📁 Sample Data Files (Place in `/home/cloudera/data/`)

csv

CopyEdit

```
# customer_info.csv

101,John,5001

102,Alice,5002

103,Bob,5003


# order_info.csv

5001,I01,2

5002,I02,1

5003,I03,5


# item_info.csv

I01,Laptop,70000

I02,Mouse,500

I03,Keyboard,1500
```

✅ a. Creating Tables in Hive

sql

CopyEdit

```
CREATE DATABASE IF NOT EXISTS customer_db;

USE customer_db;


CREATE TABLE customer_info (

    cust_id INT,

    cust_name STRING,

    order_id INT

)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

STORED AS TEXTFILE;


CREATE TABLE order_info (

    order_id INT,

    item_id STRING,

    quantity INT

)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

STORED AS TEXTFILE;
```

```
CREATE TABLE item_info (
    item_id STRING,
    item_name STRING,
    item_price FLOAT
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
STORED AS TEXTFILE;
```

✅ b. Load Data into Hive Tables

sql

CopyEdit

```
LOAD DATA LOCAL INPATH '/home/cloudera/data/customer_info.csv'
INTO TABLE customer_info;

LOAD DATA LOCAL INPATH '/home/cloudera/data/order_info.csv' INTO
TABLE order_info;

LOAD DATA LOCAL INPATH '/home/cloudera/data/item_info.csv' INTO
TABLE item_info;
```

✅ c. Perform JOIN on the Tables

sql

CopyEdit

```
SELECT
    c.cust_id,
    c.cust_name,
    o.item_id,
    i.item_name,
    i.item_price,
    o.quantity,
    (i.item_price * o.quantity) AS total_price
FROM customer_info c
JOIN order_info o ON c.order_id = o.order_id
JOIN item_info i ON o.item_id = i.item_id;
```

✅ d. Create Index on Customer Info Table

sql

CopyEdit

```
CREATE INDEX idx_cust_name
ON TABLE customer_info (cust_name)
```

```
AS 'org.apache.hadoop.hive.ql.index.compact.CompactIndexHandler'
WITH DEFERRED REBUILD;

ALTER INDEX idx_cust_name ON customer_info REBUILD;
```

✅ e. Find Total, Average Sales

sql

CopyEdit

```
SELECT
    SUM(i.item_price * o.quantity) AS total_sales,
    AVG(i.item_price * o.quantity) AS avg_sales
FROM order_info o
JOIN item_info i ON o.item_id = i.item_id;
```

✅ f. Find Order Details with Maximum Cost

sql

CopyEdit

```
SELECT
    o.order_id,
```

```
    i.item_id,
    i.item_name,
    o.quantity,
    (i.item_price * o.quantity) AS total_price
FROM order_info o
JOIN item_info i ON o.item_id = i.item_id
ORDER BY total_price DESC
LIMIT 1;
```

✅ g. Creating an External Hive Table to Connect to HBase

sql

CopyEdit

```
CREATE EXTERNAL TABLE hbase_customer_info (
    key STRING,
    cust_id STRING,
    cust_name STRING,
    order_id STRING
)
STORED BY 'org.apache.hadoop.hive.hbase.HBaseStorageHandler'
WITH SERDEPROPERTIES (
```

```
    "hbase.columns.mapping" =
":key,info:cust_id,info:cust_name,info:order_id"

)

TBLPROPERTIES (

    "hbase.table.name" = "hbase_customer_info"

);
```

You must also create the table in HBase before using it:

bash

CopyEdit

```
hbase shell


create 'hbase_customer_info', 'info'


put 'hbase_customer_info', '101', 'info:cust_id', '101'

put 'hbase_customer_info', '101', 'info:cust_name', 'John'

put 'hbase_customer_info', '101', 'info:order_id', '5001'


put 'hbase_customer_info', '102', 'info:cust_id', '102'

put 'hbase_customer_info', '102', 'info:cust_name', 'Alice'

put 'hbase_customer_info', '102', 'info:order_id', '5002'
```

✅ h. Display HBase Table Data from Hive

sql

CopyEdit

```
SELECT * FROM hbase_customer_info;
```