Complete Beginner Guide to Hadoop, Hive, and MapReduce (with Full Steps, Comments, and Theory)

---

1. Basic Definitions:

What is Hadoop?

- Hadoop is an open-source framework.
- It stores large datasets across many machines (HDFS).
- It processes data using MapReduce (parallel processing).

What is Hive?

- Hive is a Data Warehouse tool on top of Hadoop.
- Allows you to query Big Data using SQL-like language (HiveQL).
- Converts queries internally into MapReduce jobs.

What is MapReduce?

- A programming model in Hadoop.
- Processes data in two phases: Map (filter/sort) and Reduce (aggregate results).

---

2. Basic Requirements Before You Start:

- Java must be installed.
- Hadoop must be installed and running (HDFS, YARN).

- Hive must be installed.

- Linux OS (or use Virtual Machine like Cloudera Sandbox).

---

3. Step-by-Step Guide to Run Hive Queries

Step 1: Start Hadoop Services

start-dfs.sh

start-yarn.sh

Step 2: Start Hive

hive

---

4. Upload Data to HDFS

Create directory in HDFS:

hdfs dfs -mkdir /user/flightdata

Put your CSV file into HDFS:

hdfs dfs -put flightinfo.csv /user/flightdata/

Comment: HDFS is Hadoop's file system. Hive reads from HDFS.

---

5. Create Hive Database and Table

Create a database:

```
CREATE DATABASE flights;

USE flights;
```

Comment: Database helps organize multiple tables.

Create External Table:

```
CREATE EXTERNAL TABLE flight_info (
    Year INT,
    Month INT,
    DayofMonth INT,
    FlightNum INT,
    DepartureDelay INT
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
STORED AS TEXTFILE
LOCATION '/user/flightdata/';
```

Comment: External Table: Data stays outside Hive in HDFS.

---

6. Load Data into Managed Table

```
LOAD DATA INPATH '/user/flightdata/flightinfo.csv' INTO TABLE flight_info;
```

Comment: Managed tables store data inside Hive warehouse directory.

---

7. Insert New Values into Table

```
INSERT INTO TABLE flight_info VALUES ('AI-123', 'DEL', 'MUM', '2008-01-01', 15, 5);
```

Comment: Adding a new flight record manually.

---

8. Create and Insert into Another Table

```
CREATE TABLE airport_info (
    Code STRING,
    City STRING,
    State STRING
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
STORED AS TEXTFILE;

INSERT INTO TABLE airport_info VALUES ('DEL', 'New Delhi', 'DL');
INSERT INTO TABLE airport_info VALUES ('MUM', 'Mumbai', 'MH');
```

---

9. Perform Join Operation

```
SELECT fi.FlightNum, ap.City AS OriginCity, fi.DepartureDelay
FROM flight_info fi
JOIN airport_info ap
ON fi.Origin = ap.Code;
```

Comment: Joining flight and airport tables.

---

10. Create and Rebuild Index

```
CREATE INDEX flight_idx
ON TABLE flight_info (FlightNum)
AS 'org.apache.hadoop.hive.ql.index.compact.CompactIndexHandler'
WITH DEFERRED REBUILD;

ALTER INDEX flight_idx ON flight_info REBUILD;
```

Comment: Indexing improves performance.

---

11. Find Average Departure Delay

```
SELECT FlightDate, AVG(DepartureDelay) AS Avg_Departure_Delay
FROM flight_info
WHERE SUBSTR(FlightDate, 1, 4) = '2008'
GROUP BY FlightDate
ORDER BY FlightDate;
```

---

12. Drop Table

DROP TABLE flight_info;

DROP TABLE airport_info;

Comment: Dropping a table.

Important Reminders:

- Hadoop + Hive must be installed properly.

- Hadoop must be running before starting Hive.

- Hive queries are slower than normal SQL because of MapReduce.

- Always end Hive queries with ';'.

- HDFS stores data across multiple nodes (distributed).

- External tables do not delete data on DROP.

- Managed tables delete data on DROP.