**Assignment -03**

Customers and Transaction

----------------------------------------------------------------------------------

* CREATE TABLE IF NOT EXISTS customers

(custid int, last\_name String, first\_name String, age int, profession String)

COMMENT 'customers details'

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

LINES TERMINATED BY '\n'

STORED AS TEXTFILE;

* LOAD DATA LOCAL INPATH '/mnt/c/MKM/SQL/repository/futurense\_hadoop-pyspark/labs/dataset/retail/ customers.txt' OVERWRITE INTO TABLE customers;
* CREATE TABLE IF NOT EXISTS transactions

(trans\_id int, trans\_date String, cust\_id int, amount double, category String, desc String, city String, state String, pymt\_mode String)

COMMENT 'transactions details'

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

LINES TERMINATED BY '\n'

STORED AS TEXTFILE;

* LOAD DATA LOCAL INPATH '/mnt/c/MKM/SQL/repository/futurense\_hadoop-pyspark/labs/dataset/retail/ transactions.txt' OVERWRITE INTO TABLE transactions;

1) No of transactions by customer

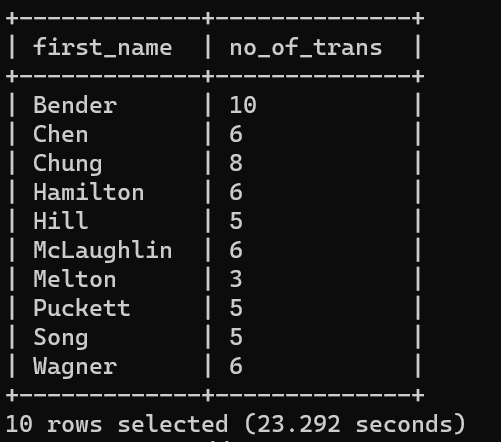
select first\_name, count(\*) as no\_of\_trans

from customers c inner join transactions t

ON c.custid = t.cust\_id

group by c.first\_name

;



2) Total transaction amount by customer

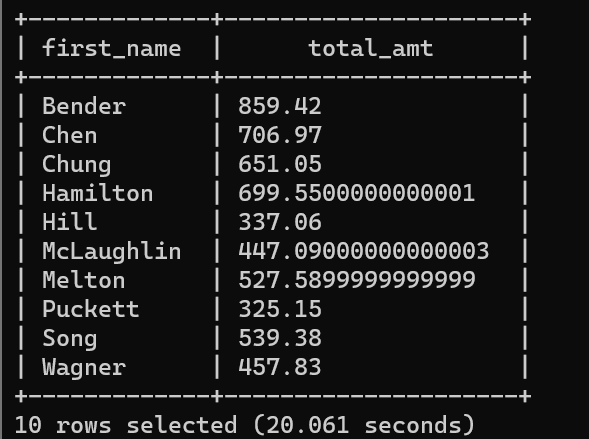
select first\_name, sum(amount) as total\_amt

from customers c inner join transactions t

ON c.custid = t.cust\_id

group by c.first\_name

;



3) Get top 3 customers by transaction amount

select first\_name, sum(amount) as total\_amt

from customers c inner join transactions t

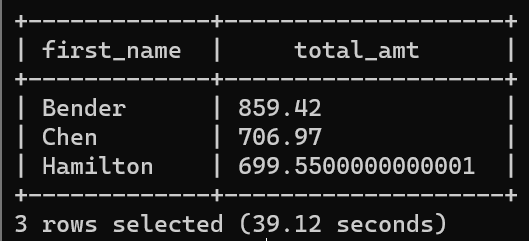
ON c.custid = t.cust\_id

group by c.first\_name

order by sum(amount) desc

limit 3

;



4) No of transactions by customer and mode of payment

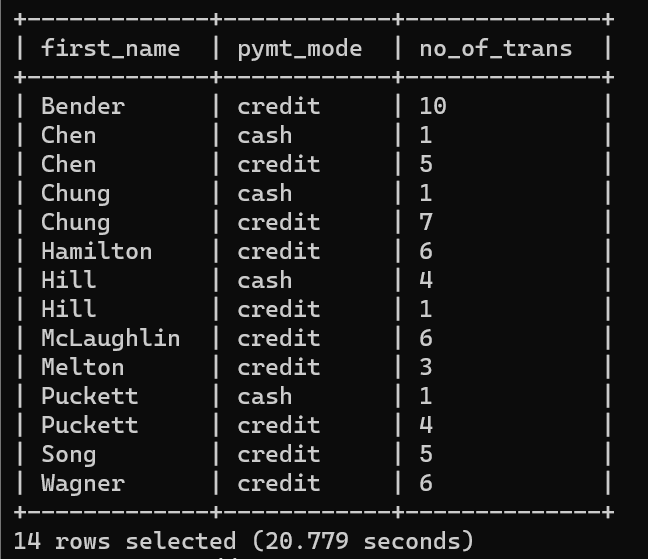
select first\_name, pymt\_mode, count(\*) as no\_of\_trans

from customers c inner join transactions t

ON c.custid = t.cust\_id

group by first\_name, pymt\_mode

;



5) Get top 3 cities which has more transactions

select city, count(\*) as no\_of\_trans

from customers c inner join transactions t

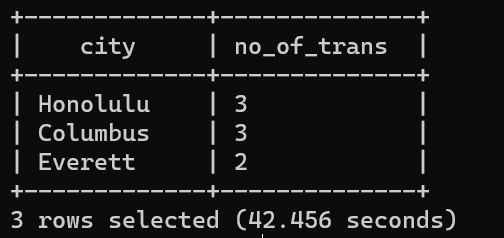
ON c.custid = t.cust\_id

group by city

order by count(\*) desc

limit 3

;



6) Get month wise highest transaction

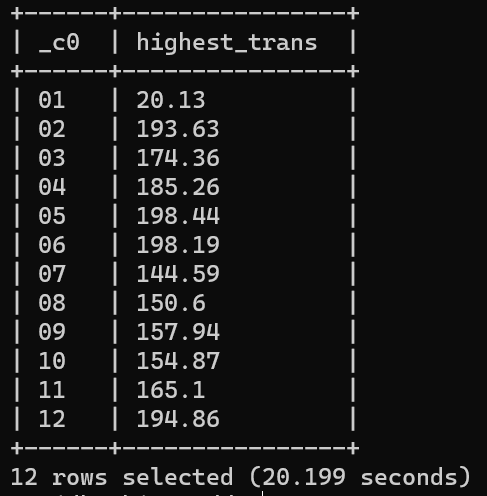
select substring(trans\_date, 0, 2), max(amount) as highest\_trans

from customers c inner join transactions t

ON c.custid = t.cust\_id

group by substring(trans\_date, 0, 2)

;



7) Get sample transactions