All solutions with pics:-

Joins:-

1. Create table customer_table:-

```
hive> create table customer_table
    > (
    > id int,
    > name string,
    > age string,
    > address string,
    > salary int
    > )
    > row format delimited
    > Fields terminated by ',';
OK
Time taken: 0.165 seconds
```

2. Create order_table:-

```
hive> create table order_table
    > (
    > oid string,
    > date string,
    > customer_id int,
    > amount int
    > )
    > row format delimited
    > Fields terminated by ',';
OK
Time taken: 0.085 seconds
hive>
```

Loading data into tables:-

```
hive> load data local inpath '/tmp/hive_challange/customer_data.csv' overwrite into table customer_table;

Loading data to table hive_challange.customer_table
Table hive_challange.customer_table stats: [numFiles=1, numRows=0, totalSize=152, rawDataSize=0]
OK
Time taken: 0.649 seconds
hive> load data local inpath '/tmp/hive_challange/order_data.csv' overwrite into table order_table;
Loading data to table hive_challange.order_table
Table hive_challange.order_table stats: [numFiles=1, numRows=0, totalSize=134, rawDataSize=0]
OK
Time taken: 0.408 seconds
hive>
```

Left Join:-

```
hive> select c_t.id,c_t.name,o_t.oid,o_t.amount
    > from customer_table as c_t left join order_table as o_t
    > on c_t.id = o_t.customer_id
    > order by c_t.name;
```

```
101     'Aadil' 'A15'   500
102     'Sania' 'A16'   600
103     'Iliyas'        'A17'   700
104     'Sam'   'A18'   800
105     'Radheshyam'    'A19'   900
Time taken: 23.037 seconds, Fetched: 5 row(s)
```

Right Join:-

```
hive> select c_t.id,c_t.name,o_t.oid,o_t.amount
    > from customer_table as c_t right join order_table as o_t
    > on c_t.id = o_t.customer_id
    > order by c_t.name;
```

```
101     'Aadil' 'A15'   500
103     'Iliyas'        'A17'   700
105     'Radheshyam'    'A19'   900
104     'Sam'   'A18'   800
102     'Sania' 'A16'   600
```

Inner join:-

```
hive> select c_t.*,o_t.* from customer_table as c_t
    > inner join order_table as o_t
    > on c_t.id = o_t.customer_id;
```

```
101     'Aadil' 23      'Sendhwa'       24000   'A15'   '15-07-2022'    101     500
102     'Sania' 18      'Indore'        0       'A16'   '16-07-2022'    102     600
103     'Iliyas'        29      'Mumbai'        30000   'A17'   17-07-2022'     103     700
104     'Sam'   23      'Dubai' 50000   'A18'   18-07-2022'     104     800
105     'Radheshyam'    28      'Varanasi'      80000   'A19'   '19-07-2022'    105     900
```

Outer join:-

```
hive> select c_t.id,c_t.name,o_t.oid,o_t.amount
    > from customer_table as c_t full outer join order_table as o_t
    > on c_t.id = o_t.customer_id
    > order by o_t.amount;
```

```
101     'Aadil' 'A15'   500
102     'Sania' 'A16'   600
103     'Iliyas'        'A17'   700
104     'Sam'   'A18'   800
105     'Radheshyam'    'A19'   900
```

Machine-Learning Database:-

Create a hive table as per given schema in your dataset:-

```
hive> create table if not exists AirQualityindex

    > (

    > Date date,

    > Time string,

    > CO_GT float,
    > PT08_S1_CO int,
    > NMHC_GT int,
    > C6H6_GT float,
    > PT08_S2_NMHC int,
    > NOx_GT int,
    > PT08_S3_NOx int,
    > NO2_GT int,
    > PT08_S4_NO2 int,
    > PT08_S5_O3 int,
    > T float,
    > RH float,
    > AH float
    > )

    > row format delimited

    > fields terminated by ','
    > TBLPROPERTIES ("skip.header.line.count"="1");
```

try to place a data into table location

```
hive> LOAD DATA INPATH '/tmp/hive_challange_hdfs/AirQualityUCI.csv' INTO table AirQualityindex;
```

Perform a select operation:-

```
select max(T) from AirQualityindex;
```

```
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 3.
Total MapReduce CPU Time Spent: 3 seconds 650 msec
OK
44.6
Time taken: 20.821 seconds, Fetched: 1 row(s)
hive>
```

Perform group by operation .

```
select DATE_FORMAT(to_date(from_unixtime(unix_timestamp(Date , 'DD/MM/YYYY'))),'MMM-YYYY'), max(PT08_S3_NOx) from AirQualityindex group by DATE_FORMAT(to_date(from_unixtime(unix_timestamp(Date , 'DD/MM/YYYY'))),'MMM-YYYY');
```

```
OK
Dec-2004        2683
Dec-2005        1881
Time taken: 19.373 seconds, Fetched: 2 row(s)
```

Perform alter operation:-

```
alter table AirQualityindex change date Date string;
```

```
alter table AirQualityindex change t Tt string;
```

Perform filter operation at least 5 kinds of filter examples .

1.

```
select count(CO_GT) from AirQualityindex where PT08_S3_NOx< 700;
```

```
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU:
Total MapReduce CPU Time Spent: 3 seconds 520 msec
OK
3200
```

2.

```
select count(C6H6_GT) from AirQualityindex where C6H6_GT >11.0;
```

```
Total MapReduce CPU Time Spent: 3 seconds 570 msec
OK
3198
```

3.

```
select count(NOx_GT) from AirQualityindex where DATE_FORMAT(to_date(from_unixtime(unix_timestamp(Date , 'DD/MM/YYYY'))),'YYYY') = '2004';
```

```
Total MapReduce CPU Time Spent: 4 seconds 670 msec
OK
7110
Time taken: 19.206 seconds, Fetched: 1 row(s)
hive>
```

4.

```
select *  from AirQualityindex where CO_GT like  '2%'  limit 2;
```

```
OK
3/10/2004       18:00:00        2.6     1360    150     11.9    1046    166     1056    113     1692    1
268     13.6    48.9    0.7578
3/10/2004       19:00:00        2.0     1292    112     9.4     955     103     1174    92      1559    9
72      13.3    47.7    0.7255
Time taken: 0.079 seconds, Fetched: 2 row(s)
hive>
```

5.

```
select CO_GT from AirQualityindex where Tt = 13.6 order by CO_GT Desc;
```

```
3.0
2.8
2.8
2.8
2.6
2.6
2.3
2.2
2.1
1.9
1.9
1.7
1.5
1.4
1.4
1.3
1.1
0.9
0.8
0.8
0.8
0.8
0.7
0.6
0.6
0.5
-200.0
-200.0
Time taken: 17.52 seconds, Fetched: 37 row(s)
hive>
```

show and example of regex operation

```
select count(Tt) from AirQualityindex where Tt REGEXP '^-'
```

```
OK
379
Time taken: 17.022 seconds, Fetched: 1 row(s)
hive>
```

order by operation .

```
select C6H6_GT from AirQualityindex where Tt = 11 order by C6G6_GT Desc;
```

```
cloudera@quickstart:~
15.7
14.3
13.8
13.3
13.1
13.0
13.0
12.9
12.8
9.2
8.5
8.2
8.1
6.8
5.4
5.1
4.1
4.0
3.9
3.8
2.9
2.8
2.5
2.2
1.4
1.2
1.1
0.4
Time taken: 19.16 seconds, Fetched: 35 row(s)
hive>
```

Where statement

```
select count(CO_GT) from AirQualityindex where PT08_S3_nox< 700
```

```
OK
3200
Time taken: 18.526 seconds, Fetched: 1 row(s)
hive>
```

Sort by statement

```
select C6H6_GT from AirQualityindex where Tt = 11 sort by co Desc;
```

```
cloudera@quickstart:~
15.7
14.3
13.8
13.3
13.1
13.0
13.0
12.9
12.8
9.2
8.5
8.2
8.1
6.8
5.4
5.1
4.1
4.0
3.9
3.8
2.9
2.8
2.5
2.2
1.4
1.2
1.1
0.4
Time taken: 18.145 seconds, Fetched: 35 row(s)
hive>
```

count statement

```
select count(Tt) from AirQualityindex;
```

```
OK
9357
Time taken: 18.145 seconds, Fetched: 1 row(s)
hive>
```

distinct operation you have to perform .

```
select Distinct Tt from AirQualityindex where CO_GT like '3%' limit 10;
```

```
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 3.8
Total MapReduce CPU Time Spent: 3 seconds 860 msec
OK
-200
0.9
1.3
1.4
1.8
10.0
10.1
10.2
10.3
10.4
Time taken: 18.574 seconds, Fetched: 10 row(s)
hive>
```

like an operation you have to perform .

select Distinct Tt from AirQualityindex where CO_GT like '4%' limit 10;

```
Total MapReduce CPU Time Spent: 3 seconds 690
OK
-200
10.0
10.1
10.2
10.3
10.4
10.5
10.6
10.7
10.8
```

Union:-

select count(NMHC_GT) from AirQualityindex where RH>40 and RH<60 union

all

select count(CO_GT) from AirQualityindex where AH>1;

```
Stage-Stage-2: Map: 2   Cumulative CPU: 2.87 sec   HDFS Read: 60
Total MapReduce CPU Time Spent: 10 seconds 70 msec
OK
3485
4447
Time taken: 52.018 seconds, Fetched: 2 row(s)
hive>
```

View:-

create view aadilview as select DATE_FORMAT(to_date(from_unixtime(unix_timestamp(Date , 'DD/MM/YYYY'))),'YYYY'), count(C6H6_GT) from AirQualityindex group by DATE_FORMAT(to_date(from_unixtime(unix_timestamp(Date , 'DD/MM/YYYY'))),'YYYY');

select* from aadilview;

```
Total MapReduce CPU Time Spent: 4 seconds 840 msec
OK
2004    7110
2005    2247
Time taken: 21.666 seconds, Fetched: 2 row(s)
hive>
```

Store raw data into hdfs location:-

hdfs dfs -copyFromLocal /tmp/hive_challange/sales_order_data /tmp/hive_challange_hdfs/;

```
Create a internal hive table "sales_order_csv" which will store csv data sales_order_csv ..
make sure to skip header row while creating table
```

create table sales_order

(

ORDERNUMBER int,

QUANTITYORDERED int,

PRICEEACH float,

ORDERLINENUMBER int,

SALES float,

SATUS string,

QTR_ID int,

MONTH_ID int,

YEAR_ID int,

PRODUCTLINE string,

MSRP int,

PRODUCTCODE string,

PHONE string,

CITY string,

STATE string,

```
POSTALCODE string,
COUNTRY string,
TERRITORY string,
CONTACTLASTNAME string,
CONTACTFIRSTNAME string,
DEALSIZE string
)
row format delimited
fields terminated by ','
TBLPROPERTIES ("skip.header.line.count"="1");
```

```
Load data from hdfs path into "sales_order_csv"
LOAD DATA INPATH '/tmp/hive_challange_hdfs/sales_order_data.csv' INTO table sales_order;
```

```
Create an internal hive table which will store data in ORC format "sales_order_orc"
create table sales_order_orc
(
ORDERNUMBER int,
QUANTITYORDERED int,
PRICEEACH float,
ORDERLINENUMBER int,
SALES float,
SATUS string,
QTR_ID int,
MONTH_ID int,
YEAR_ID int,
PRODUCTLINE string,
MSRP int,
PRODUCTCODE string,
PHONE string,
CITY string,
```

STATE string,

POSTALCODE string,

COUNTRY string,

TERRITORY string,

CONTACTLASTNAME string,

CONTACTFIRSTNAME string,

DEALSIZE string

)

row format delimited

fields terminated by ','

STORED AS ORC

TBLPROPERTIES ("skip.header.line.count"="1");

```
Load data from "sales_order_csv" into "sales_order_orc"
```

INSERT INTO TABLE sales_order_orc SELECT * FROM sales_order;

```
a. Calculatye total sales per year
```

select sum(sales) as total_sales, year_id from sales_order_orc group by year;

```
total_sales      year_id
3514108.547241211        2003
4724162.593383789        2004
1791486.7086791992       2005
Time taken: 17.42 seconds, Fetched: 3 row(s)
hive>
```

b. find a product for which maximum orders were placed;

```
select productline, count(quantityordered) from sales_order_orc group by productline limit
1;
```

```
OK
productline      _c1
Classic Cars     967
Time taken: 18.027 seconds, Fetched: 1 row(s)
hive>
```

c. Calculate the total sales for each quarter

```
select sum(sales) as total_sales_per_year ,qtr_id from sales_order_orc group by qtr_id
order by total_sales_per_year;
```

```
OK
total_sales_per_year    qtr_id
1758910.808959961       3
2048120.3029174805      2
2347946.726501465       1
3874780.010925293       4
Time taken: 37.076 seconds, Fetched: 4 row(s)
hive>
```

d. In which quarter sales was minimum

```
select sum(sales) as total_sales ,year_id  ,qtr_id from sales_order_orc group by
qtr_id,year_id order by total_sales limit 1;
```

```
OK
total_sales     year_id qtr_id
442223.6897583008       2003    1
Time taken: 38.097 seconds, Fetched: 1 row(s)
hive>
```

```
In which country sales was maximum and in which country sales was minimum
```

select country, max(sales) as max_min from sales_order_orc group by country order by max_min desc limit 1

union all

select country,min(sales) as max_min from sales_order_orc group by country order by max_min asc limit 1;

```
France  482.13
USA     14082.8
Time taken: 87.14 seconds, Fetched: 2 row(s)
hive>
```

Calculate quarterly sales for each city

```
select qtr_id,city,sum(sales) from sales_order_orc group by qtr_id,city;
```

```
cloudera@quickstart:~
Total MapReduce CPU Time Spent: 3 seconds 280 msec
OK
qtr_id  city     _c2
1       Bergamo 56181.320068359375
1       Boras   31606.72021484375
1       Brickhaven      31474.7802734375
1       Brisbane        16118.479858398438
1       Bruxelles       18800.089721679688
1       Burbank 37850.07958984375
1       Burlingame      13529.570190429688
1       Cambridge       21782.699951171875
1       Charleroi       16628.16015625
1       Cowes   26906.68017578125
1       Dublin  38784.470458984375
1       Espoo   51373.49072265625
1       Frankfurt       48698.82922363281
1       Gensve  50432.549560546875
1       Glendale        3987.199951171875
1       Graz    8775.159912109375
1       Helsinki        26422.819458007812
1       Kobenhavn       58871.110107421875
1       Lille   20178.1298828125
1       London  8477.219970703125
1       Los Angeles     23889.320068359375
1       Lule    9748.999755859375
1       Lyon    101339.13977050781
1       Madrid  357668.4899291992
1       Makati City     55245.02014160156
1       Manchester      51017.919860839844
1       Marseille       2317.43994140625
```

```
Find a month for each year in which maximum number of quantities were sold
```

All year wise:-

```
select sum(sales) as total_sales_per_year ,qtr_id from sales_order_orc group by qtr_id;
```

```
total_sales_per_year    qtr_id
2347946.726501465       1
2048120.3029174805      2
1758910.808959961       3
3874780.010925293       4
Time taken: 17.015 seconds, Fetched: 4 row(s)
```

Or

Per Year wise

```
select sum(sales),year_id  ,qtr_id from sales_order_orc group by qtr_id,year_id;
```

```
OK
_c0      year_id qtr_id
442223.6897583008       2003    1
833730.6786499023       2004    1
1071992.3580932617      2005    1
562365.2218017578       2003    2
766260.7305297852       2004    2
719494.3505859375       2005    2
649514.5415039062       2003    3
1109396.2674560547      2004    3
1860005.094177246       2003    4
2014774.9167480469      2004    4
Time taken: 17.994 seconds, Fetched: 10 row(s)
hive>
```