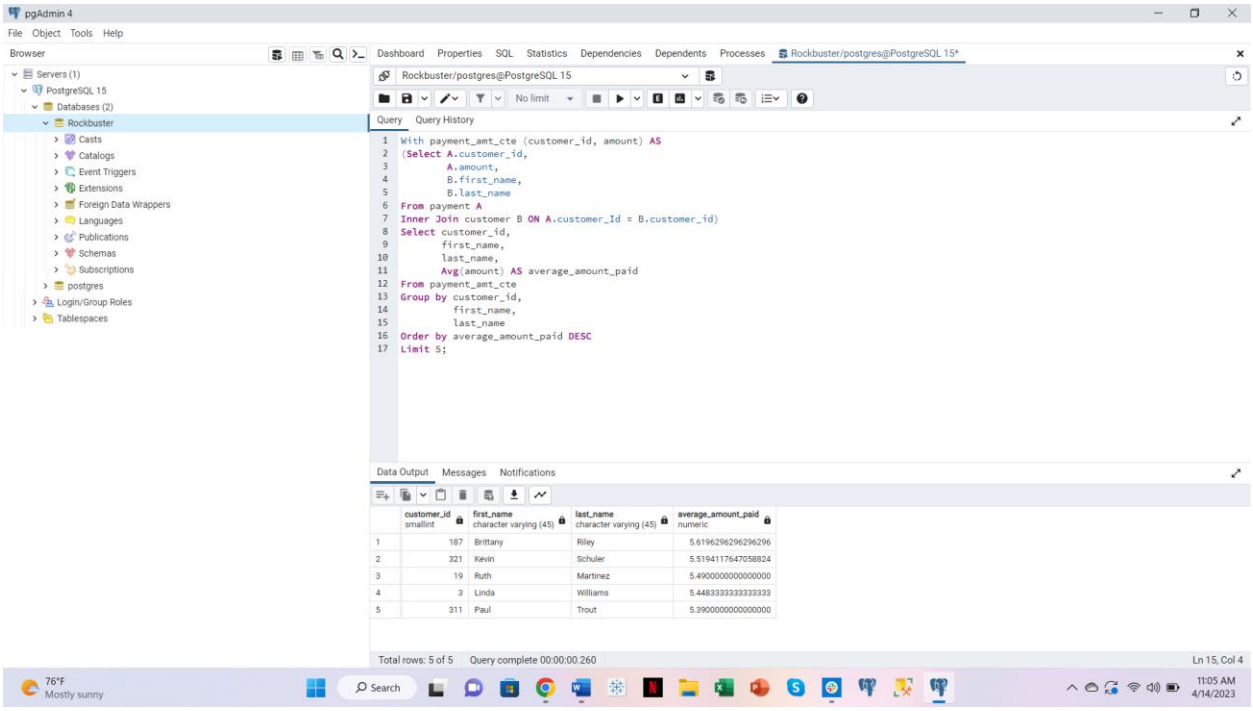
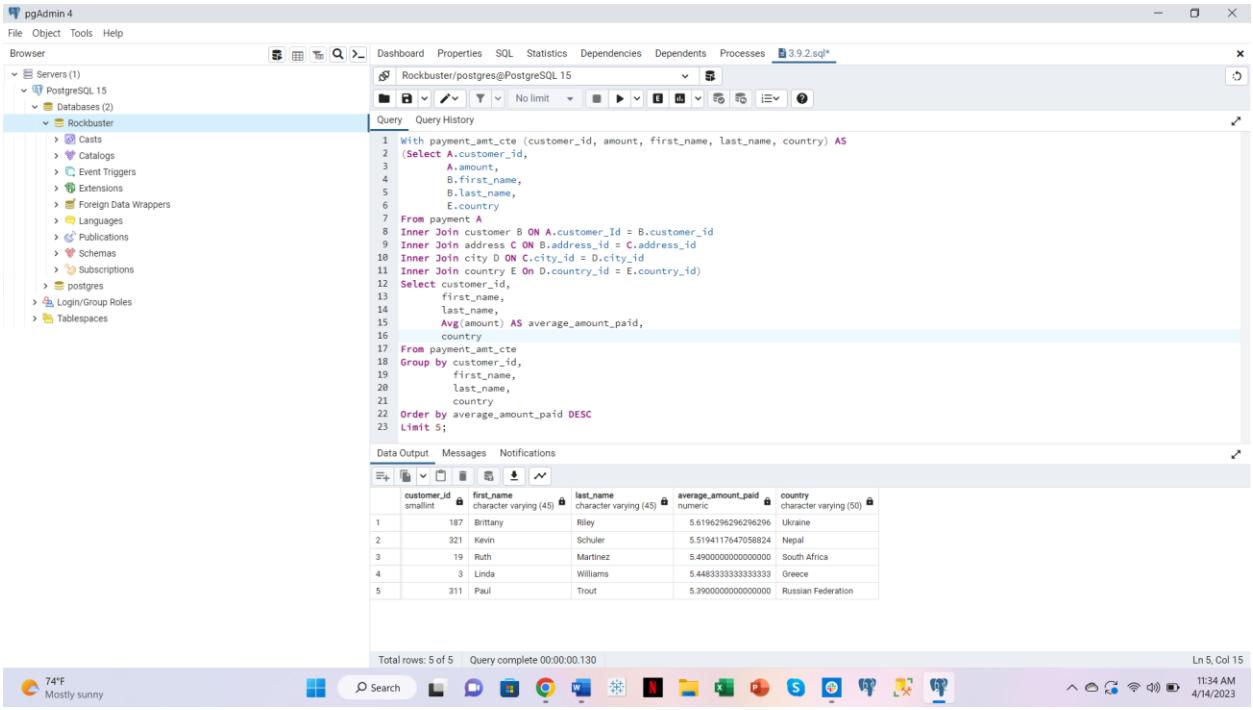
Exercise 3.9

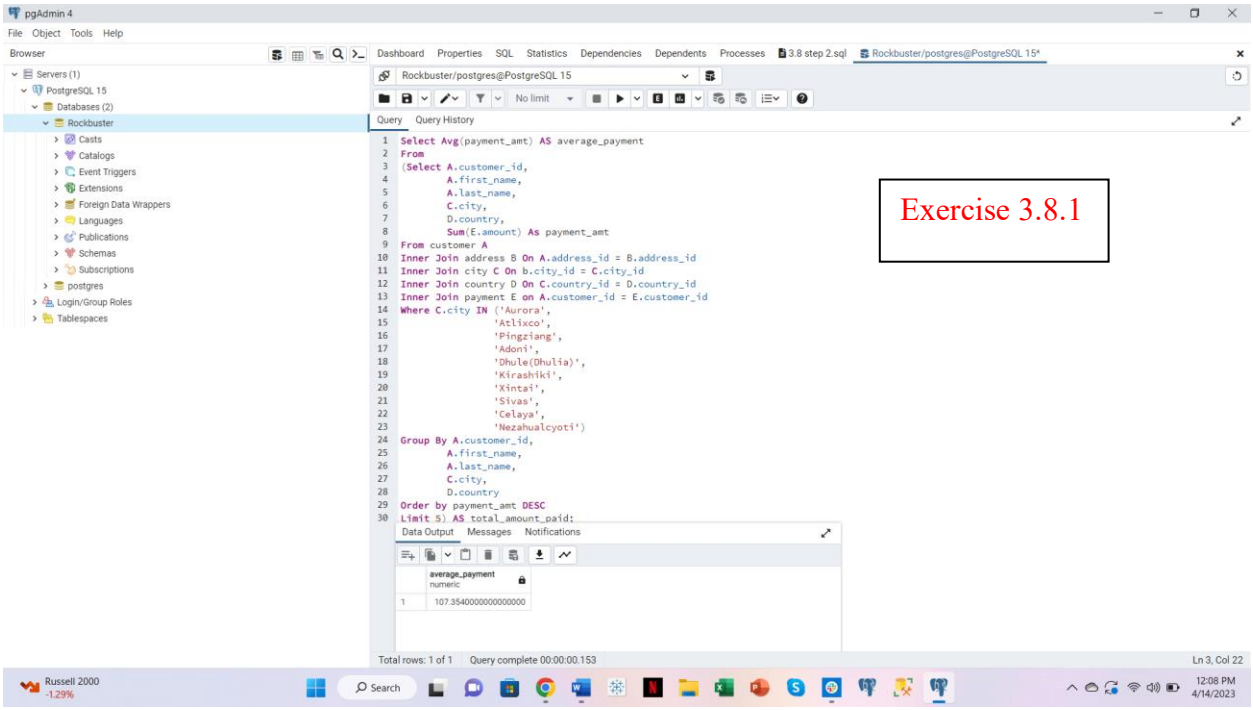
Chrisie Hunter

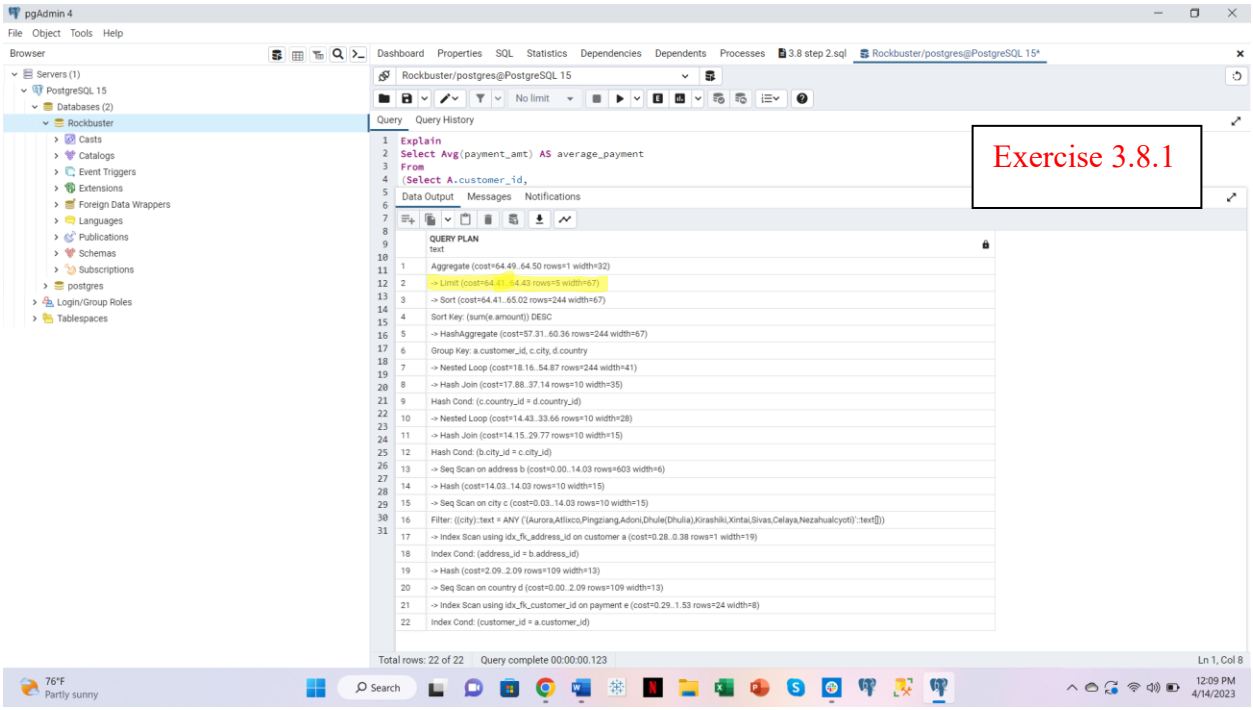
Step 1: Find the average amount paid by the top 5 customers.

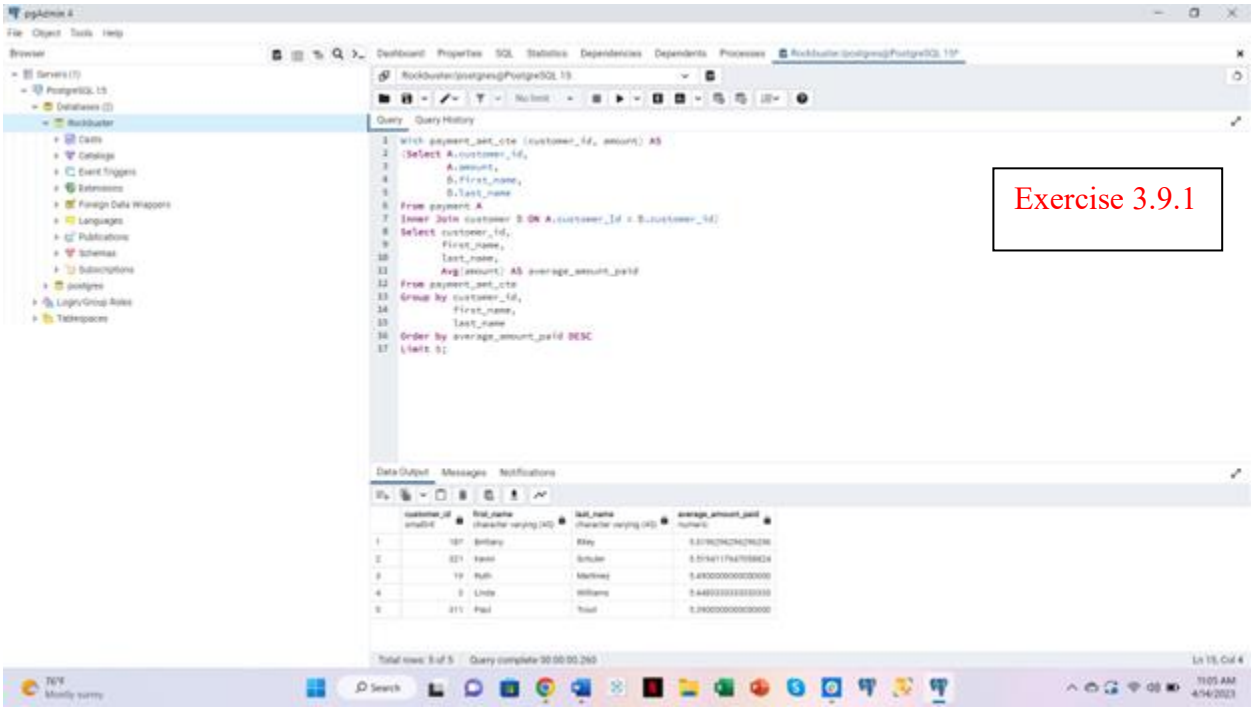


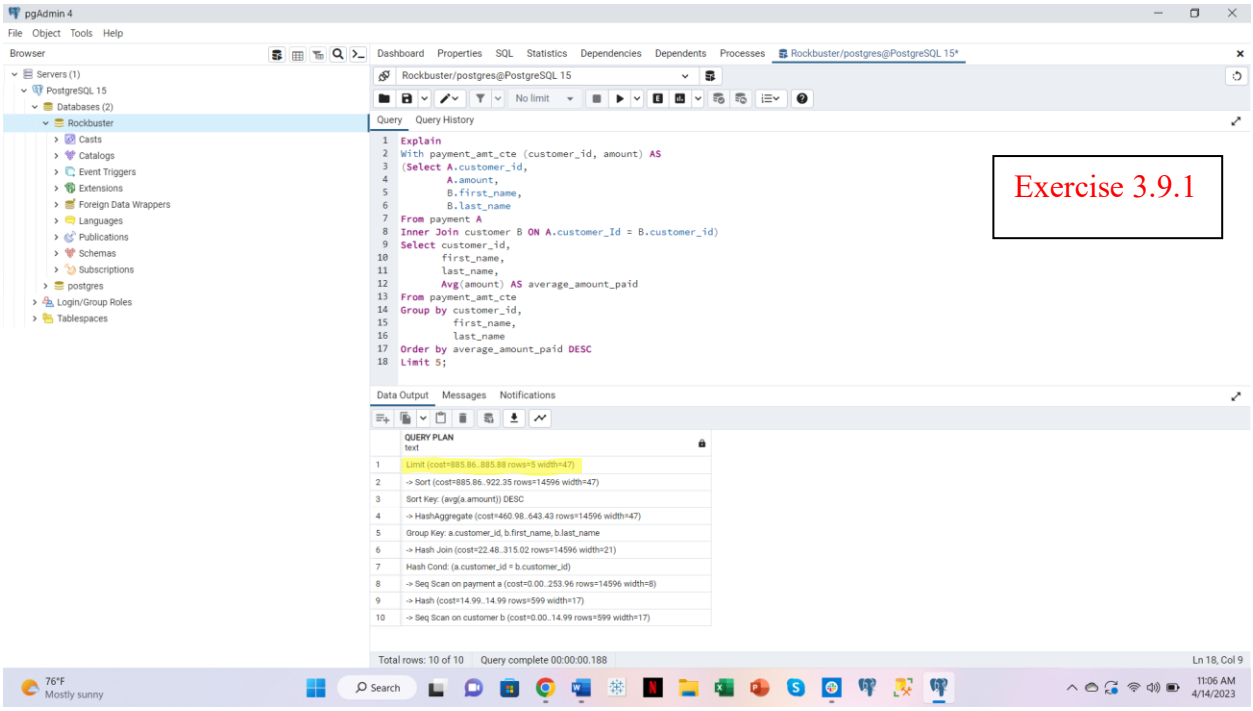


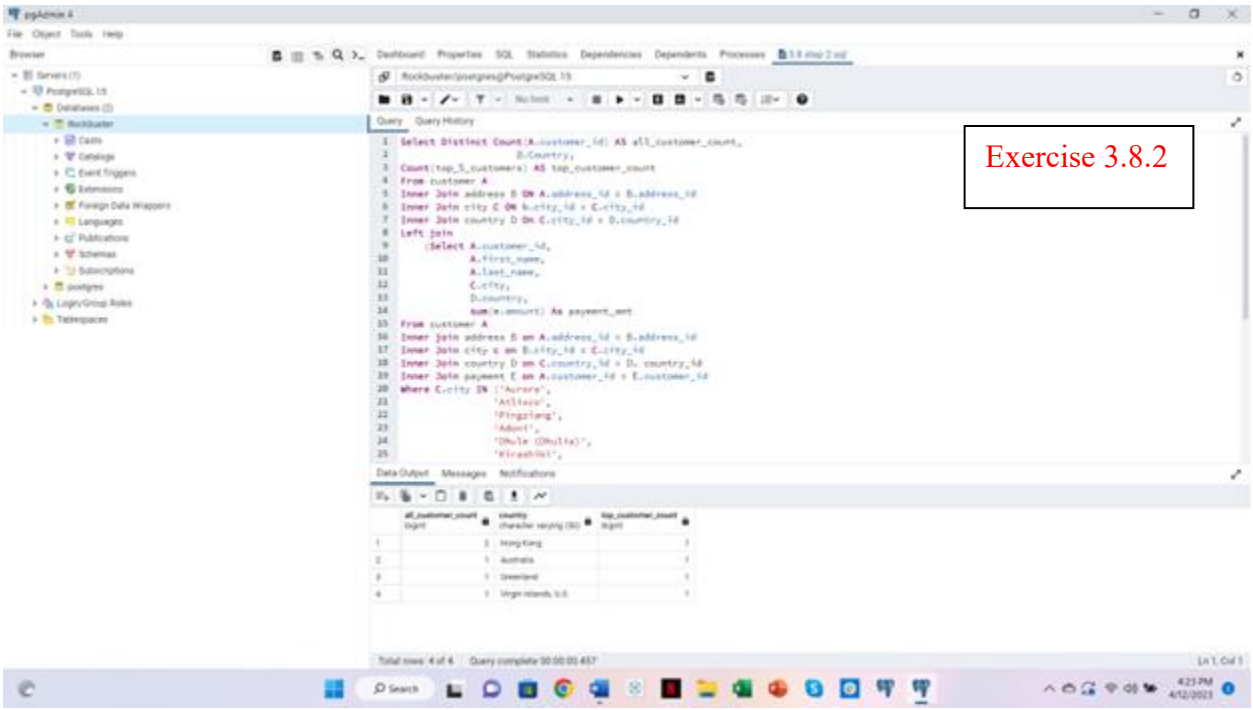
Step 2.

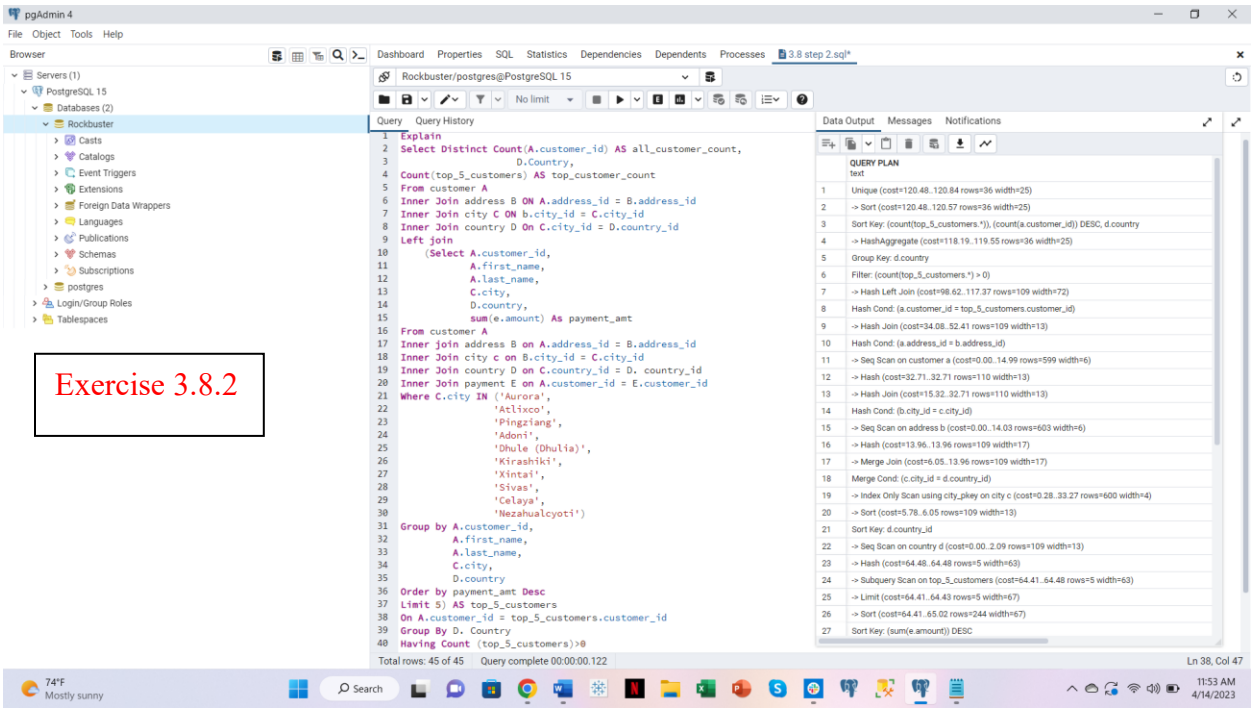

pgAdmin 4
File Object Tools Help
Browser
Dashboard Properties SQL Statistics Dependencies Dependents Processes 3.8 step 2.sql Rockbuster/postgres@PostgreSQL 15*
Rockbuster/postgres@PostgreSQL 15
Query Query History
Select Avg(payment_amt) AS average_payment
From
(Select A.customer_id,
A.first_name,
A.last_name,
C.city,
D.country,
Sum(E.amount) As payment_amt
From customer A
Inner Join address B On A.address_id = B.address_id
Inner Join city C On b.city_id = C.city_id
Inner Join country D On C.country_id = D.country_id
Inner Join payment E on A.customer_id = E.customer_id
Where C.city IN ('Aurora',
'Atlixco',
'Pingziang',
'Adoni',
'Dhule(Dhulia)',
'Kirashiki',
'Xintai',
'Sivas',
'Celaya',
'Nezahualcyoti')
Group By A.customer_id,
A.first_name,
A.last_name,
C.city,
D.country
Order by payment_amt DESC
Limit 5) AS total_amount_paid;
Exercise 3.8.1
Data Output Messages Notifications
average_payment
numeric
1 107.3540000000000000
Total rows: 1 of 1 Query complete 00:00:00.153
Ln 3, Col 22
Russell 2000
-1.29%
Search
12:08 PM
4/14/2023

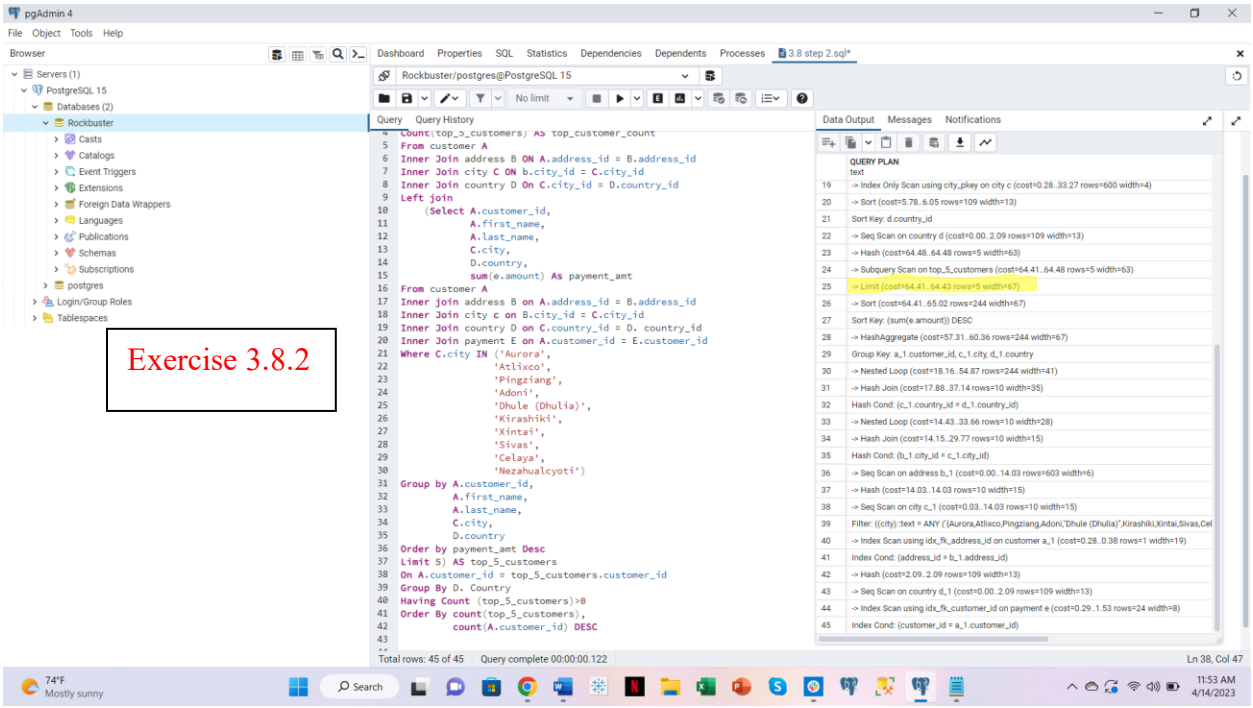

pgAdmin 4
File Object Tools Help
Browser
Dashboard Properties SQL Statistics Dependencies Dependents Processes 3.8 step 2.sql Rockbuster/postgres@PostgreSQL 15*
Rockbuster/postgres@PostgreSQL 15
Query Query History
Explain
Select Avg(payment_amt) AS average_payment
From
(Select A.customer_id,
Exercise 3.8.1
Data Output Messages Notifications
QUERY PLAN
text
1 Aggregate (cost=64.49..64.50 rows=1 width=32)
2 -> Limit (cost=64.41..64.43 rows=5 width=67)
3 -> Sort (cost=64.41..65.02 rows=244 width=67)
4 Sort Key: (sum(e.amount)) DESC
5 -> HashAggregate (cost=57.31..60.36 rows=244 width=67)
6 Group Key: a.customer_id, c.city, d.country
7 -> Nested Loop (cost=18.16..54.87 rows=244 width=41)
8 -> Hash Join (cost=17.88..37.14 rows=10 width=35)
9 Hash Cond: (c.country_id = d.country_id)
10 -> Nested Loop (cost=14.43..33.66 rows=10 width=28)
11 -> Hash Join (cost=14.15..29.77 rows=10 width=15)
12 Hash Cond: (b.city_id = c.city_id)
13 -> Seq Scan on address b (cost=0.00..14.03 rows=603 width=6)
14 -> Hash (cost=14.03..14.03 rows=10 width=15)
15 -> Seq Scan on city c (cost=0.03..14.03 rows=10 width=15)
16 Filter: ((city)::text = ANY ('{Aurora,Atlixco,Pingziang,Adoni,Dhule(Dhulia),Kirashiki,Xintai,Sivas,Celaya,Nezahualcyoti}'::text[]))
17 -> Index Scan using idx_fk_address_id on customer a (cost=0.28..0.38 rows=1 width=19)
18 Index Cond: (address_id = b.address_id)
19 -> Hash (cost=2.09..2.09 rows=109 width=13)
20 -> Seq Scan on country d (cost=0.00..2.09 rows=109 width=13)
21 -> Index Scan using idx_fk_customer_id on payment e (cost=0.29..1.53 rows=24 width=8)
22 Index Cond: (customer_id = a.customer_id)
Total rows: 22 of 22 Query complete 00:00:00.123
Ln 1, Col 8
76°F
Partly sunny
Search
12:09 PM
4/14/2023


Exercise 3.9.1


Exercise 3.9.1

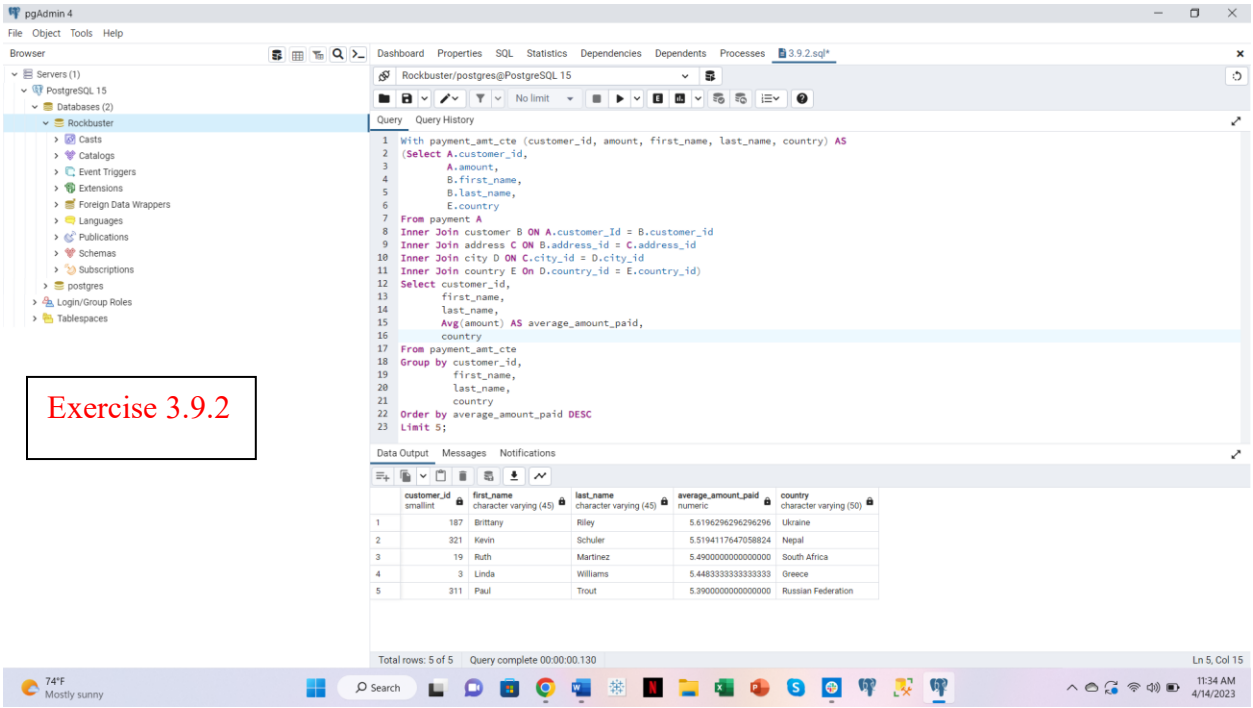

Exercise 3.8.2


pgAdmin 4
Exercise 3.8.2



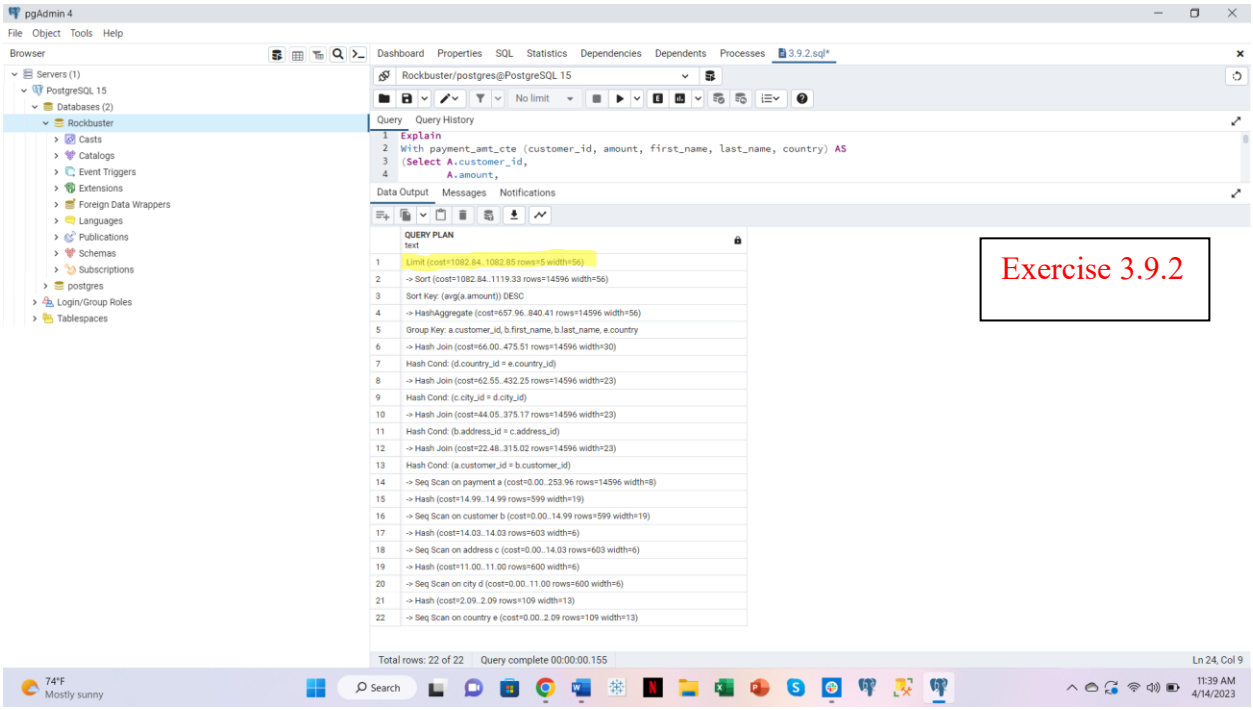

```
Count(top_5_customers) As top_customer_count
From customer A
Inner Join address B ON A.address_id = B.address_id
Inner Join city C ON b.city_id = C.city_id
Inner Join country D On C.city_id = D.country_id
Left join
    (Select A.customer_id,
            A.first_name,
            A.last_name,
            C.city,
            D.country,
            sum(e.amount) As payment_amt
From customer A
Inner join address B on A.address_id = B.address_id
Inner Join city c on B.city_id = C.city_id
Inner Join country D on C.country_id = D. country_id
Inner Join payment E on A.customer_id = E.customer_id
Where C.city IN ('Aurora',
                 'Atlixco',
                 'Pingziang',
                 'Adoni',
                 'Dhule (Dhulia)',
                 'Kirashiki',
                 'Xintai',
                 'Sivas',
                 'Celaya',
                 'Nezahualcyoti')
Group by A.customer_id,
         A.first_name,
         A.last_name,
         C.city,
         D.country
Order by payment_amt Desc
Limit 5) AS top_5_customers
On A.customer_id = top_5_customers.customer_id
Group By D. Country
Having Count (top_5_customers)>0
Order By count(top_5_customers),
         count(A.customer_id) DESC
```

```
QUERY PLAN
text
-> Index Only Scan using city_pkey on city c (cost=0.28..33.27 rows=600 width=4)
-> Sort (cost=5.78..6.05 rows=109 width=13)
Sort Key: d.country_id
-> Seq Scan on country d (cost=0.00..2.09 rows=109 width=13)
-> Hash (cost=64.48..64.48 rows=5 width=63)
-> Subquery Scan on top_5_customers (cost=64.41..64.48 rows=5 width=63)
-> Limit (cost=64.41..64.43 rows=5 width=67)
-> Sort (cost=64.41..65.02 rows=244 width=67)
Sort Key: (sum(e.amount)) DESC
-> HashAggregate (cost=57.31..60.36 rows=244 width=67)
Group Key: a_1.customer_id, c_1.city, d_1.country
-> Nested Loop (cost=18.16..54.87 rows=244 width=41)
-> Hash Join (cost=17.88..37.14 rows=10 width=35)
Hash Cond: (c_1.country_id = d_1.country_id)
-> Nested Loop (cost=14.43..33.66 rows=10 width=28)
-> Hash Join (cost=14.15..29.77 rows=10 width=15)
Hash Cond: (b_1.city_id = c_1.city_id)
-> Seq Scan on address b_1 (cost=0.00..14.03 rows=603 width=6)
-> Hash (cost=14.03..14.03 rows=10 width=15)
-> Seq Scan on city c_1 (cost=0.03..14.03 rows=10 width=15)
Filter: ((city)::text = ANY ('{Aurora,Atlixco,Pingziang,Adoni,"Dhule (Dhulia)",Kirashiki,Xintai,Sivas,Cel
-> Index Scan using idx_fk_address_id on customer a_1 (cost=0.28..0.38 rows=1 width=19)
Index Cond: (address_id = b_1.address_id)
-> Hash (cost=2.09..2.09 rows=109 width=13)
-> Seq Scan on country d_1 (cost=0.00..2.09 rows=109 width=13)
-> Index Scan using idx_fk_customer_id on payment e (cost=0.29..1.53 rows=24 width=8)
Index Cond: (customer_id = a_1.customer_id)
```




```
With payment_amt_cte (customer_id, amount, first_name, last_name, country) AS
(Select A.customer_id,
        A.amount,
        B.first_name,
        B.last_name,
        E.country
From payment A
Inner Join customer B ON A.customer_Id = B.customer_id
Inner Join address C ON B.address_id = C.address_id
Inner Join city D ON C.city_id = D.city_id
Inner Join country E On D.country_id = E.country_id)
Select customer_id,
       first_name,
       last_name,
       Avg(amount) AS average_amount_paid,
       country
From payment_amt_cte
Group by customer_id,
         first_name,
         last_name,
         country
Order by average_amount_paid DESC
Limit 5;
```

	customer_id smallint	first_name character varying (45)	last_name character varying (45)	average_amount_paid numeric	country character varying (50)
1	187	Brittany	Riley	5.6196296296296296	Ukraine
2	321	Kevin	Schuler	5.5194117647058824	Nepal
3	19	Ruth	Martinez	5.4900000000000000	South Africa
4	3	Linda	Williams	5.4483333333333333	Greece
5	311	Paul	Trout	5.3900000000000000	Russian Federation


pgAdmin 4
File Object Tools Help
Browser
Servers (1)
PostgreSQL 15
Databases (2)
Rockbuster
Casts
Catalogs
Event Triggers
Extensions
Foreign Data Wrappers
Languages
Publications
Schemas
Subscriptions
postgres
Login/Group Roles
Tablespaces
Dashboard Properties SQL Statistics Dependencies Dependents Processes 3.9.2.sql*
Rockbuster/postgres@PostgreSQL 15
No limit
Query Query History
1 Explain
2 With payment_amt_cte (customer_id, amount, first_name, last_name, country) AS
3 (Select A.customer_id,
4 A.amount,
Data Output Messages Notifications
QUERY PLAN
text
1 Limit (cost=1082.84..1082.85 rows=5 width=56)
2 -> Sort (cost=1082.84..1119.33 rows=14596 width=56)
3 Sort Key: (avg(a.amount)) DESC
4 -> HashAggregate (cost=657.96..840.41 rows=14596 width=56)
5 Group Key: a.customer_id, b.first_name, b.last_name, e.country
6 -> Hash Join (cost=66.00..475.51 rows=14596 width=30)
7 Hash Cond: (d.country_id = e.country_id)
8 -> Hash Join (cost=62.55..432.25 rows=14596 width=23)
9 Hash Cond: (c.city_id = d.city_id)
10 -> Hash Join (cost=44.05..375.17 rows=14596 width=23)
11 Hash Cond: (b.address_id = c.address_id)
12 -> Hash Join (cost=22.48..315.02 rows=14596 width=23)
13 Hash Cond: (a.customer_id = b.customer_id)
14 -> Seq Scan on payment a (cost=0.00..253.96 rows=14596 width=8)
15 -> Hash (cost=14.99..14.99 rows=599 width=19)
16 -> Seq Scan on customer b (cost=0.00..14.99 rows=599 width=19)
17 -> Hash (cost=14.03..14.03 rows=603 width=6)
18 -> Seq Scan on address c (cost=0.00..14.03 rows=603 width=6)
19 -> Hash (cost=11.00..11.00 rows=600 width=6)
20 -> Seq Scan on city d (cost=0.00..11.00 rows=600 width=6)
21 -> Hash (cost=2.09..2.09 rows=109 width=13)
22 -> Seq Scan on country e (cost=0.00..2.09 rows=109 width=13)
Total rows: 22 of 22
Query complete 00:00:00.155
Ln 24, Col 9
Exercise 3.9.2
74°F
Mostly sunny
Search
11:39 AM
4/14/2023