Name,get_json_object(str, '$.Age') AS Age,get_json_object(str, '$.Address') AS Address,get_json_object(str, '$.Salary')
AS Salary,get_json_object(str, '$.Department') AS Department FROM json1;
OK
1       OM      30      123 Main St     50000.0 IT
2       Mihir   35      456 Elm St      60000.0 HR
3       Bhuvi   40      789 Oak St      70000.0 Finance
4       Vishma  25      567 Pine St     55000.0 Marketing
5       Atharv  28      890 Maple St    52000.0 Sales
6       Yash    32      901 Cedar St    58000.0 IT
7       Hiya    29      234 Oak St      54000.0 Finance
8       Anuradha        37      345 Pine St     62000.0 HR
Time taken: 0.08 seconds, Fetched: 8 row(s)
hive>
```

```
Time taken: 0.171 seconds
hive> show tables;
OK
Time taken: 0.041 seconds
hive> create table employee(str string);
OK
Time taken: 0.298 seconds
hive> LOAD DATA LOCAL INPATH '/home/hadoop/test.xml' INTO TABLE employees;
FAILED: SemanticException [Error 10001]: Line 1:58 Table not found 'employees'
hive> LOAD DATA LOCAL INPATH '/home/hadoop/test.xml' INTO TABLE employee;
Loading data to table default.employee
OK
Time taken: 0.374 seconds
hive> select xpath(str, '/emp/esal/text()'), xpath(str, '/emp/ename/text()') from employee;
OK
["340000"]      ["Om"]
["520000"]      ["Mihir"]
["440000"]      ["Bhuvi"]
["420000"]      ["Yash"]
["520000"]      ["Varun"]
["350000"]      ["Sanotsh"]
["388000"]      ["Sravani"]
["420000"]      ["Promod"]
Time taken: 1.213 seconds, Fetched: 8 row(s)
hive> select * from employee
    > select * from employee;
FAILED: ParseException line 2:0 missing EOF at 'select' near 'employee'
hive> select * from employee;
OK
<emp><ename>Om</ename><esal>340000</esal></emp>
<emp><ename>Mihir</ename><esal>520000</esal></emp>
<emp><ename>Bhuvi</ename><esal>440000</esal></emp>
<emp><ename>Yash</ename><esal>420000</esal></emp>
<emp><ename>Varun</ename><esal>520000</esal></emp>
<emp><ename>Sanotsh</ename><esal>350000</esal></emp>
<emp><ename>Sravani</ename><esal>388000</esal></emp>
<emp><ename>Promod</ename><esal>420000</esal></emp>
Time taken: 0.103 seconds, Fetched: 8 row(s)
hive>
Time taken: 0.081 seconds
hive> create table json(json string);
OK
Time taken: 0.051 seconds
hive> LOAD DATA LOCAL INPATH '/home/hadoop/employee.json' INTO TABLE json;
Loading data to table default.json
OK
Time taken: 0.125 seconds
hive> select * from json;
OK
{"Id": 1, "Name": "OM", "Age": 30, "Address": "123 Main St", "Salary": 50000.0, "Department": "IT"},
{"Id": 2, "Name": "Mihir", "Age": 35, "Address": "456 Elm St", "Salary": 60000.0, "Department": "HR"},
{"Id": 3, "Name": "Bhuvi", "Age": 40, "Address": "789 Oak St", "Salary": 70000.0, "Department": "Finance"},
{"Id": 4, "Name": "Vishma", "Age": 25, "Address": "567 Pine St", "Salary": 55000.0, "Department": "Marketing"},
{"Id": 5, "Name": "Atharv", "Age": 28, "Address": "890 Maple St", "Salary": 52000.0, "Department": "Sales"},
{"Id": 6, "Name": "Yash", "Age": 32, "Address": "901 Cedar St", "Salary": 58000.0, "Department": "IT"},
{"Id": 7, "Name": "Hiya", "Age": 29, "Address": "234 Oak St", "Salary": 54000.0, "Department": "Finance"},
{"Id": 8, "Name": "Anuradha", "Age": 37, "Address": "345 Pine St", "Salary": 62000.0, "Department": "HR"}
Time taken: 0.085 seconds, Fetched: 8 row(s)
hive> SELECT
    >     get_json_object(json, '$.Id') AS Id,
    >     get_json_object(json, '$.Name') AS Name,
    >     get_json_object(json, '$.Age') AS Age,
    >     get_json_object(json, '$.Address') AS Address,
    >     get_json_object(json, '$.Salary') AS Salary,
    >     get_json_object(json, '$.Department') AS Department
    > FROM json;
OK
1       OM      30      123 Main St     50000.0 IT
2       Mihir   35      456 Elm St      60000.0 HR
3       Bhuvi   40      789 Oak St      70000.0 Finance
4       Vishma  25      567 Pine St     55000.0 Marketing
5       Atharv  28      890 Maple St    52000.0 Sales
6       Yash    32      901 Cedar St    58000.0 IT
7       Hiya    29      234 Oak St      54000.0 Finance
8       Anuradha        37      345 Pine St     62000.0 HR
Time taken: 0.092 seconds, Fetched: 8 row(s)
hive>
```