9. Create partitioned table using Static partitioning technique load the data into corresponding partitions.

To create a partitioned table using static partitioning in Hive and load data into corresponding partitions, follow these steps:

**Step 1: Prepare the Data**

Assume we have a CSV file data.csv with the following content:

1,John,2000,2023

2,Jane,3000,2023

3,Bob,1500,2022

4,Alice,2500,2022

5,Tom,1200,2021

The dataset represents records with the following fields: id, name, salary, and year.

### Step 2: Create the Partitioned Table

1. **Start Hive**:

hive

**Create the Partitioned Table**:

CREATE TABLE employee\_partitioned (

id INT,

name STRING,

salary DOUBLE

)

PARTITIONED BY (year INT)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

STORED AS TEXTFILE;

**Step 3: Load Data into Corresponding Partitions**

Since we're using static partitioning, we need to load data into each partition separately. First, we should split the data.csv file into separate files for each partition.

1. **Split Data into Separate Files**:
   * data\_2023.csv:

1,John,2000

2,Jane,3000

data\_2022.csv:

3,Bob,1500

4,Alice,2500

data\_2021.csv:

5,Tom,1200

**Load Data into Partitions**:

-- Load data for the year 2023

LOAD DATA LOCAL INPATH '/path/to/data\_2023.csv' INTO TABLE employee\_partitioned PARTITION (year=2023);

-- Load data for the year 2022

LOAD DATA LOCAL INPATH '/path/to/data\_2022.csv' INTO TABLE employee\_partitioned PARTITION (year=2022);

-- Load data for the year 2021

LOAD DATA LOCAL INPATH '/path/to/data\_2021.csv' INTO TABLE employee\_partitioned PARTITION (year=2021);

**Step 4: Verify the Data**

1. **Run Queries to Verify Data**:

-- Select all records for the year 2023

SELECT \* FROM employee\_partitioned WHERE year=2023;

-- Select all records for the year 2022

SELECT \* FROM employee\_partitioned WHERE year=2022;

-- Select all records for the year 2021

SELECT \* FROM employee\_partitioned WHERE year=2021;

-- Select all records from the partitioned table

SELECT \* FROM employee\_partitioned;

### Complete Commands

Here's the complete sequence of commands to run in the Hive terminal and the shell:

**Hive Terminal:**

-- Create the partitioned table

CREATE TABLE employee\_partitioned (

id INT,

name STRING,

salary DOUBLE

)

PARTITIONED BY (year INT)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

STORED AS TEXTFILE;

-- Load data into corresponding partitions

LOAD DATA LOCAL INPATH '/path/to/data\_2023.csv' INTO TABLE employee\_partitioned PARTITION (year=2023);

LOAD DATA LOCAL INPATH '/path/to/data\_2022.csv' INTO TABLE employee\_partitioned PARTITION (year=2022);

LOAD DATA LOCAL INPATH '/path/to/data\_2021.csv' INTO TABLE employee\_partitioned PARTITION (year=2021);

-- Run queries to verify data

SELECT \* FROM employee\_partitioned WHERE year=2023;

SELECT \* FROM employee\_partitioned WHERE year=2022;

SELECT \* FROM employee\_partitioned WHERE year=2021;

SELECT \* FROM employee\_partitioned;

Replace /path/to/data\_2023.csv, /path/to/data\_2022.csv, and /path/to/data\_2021.csv with the actual paths to your split CSV files. This sequence will create a partitioned table, load data into the respective partitions, and verify the data through queries.