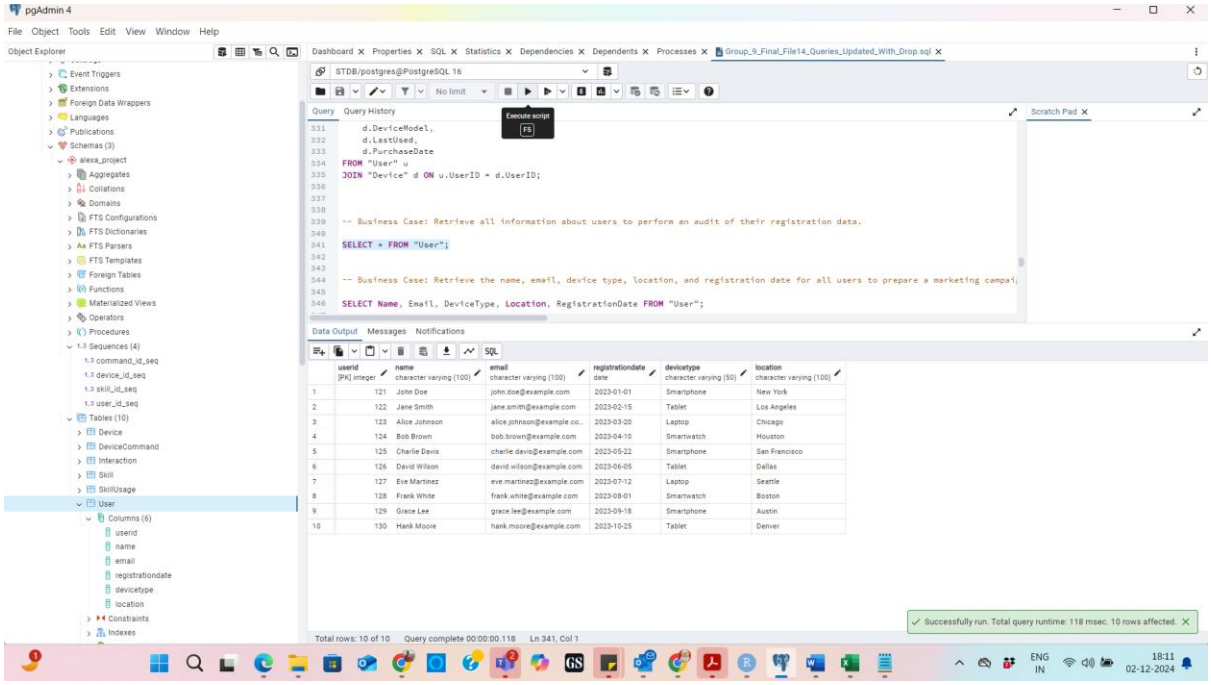
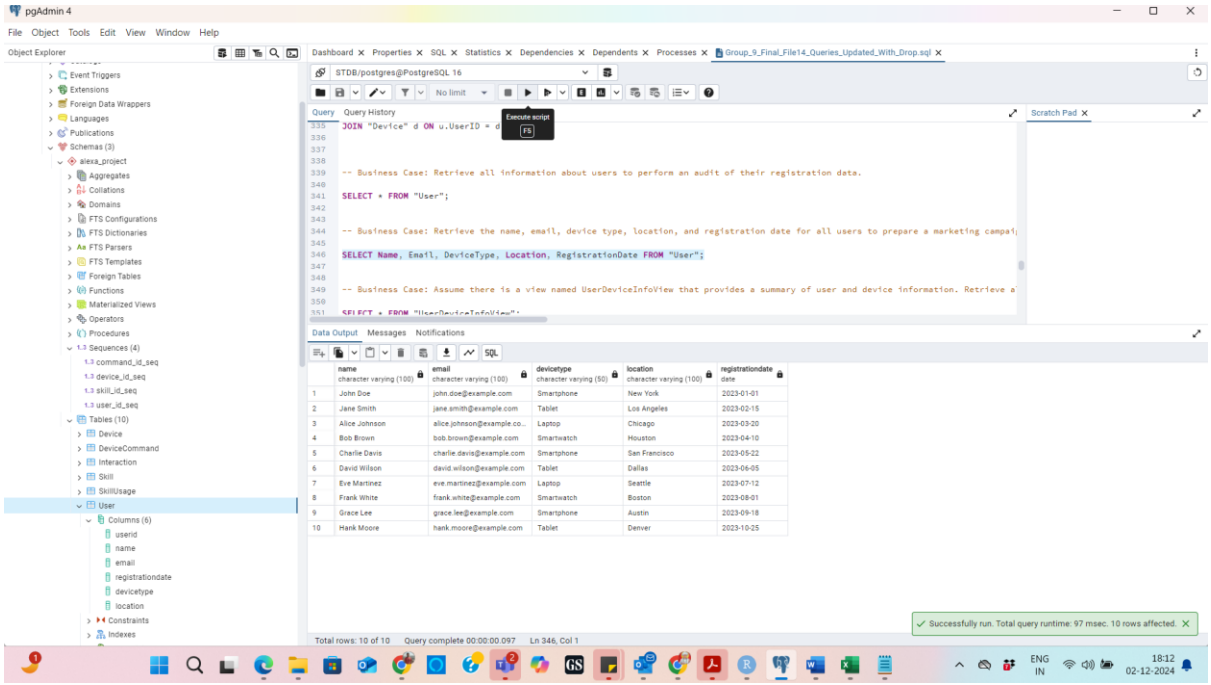
14 SQL Queries (12 Basic, 2 Advanced) Minimum Requirements:

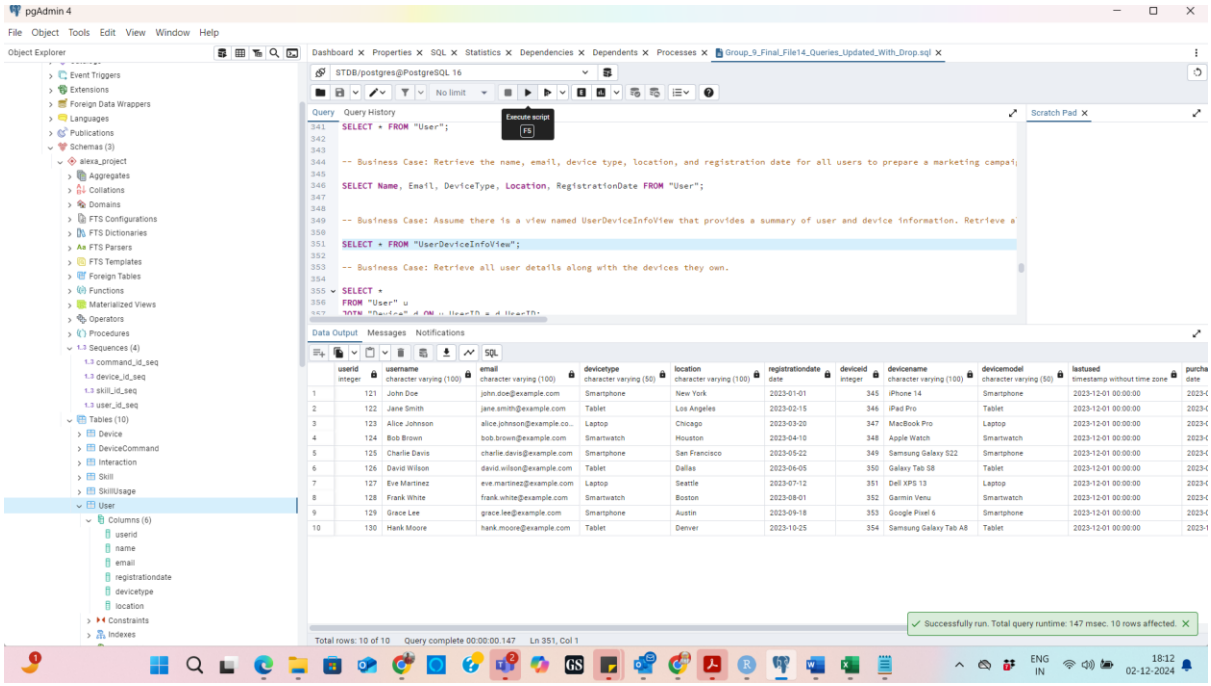
- Query 1: Select all columns and all rows from one table (5 points)



- Query 2: Select five columns and all rows from one table (5 points)

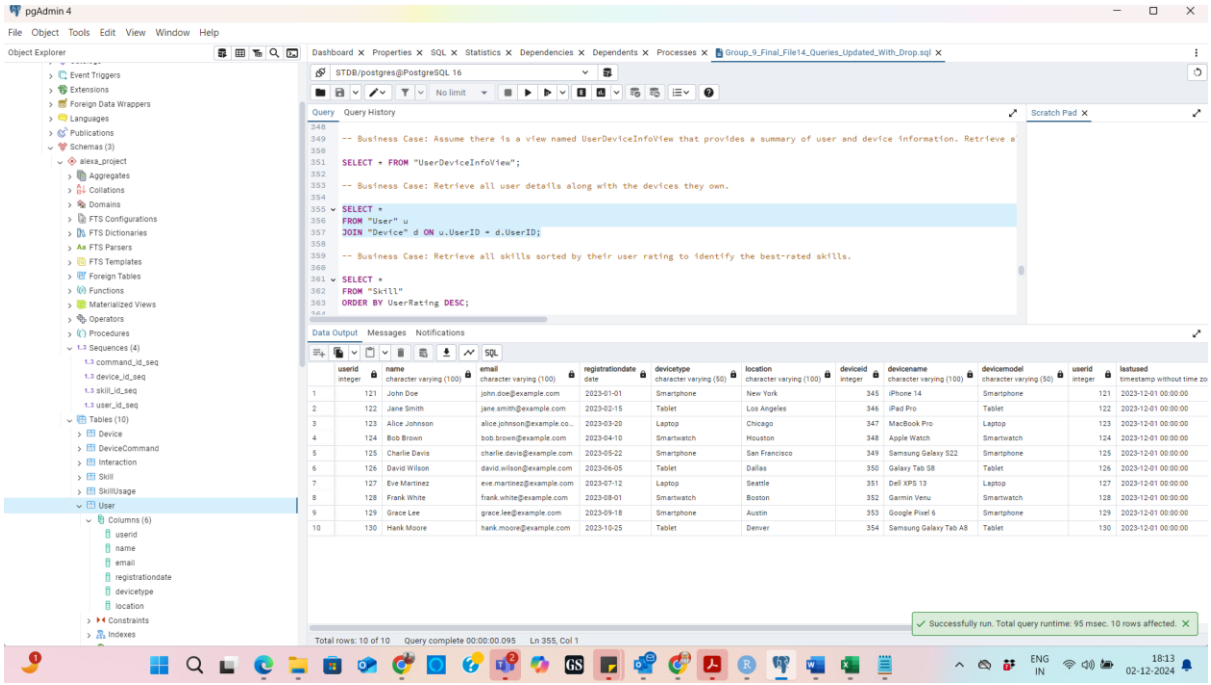


- Query 3: Select all columns from all rows from one view (5 points)

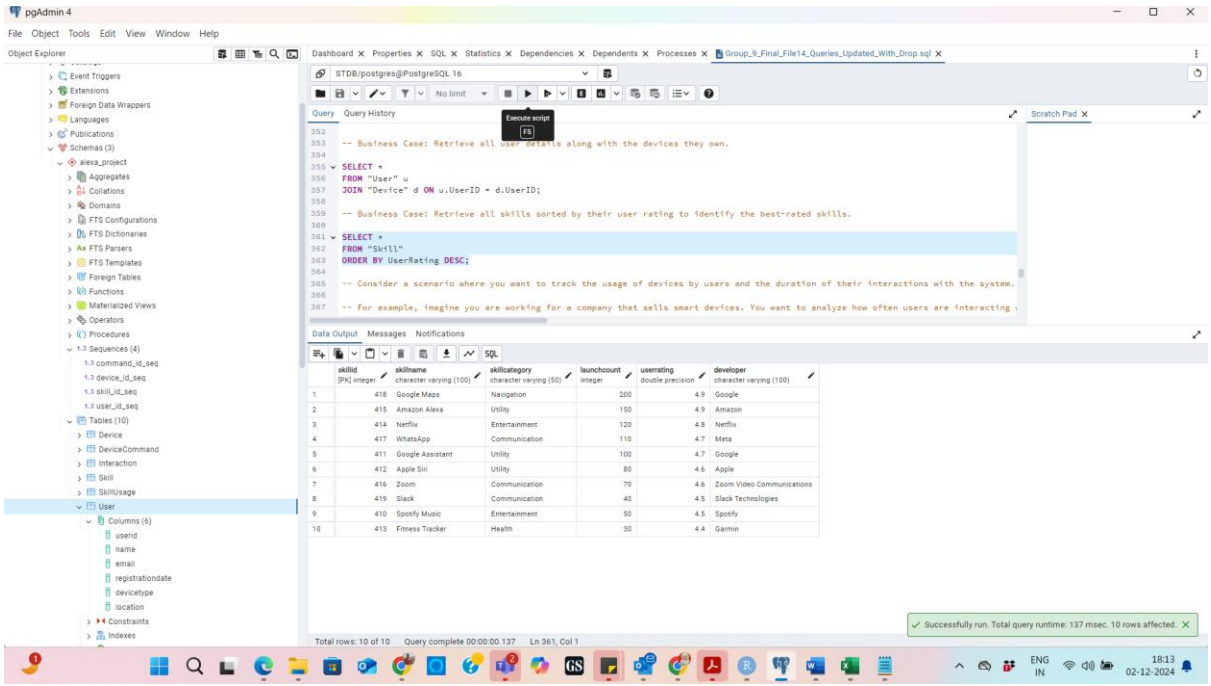



```sql
CREATE VIEW "UserDeviceInfoView" AS
SELECT
    u.UserID,
    u.Name AS UserName,
    u.Email,
    u.DeviceType,
    u.Location,
    u.RegistrationDate,
    d.DeviceID,
    d.DeviceName,
    d.DeviceModel,
    d.LastUsed,
    d.PurchaseDate
FROM "User" u
JOIN "Device" d ON u.UserID = d.UserID;
```

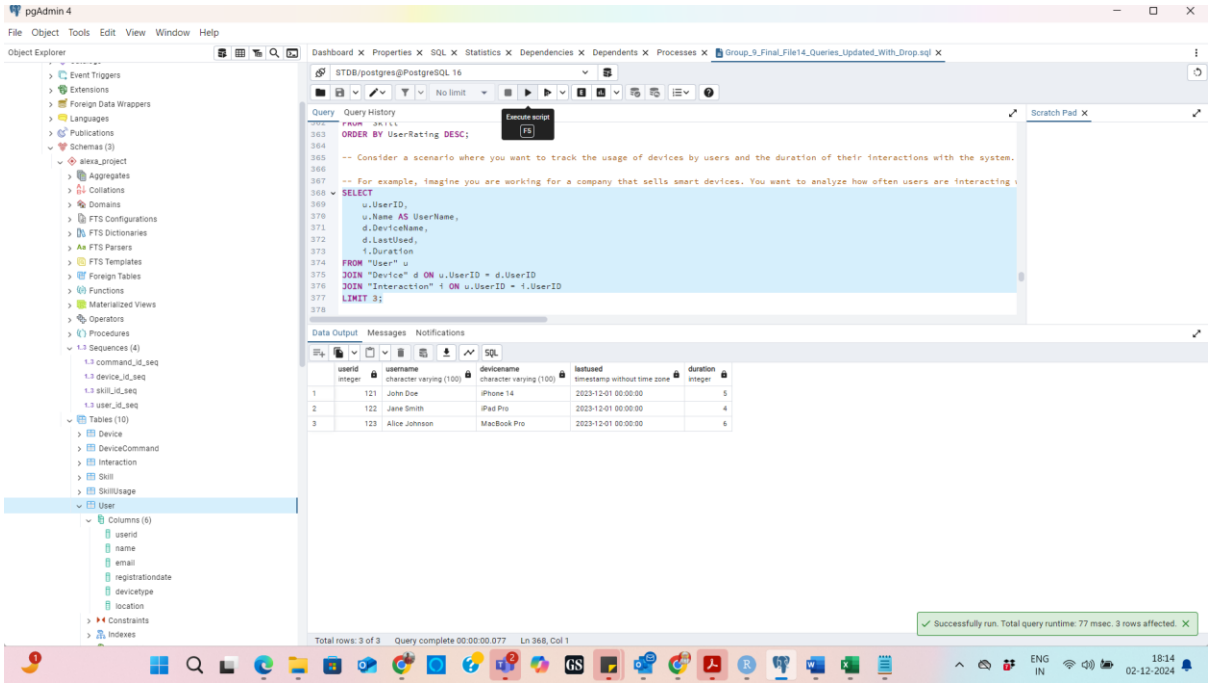
- Query 4: Using a join on 2 tables, select all columns and all rows from the tables without the use of a Cartesian product (5 points)



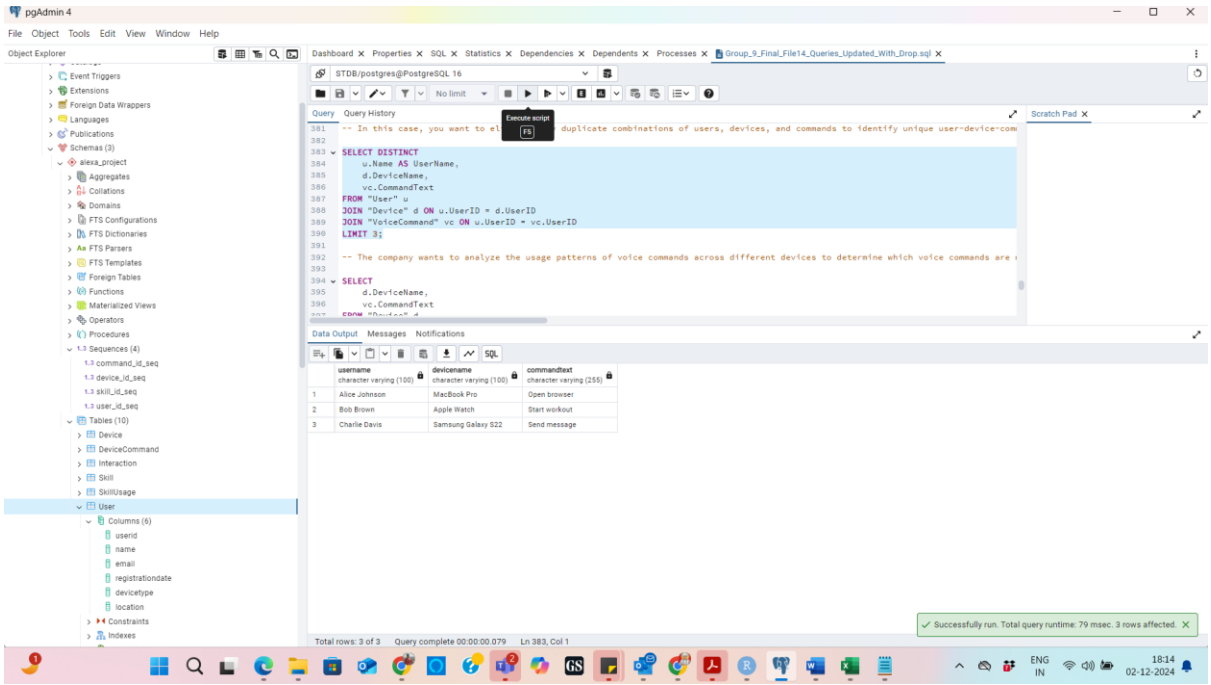
- Query 5: Select and order data retrieved from one table (5 points)



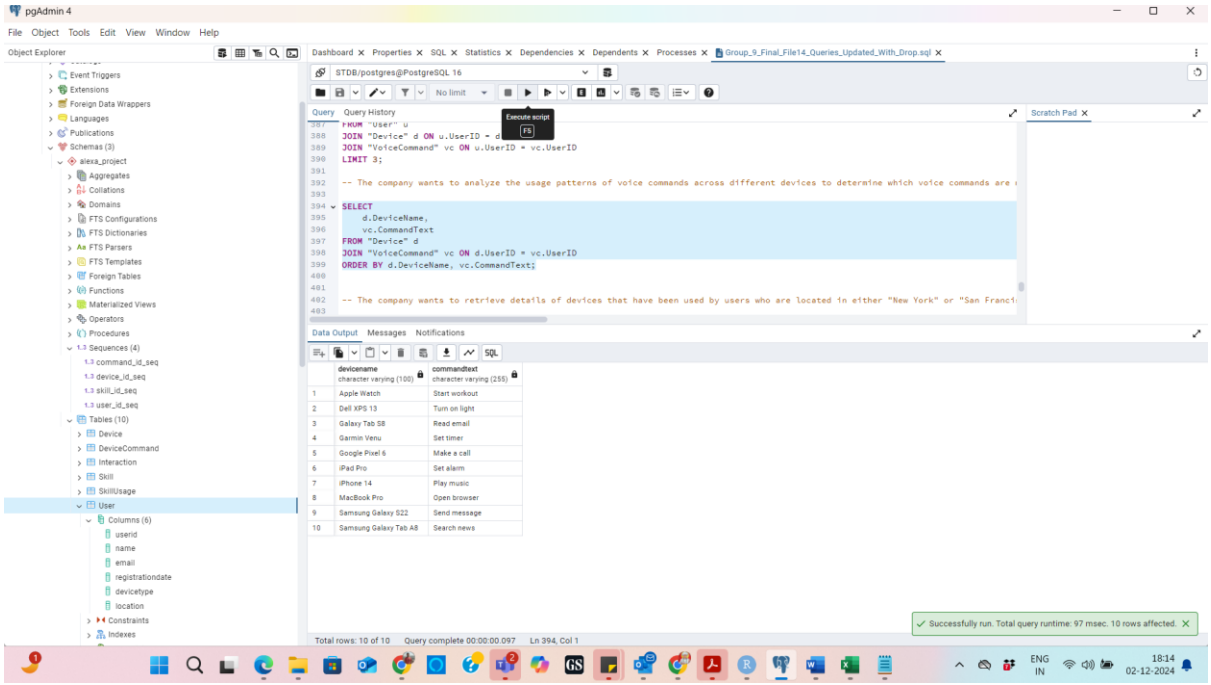
- Query 6: Using a join on 3 tables, select 5 columns from the 3 tables. Use syntax that would limit the output to 3 rows (5 points)



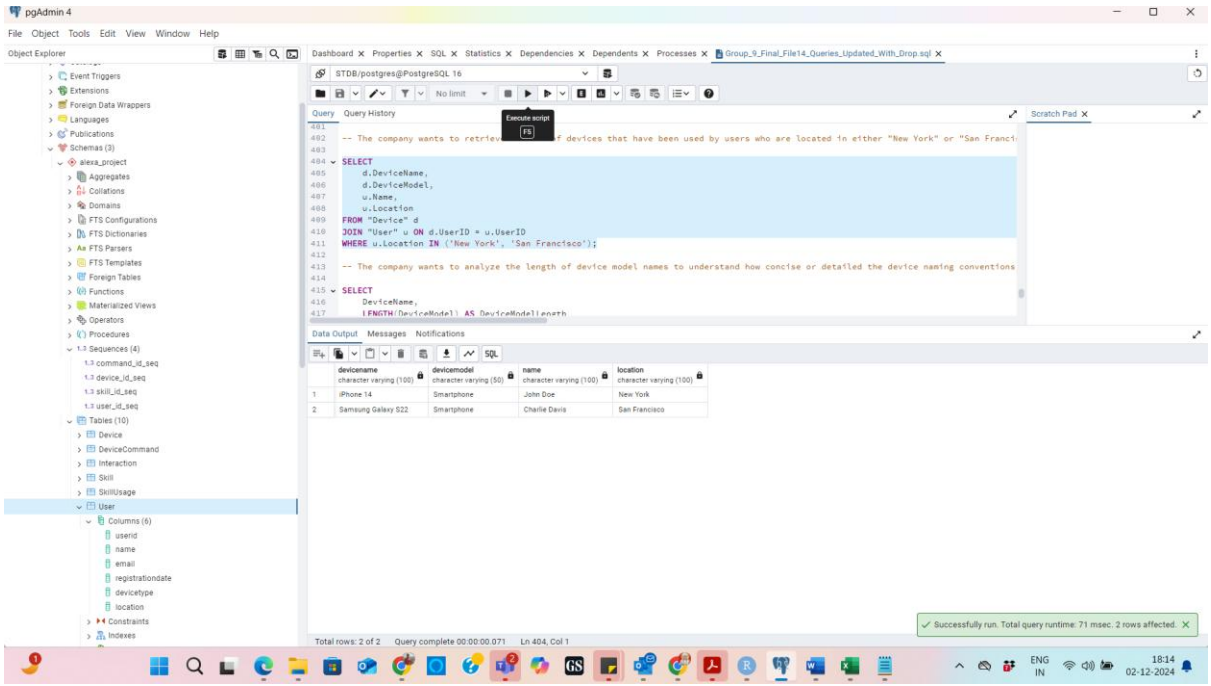
- Query 7: Select distinct rows using joins on 3 tables (5 points)



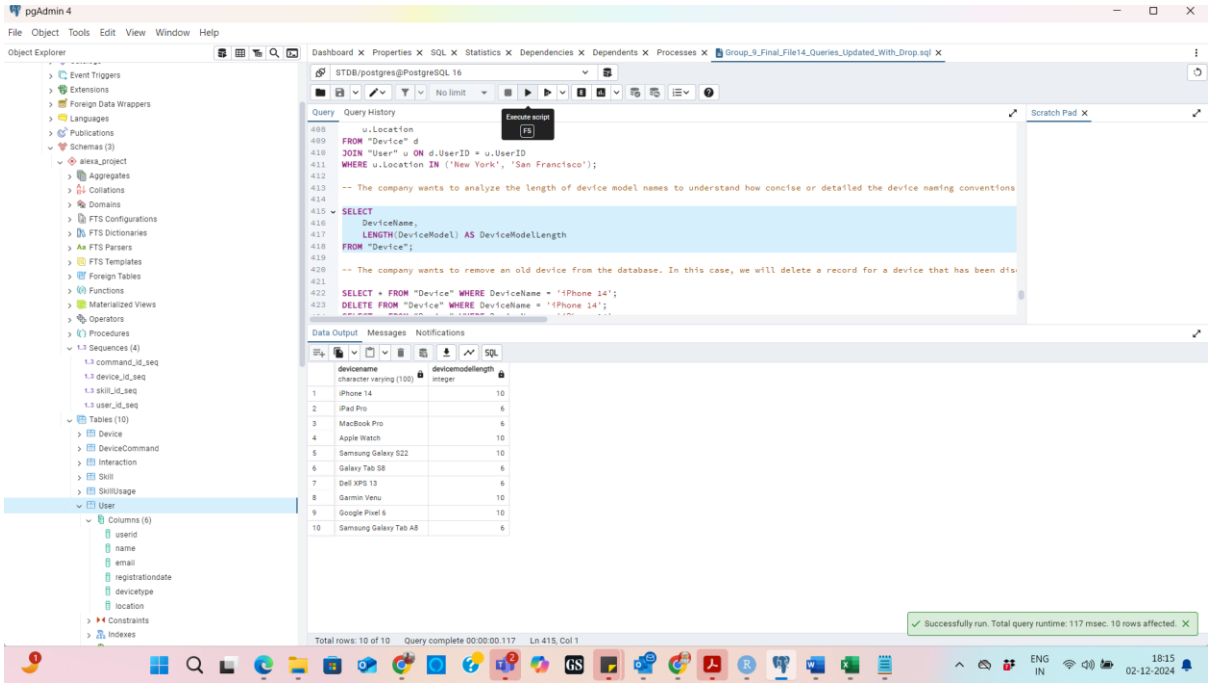
- Query 8: Use GROUP BY and HAVING in a select statement using one or more tables (5 points)



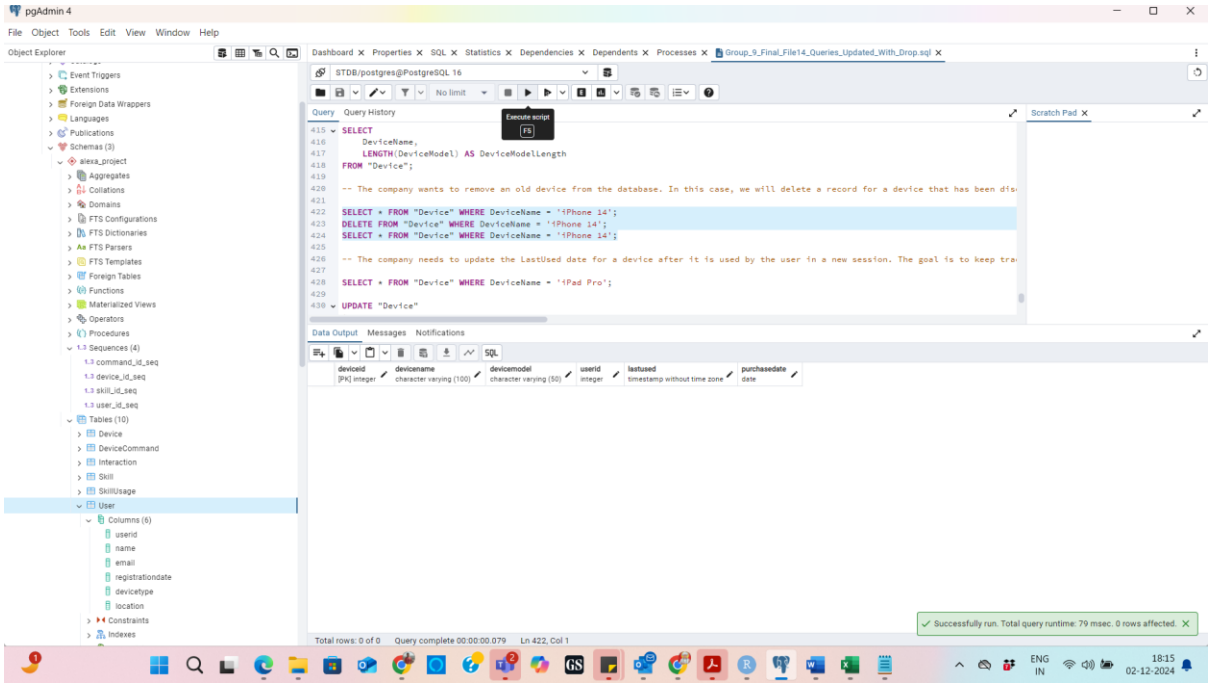
- Query 9: Use IN clause to select data from one or more tables (5 points)



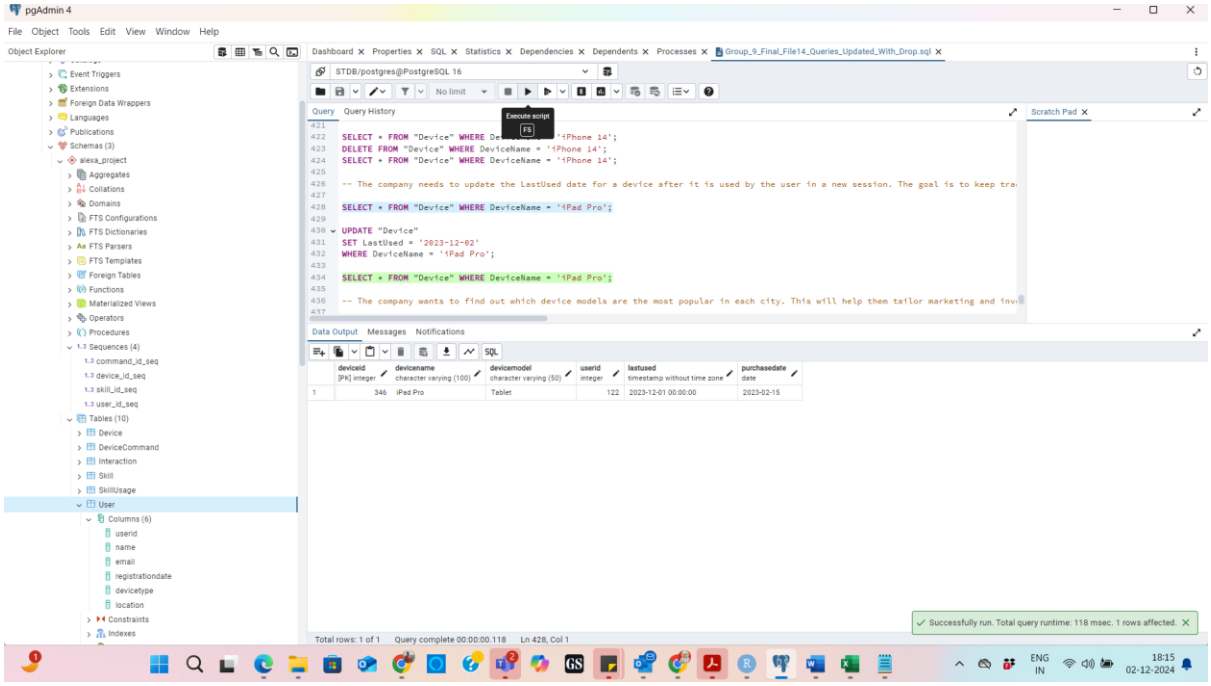
- Query 10: Select length of one column from one table (use LENGTH function) (5 points)

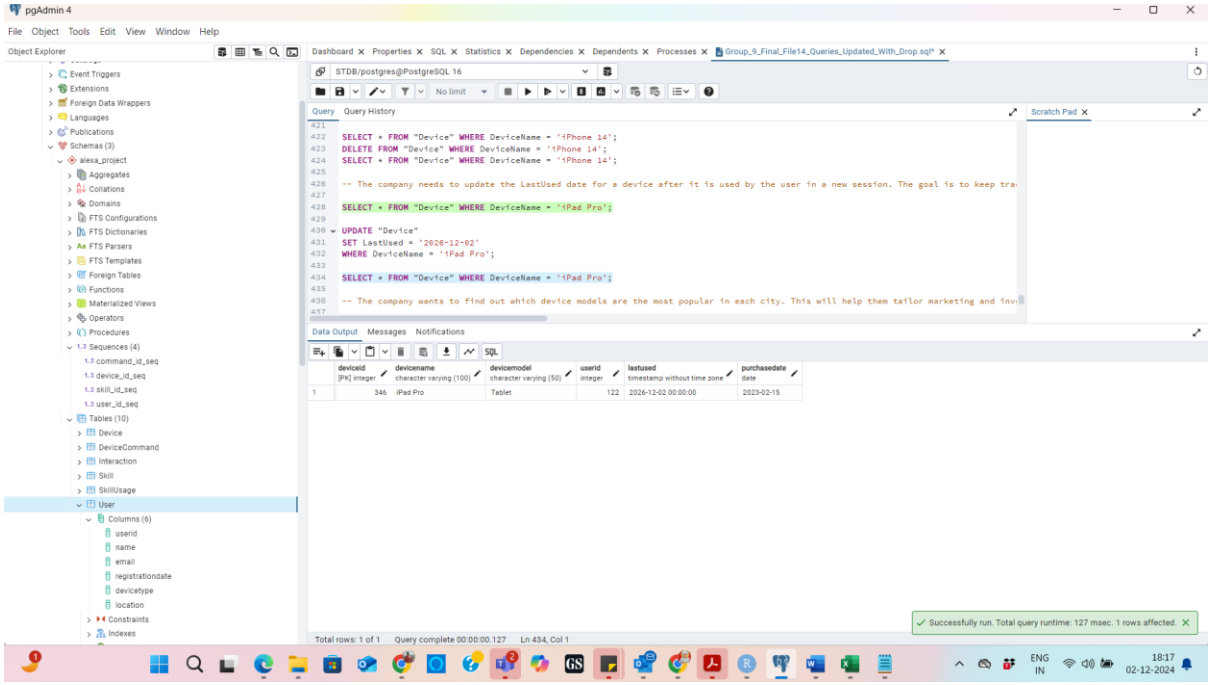


- Query 11: Delete one record from one table. Use select statements to demonstrate the table contents before and after the DELETE statement. Make sure you use ROLLBACK afterwards so that the data will not be physically removed (5 points)



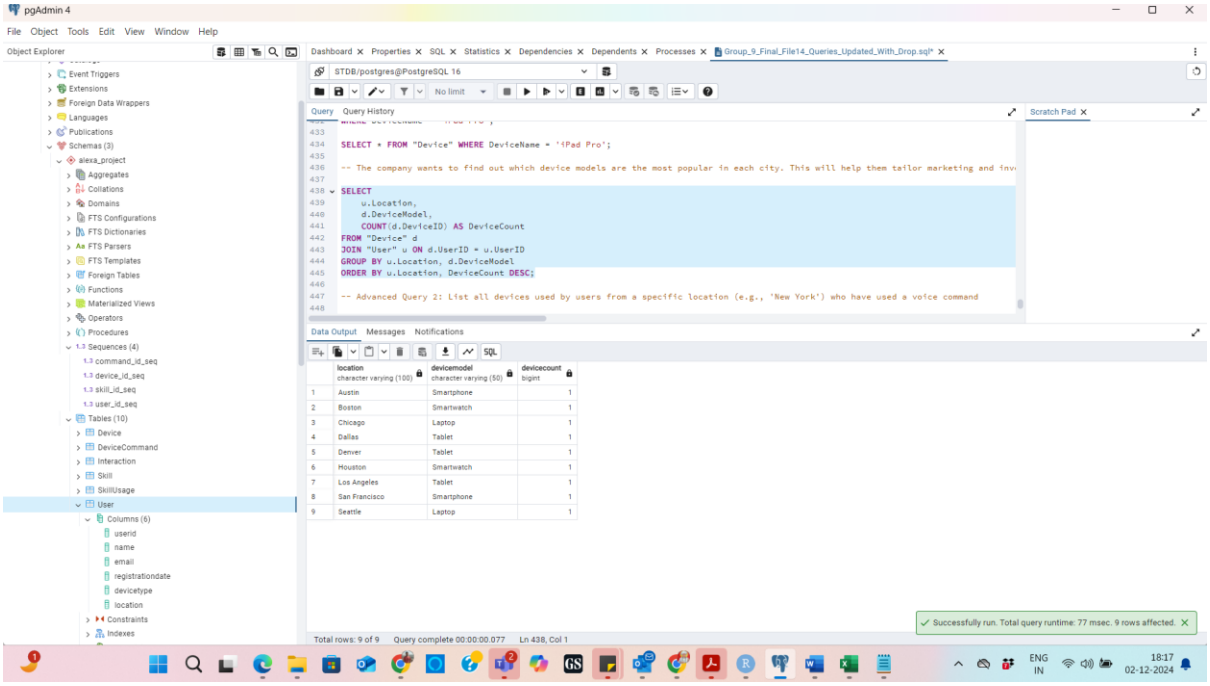
- Query 12: Update one record from one table. Use select statements to demonstrate the table contents before and after the UPDATE statement. Make sure you use ROLLBACK afterwards so that the data will not be physically removed (5 points)



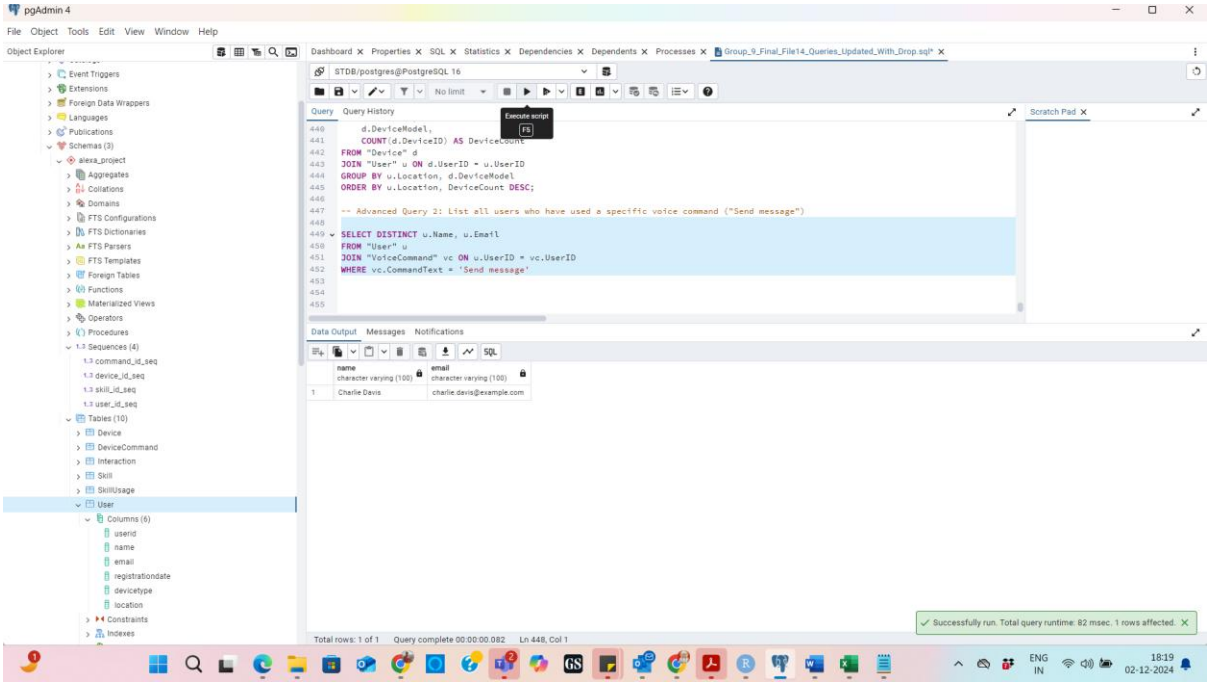


Perform 2 Additional Advanced Queries (40 points)

1)


pgAdmin 4
File Object Tools Edit View Window Help
Object Explorer
Dashboard Properties SQL Statistics Dependencies Dependents Processes Group_9_Final_File14_Queries_Updated_With_Drop.sql*
STDB/postgres@PostgreSQL 16
Query Query History
Scratch Pad
SELECT * FROM "Device" WHERE DeviceName = 'iPad Pro';
-- The company wants to find out which device models are the most popular in each city. This will help them tailor marketing and inv
SELECT
u.Location,
d.DeviceModel,
COUNT(d.DeviceID) AS DeviceCount
FROM "Device" d
JOIN "User" u ON d.UserID = u.UserID
GROUP BY u.Location, d.DeviceModel
ORDER BY u.Location, DeviceCount DESC;
-- Advanced Query 2: List all devices used by users from a specific location (e.g., 'New York') who have used a voice command
Data Output Messages Notifications
location character varying (100)
devicemodel character varying (50)
devicecount bigint
1 Austin Smartphone 1
2 Boston Smartwatch 1
3 Chicago Laptop 1
4 Dallas Tablet 1
5 Denver Tablet 1
6 Houston Smartwatch 1
7 Los Angeles Tablet 1
8 San Francisco Smartphone 1
9 Seattle Laptop 1
Total rows: 9 of 9 Query complete 00:00:00.077 Ln 438, Col 1
Successfully run. Total query runtime: 77 msec. 9 rows affected.
ENG IN 18:17 02-12-2024

2)


pgAdmin 4
File Object Tools Edit View Window Help
Object Explorer
Dashboard Properties SQL Statistics Dependencies Dependents Processes Group_9_Final_File14_Queries_Updated_With_Drop.sql*
STDB/postgres@PostgreSQL 16
Query Query History
Execute script
F5
Scratch Pad
d.DeviceModel,
COUNT(d.DeviceID) AS DeviceCount
FROM "Device" d
JOIN "User" u ON d.UserID = u.UserID
GROUP BY u.Location, d.DeviceModel
ORDER BY u.Location, DeviceCount DESC;
-- Advanced Query 2: List all users who have used a specific voice command ("Send message")
SELECT DISTINCT u.Name, u.Email
FROM "User" u
JOIN "VoiceCommand" vc ON u.UserID = vc.UserID
WHERE vc.CommandText = 'Send message'
Data Output Messages Notifications
name character varying (100)
email character varying (100)
1 Charlie Davis charlie.davis@example.com
Total rows: 1 of 1 Query complete 00:00:00.082 Ln 448, Col 1
Successfully run. Total query runtime: 82 msec. 1 rows affected.
ENG IN 18:19 02-12-2024