1.

In terminal.

$vi learning.txt.

$pwd

$ mkdir hadoop

$cp /home/cloudera/ learning. Txt/ homel/cloudera/Hadoop/

$mv /home/cloudera/learning.txt /home /cloudera /hadoop.

$ cd Hadoop/

$ chmod 764 learning.txt

$ll

$ chmod 664 learning.txt.

$ll

$ chmod 717 learning txt.

$ll

$cd..

New terminal

$hadoop fs-ls

# Step 1: Create directories in HDFS

hadoop fs -mkdir /learning.1

hadoop fs -mkdir /learning.2

# Step 2: Copy a file from local to HDFS

hadoop fs -copyFromLocal /home/cloudera/Hadoop/learning.txt /learning.1/

# Step 3: List files in both HDFS directories

hadoop dfs -ls /learning.1

hadoop dfs -ls /learning.2

# Step 4: Copy file from HDFS to local

hadoop dfs -get /user/cloudera/learning.1/\*

# Step 5: Copy all user files from HDFS to local hadoop folder

hadoop dfs -copyToLocal /user/cloudera/\* /home/cloudera/hadoop/

# Step 6: Change to local directory

cd Hadoop

ll

hdfs dfs -moveFrom Local /home/cloudera/enterprise --/deployment.json /user/clousera/

hdfs dfs -ls

2.  
use collection;

db.createCollection(“students”)

db.students.insertOne({name:”aaaa”,age:21,})

db.students.find()

db.students.drop()

3.

In terminal hive

CREATE DATABASE learning;

USE learning;

New terminal:

mkdir hadoop.learning

cd hadoop.learning

in hive terminal:

# Create employee data files using any editor

vi emp\_data\_2025.csv

vi emp\_data\_2024.csv

CREATE TABLE emp\_static (

id INT,

name STRING

)

PARTITIONED BY (year STRING)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ',';

CREATE TABLE emp\_stg (

id INT,

name STRING,

year STRING

)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ',';

LOAD DATA LOCAL INPATH '/home/cloudera/hadoop.learning/emp\_data\_2025.csv'

INTO TABLE emp\_stg;

-- Repeat for other years:

LOAD DATA LOCAL INPATH '/home/cloudera/hadoop.learning/emp\_data\_2024.csv'

INTO TABLE emp\_stg;

INSERT INTO TABLE emp\_static PARTITION (year='2023')

SELECT id, name FROM emp\_stg WHERE year = '2023';

INSERT INTO TABLE emp\_static PARTITION (year='2020')

SELECT id, name FROM emp\_stg WHERE year = '2020';

INSERT INTO TABLE emp\_static PARTITION (year='2021')

SELECT id, name FROM emp\_stg WHERE year = '2021';

SHOW CREATE TABLE emp\_static;

Copy location

In hdfs terminal:

Hadoop dfs-ls (paste location)

In hive terminal:

Enable dynamic partitioning

SET hive.exec.dynamic.partition;

SET hive.exec.dynamic.partition

SET hive.exec.dynamic.partition.mode = nonstrict;

INSERT INTO TABLE emp\_dynamic PARTITION (year)

SELECT id, name, year FROM emp\_stg;

SHOW PARTITIONS emp\_dynamic;

SHOW CREATE TABLE emp\_dynamic;(copy location)

In hdfs terminal:

hadoop fs -ls paste location

4.

Use fruit;

db.createCollection(“food”)

db.food.insertOne({

... name: "Basket 1",

... fruit: ["apple", "banana", "mango"]

... })

db.food.createIndex({ fruit: 1 })

db.food.find({ fruit: "banana" })

db.food.updateOne(

... { name: "Basket 1" },

... { $push: { fruit: "kiwi" } }

... )

db.food.updateOne(

... { name: "Basket 1" },

... { $pop: { fruit: 1 } } // 1 = remove last, -1 = remove first

... )

db.food.updateOne(

... { name: "Basket 2", fruit: "banana" },

... { $set: { "fruit.$": "papaya" } }

... )

db.dropDatabase()

5.terminal in hive

New terminal

$ mkdir learning

$ cd learning

# Create your CSV file (use nano/vi/editor of choice)

$ vi emp-data.csv

$pwd

# Create HDFS directory

$ hadoop fs -mkdir /hadoop-learning

# Copy the file to HDFS

$ hadoop fs -copyFromLocal /home/cloudera/learning/emp-data.csv /hadoop-learning/

$ hadoop fs -ls /hadoop-learning

In hive terminal:

SHOW DATABASES;

CREATE DATABASE learning;

USE learning;

SET hive.cli.print.current.db=true;

CREATE TABLE emp\_test (

id INT,

name STRING,

location STRING

)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ',';

LOAD DATA LOCAL INPATH '/home/cloudera/learning/emp-data.csv'

INTO TABLE emp\_test;

LOAD DATA INPATH '/hadoop-learning/emp-data.csv'

INTO TABLE emp\_test;

DESCRIBE emp\_test;

SELECT \* FROM emp\_test;

-- Aggregate Function Example

SELECT COUNT(\*) FROM emp\_test;

-- Projection + Filtering

SELECT name, location FROM emp\_test WHERE id > 101;

SHOW CREATE TABLE emp\_test;(copy location)

In hdfs   
**# View file in HDFS**

**$ hadoop fs -cat /hadoop-learning/emp-data.csv**

**Paste location** DROP TABLE emp\_test;

6.

use collection

db.orders.insertMany([

... { orderId: 1, customer: "mm", fruit: "apple", quantity: 5, price: 10 },

... { orderId: 2, customer: "Sara", fruit: "banana", quantity: 3, price: 5 },

... { orderId: 3, customer: "yy", fruit: "mango", quantity: 2, price: 15 },

... { orderId: 4, customer: "Ayaan", fruit: "grapes", quantity: 4, price: 8 },

... { orderId: 5, customer: "Sara", fruit: "apple", quantity: 1, price: 10 },

... { orderId: 6, customer: "Ayaan", fruit: "banana", quantity: 6, price: 5 }

... ])

db.orders.countDocuments()

db.orders.countDocuments({ fruit: "apple" })

db.orders.find().limit(3)

db.orders.find().sort({ price: -1 }).skip(2).limit(2)

... { $group: { \_id: "$customer", totalQuantity: { $sum: "$quantity" } } }

... ])

7.

In hive  
hive

SHOW DATABASES;

CREATE DATABASE learning;

USE learning;

CREATE TABLE emp\_stg (

id INT,

name STRING,

dept STRING,

hire\_date STRING

)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ',';

LOAD DATA LOCAL INPATH '/home/cloudera/learning/emp-data.csv'

INTO TABLE emp\_stg;

CREATE TABLE employee\_bucket (

id INT,

name STRING,

dept STRING,

hire\_date STRING

)

CLUSTERED BY (id)

INTO 4 BUCKETS

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ',';

New terminal:

SET hive.enforce.bucketing = true;

INSERT INTO TABLE employee\_bucket

SELECT id, name, dept, hire\_date FROM emp\_stg

SORT BY id;

SHOW CREATE TABLE employee\_bucket;

SELECT \* FROM employee\_bucket;

CREATE TABLE emp\_part\_bucket (

id INT,

name STRING,

dept STRING,

hire\_date STRING

)

PARTITIONED BY (year STRING)

CLUSTERED BY (id)

INTO 4 BUCKETS

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ',';

SET hive.exec.dynamic.partition = true;

SET hive.exec.dynamic.partition.mode = nonstrict;

SET hive.enforce.bucketing = true;

INSERT INTO TABLE emp\_part\_bucket PARTITION (year)

SELECT

id, name, dept, hire\_date,

SUBSTR(hire\_date, 1, 4) AS year

FROM emp\_stg

SORT BY id;

SHOW PARTITIONS emp\_part\_bucket;

SHOW CREATE TABLE emp\_part\_bucket;

SELECT \* FROM emp\_part\_bucket;

# New terminal

hadoop fs -ls /user/hive/warehouse/learning.db/emp\_part\_bucket

hadoop fs -ls /user/hive/warehouse/learning.db/emp\_part\_bucket/year=2024

hadoop fs -cat /user/hive/warehouse/learning.db/emp\_part\_bucket/year=2025/000000\_0