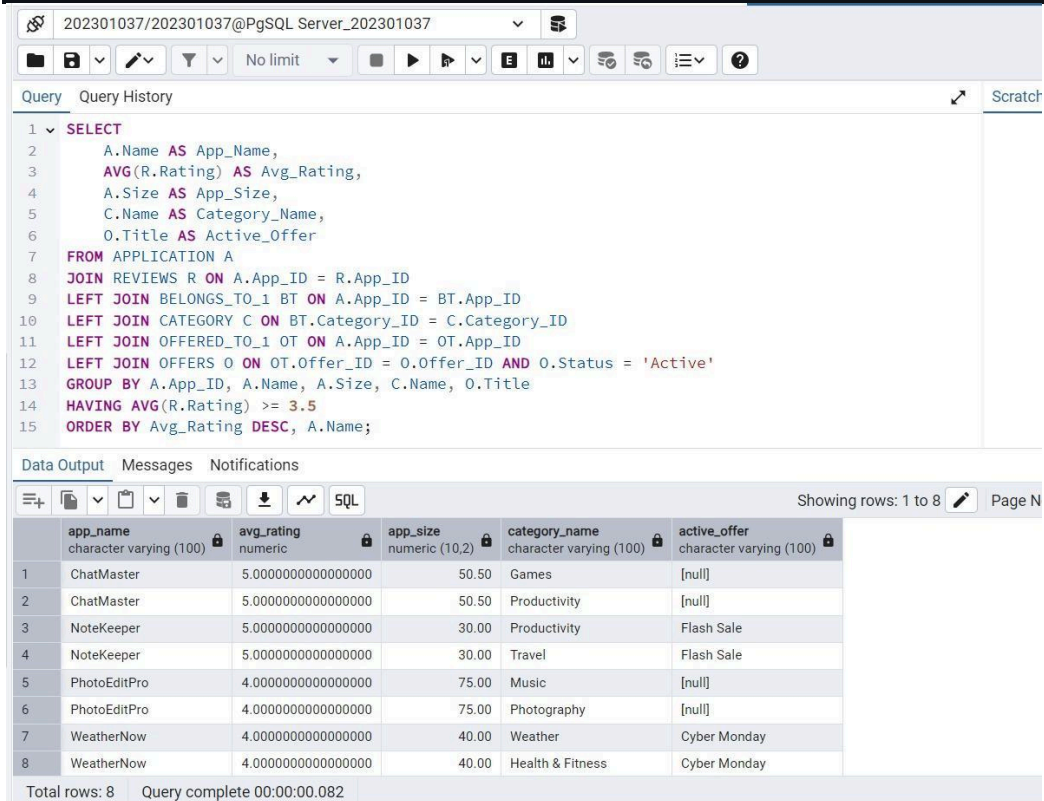


Querying Operations:

1. Top Rated Apps (Average Rating ≥ 3.5)

```
SELECT
    A.Name AS App_Name,
    AVG(R.Rating) AS Avg_Rating,
    A.Size AS App_Size,
    C.Name AS Category_Name,
    O.Title AS Active_Offer
FROM APPLICATION A
JOIN REVIEWS R ON A.App_ID = R.App_ID
LEFT JOIN BELONGS_TO_1 BT ON A.App_ID = BT.App_ID
LEFT JOIN CATEGORY C ON BT.Category_ID = C.Category_ID
LEFT JOIN OFFERED_TO_1 OT ON A.App_ID = OT.App_ID
LEFT JOIN OFFERS O ON OT.Offer_ID = O.Offer_ID AND O.Status = 'Active'
GROUP BY A.App_ID, A.Name, A.Size, C.Name, O.Title
HAVING AVG(R.Rating) >= 3.5
ORDER BY Avg_Rating DESC, A.Name;
```



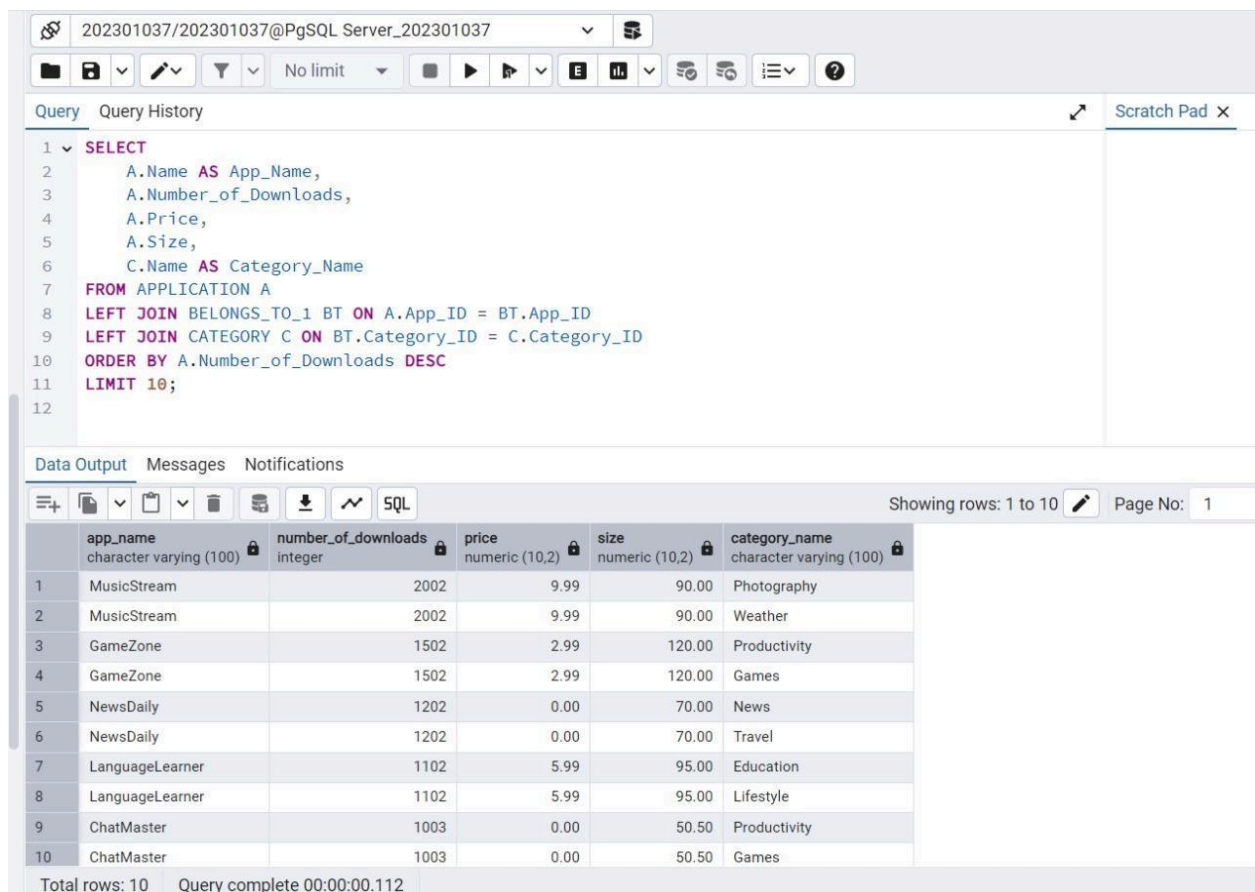
The screenshot shows a PostgreSQL query editor interface. The top bar displays the connection string '202301037/202301037@PgSQL Server_202301037'. Below the toolbar, the 'Query' tab is active, showing the SQL query. The 'Data Output' tab is also visible, showing the results of the query. The results are displayed in a table with 5 columns: app_name, avg_rating, app_size, category_name, and active_offer. The table contains 8 rows of data, sorted by average rating in descending order. The bottom status bar indicates 'Total rows: 8' and 'Query complete 00:00:00.082'.

	app_name character varying (100)	avg_rating numeric	app_size numeric (10,2)	category_name character varying (100)	active_offer character varying (100)
1	ChatMaster	5.0000000000000000	50.50	Games	[null]
2	ChatMaster	5.0000000000000000	50.50	Productivity	[null]
3	NoteKeeper	5.0000000000000000	30.00	Productivity	Flash Sale
4	NoteKeeper	5.0000000000000000	30.00	Travel	Flash Sale
5	PhotoEditPro	4.0000000000000000	75.00	Music	[null]
6	PhotoEditPro	4.0000000000000000	75.00	Photography	[null]
7	WeatherNow	4.0000000000000000	40.00	Weather	Cyber Monday
8	WeatherNow	4.0000000000000000	40.00	Health & Fitness	Cyber Monday

Total rows: 8 Query complete 00:00:00.082

2. Most Downloaded Apps:

```
SELECT
    A.Name AS App_Name,
    A.Number_of_Downloads,
    A.Price,
    A.Size,
    C.Name AS Category_Name
FROM APPLICATION A
LEFT JOIN BELONGS_TO_1 BT ON A.App_ID = BT.App_ID
LEFT JOIN CATEGORY C ON BT.Category_ID = C.Category_ID
ORDER BY A.Number_of_Downloads DESC
LIMIT 10;
```



The screenshot shows a PostgreSQL query editor interface. The top bar displays the connection string '202301037/202301037@PgSQL Server_202301037'. Below the toolbar, the 'Query' tab is active, showing the SQL query. The 'Data Output' tab is also visible, displaying the results of the query in a table format. The table has 6 columns: app_name, number_of_downloads, price, size, and category_name. The results are sorted by number_of_downloads in descending order, with the top 10 rows displayed.

	app_name character varying (100)	number_of_downloads integer	price numeric (10,2)	size numeric (10,2)	category_name character varying (100)
1	MusicStream	2002	9.99	90.00	Photography
2	MusicStream	2002	9.99	90.00	Weather
3	GameZone	1502	2.99	120.00	Productivity
4	GameZone	1502	2.99	120.00	Games
5	NewsDaily	1202	0.00	70.00	News
6	NewsDaily	1202	0.00	70.00	Travel
7	LanguageLearner	1102	5.99	95.00	Education
8	LanguageLearner	1102	5.99	95.00	Lifestyle
9	ChatMaster	1003	0.00	50.50	Productivity
10	ChatMaster	1003	0.00	50.50	Games

Total rows: 10 Query complete 00:00:00.112

3. Find All Apps/Books Under Active Offers:

```
SELECT
    A.Name AS Product_Name,
    'App' AS Product_Type,
    A.Price,
    D.Name AS Developer_or_Publisher,
    O.Title AS Offer_Title,
    O.Discount_Percentage,
    O.Start_Date,
    O.End_Date
FROM APPLICATION A
JOIN OFFERED_TO_1 OT ON A.App_ID = OT.App_ID
JOIN OFFERS O ON OT.Offer_ID = O.Offer_ID
LEFT JOIN CREATED_BY CB ON A.App_ID = CB.App_ID
LEFT JOIN DEVELOPER D ON CB.Developer_ID = D.Developer_ID
WHERE CURRENT_DATE BETWEEN O.Start_Date AND O.End_Date

UNION

SELECT
    B.Title AS Product_Name,
    'Book' AS Product_Type,
    B.Price,
    B.Publisher AS Developer_or_Publisher,
    O.Title AS Offer_Title,
    O.Discount_Percentage,
    O.Start_Date,
    O.End_Date
FROM BOOKS B
JOIN OFFERED_TO_2 OT ON B.Book_ID = OT.Book_ID
JOIN OFFERS O ON OT.Offer_ID = O.Offer_ID
WHERE CURRENT_DATE BETWEEN O.Start_Date AND O.End_Date;
```

202301037/202301037@PgSQL Server_202301037

Query Query History Scratch Pad x

```

1 SELECT
2     A.Name AS Product_Name,
3     'App' AS Product_Type,
4     A.Price,
5     D.Name AS Developer_or_Publisher,
6     O.Title AS Offer_Title,
7     O.Discount_Percentage,
8     O.Start_Date,
9     O.End_Date
10  FROM APPLICATION A
11  JOIN OFFERED_TO_1 OT ON A.App_ID = OT.App_ID
12  JOIN OFFERS O ON OT.Offer_ID = O.Offer_ID
13  LEFT JOIN CREATED_BY CB ON A.App_ID = CB.App_ID
14  LEFT JOIN DEVELOPER D ON CB.Developer_ID = D.Developer_ID
15  WHERE CURRENT_DATE BETWEEN O.Start_Date AND O.End_Date
16
17  UNION
18
19  SELECT
20     B.Title AS Product_Name,
21     B.Product_Type AS Product_Type,

```

Data Output Messages Notifications

Showing rows: 1 to 1 Page No: 1

	product_name character varying (100)	product_type text	price numeric (10,2)	developer_or_publisher character varying (100)	offer_title character varying (100)	discount_percentage numeric (5,2)	start_date date	end_date date
1	ChatMaster	App	0.00	GammaTech	Student Discount	25.00	2025-04-01	2025-04-30

4. Purchase History of a User:

```

SELECT B1.User_ID,
       P.Purchase_Date,
       P.Amount,
       P.Payment_Method,
       'App' AS Product_Type,
       A.Name AS Product_Name
FROM PURCHASE P
JOIN BUY_APP B1 ON P.Order_ID = B1.Order_ID
JOIN APPLICATION A ON P.Product_ID = A.App_ID
WHERE B1.User_ID = 4

UNION

SELECT B2.User_ID,
       P.Purchase_Date,
       P.Amount,

```

```

        P.Payment_Method,
        'Book' AS Product_Type,
        B.Title AS Product_Name
FROM PURCHASE P
JOIN BUY_BOOK B2 ON P.Order_ID = B2.Order_ID
JOIN BOOKS B ON P.Product_ID = B.Book_ID
WHERE B2.User_ID = 4

ORDER BY Purchase_Date DESC;

```

202301037/202301037@PgSQL Server_202301037

Execute script (F5)

```

1 SELECT B1.User_ID,
2     P.Purchase_Date,
3     P.Amount,
4     P.Payment_Method,
5     'App' AS Product_Type,
6     A.Name AS Product_Name
7 FROM PURCHASE P
8 JOIN BUY_APP B1 ON P.Order_ID = B1.Order_ID
9 JOIN APPLICATION A ON P.Product_ID = A.App_ID
10 WHERE B1.User_ID = 4
11
12 UNION
13
14 SELECT B2.User_ID,
15     P.Purchase_Date,
16     P.Amount,
17     P.Payment_Method,
18     'Book' AS Product_Type,
19     B.Title AS Product_Name
20 FROM PURCHASE P
21 JOIN BUY_BOOK B2 ON P.Order_ID = B2.Order_ID

```

Data Output Messages Notifications

Showing rows: 1 to 2 Page 1

	user_id integer	purchase_date date	amount numeric (10,2)	payment_method character varying (50)	product_type text	product_name character varying (100)
1	4	2023-07-11	12.99	Credit Card	App	WeatherNow
2	4	2023-04-26	11.99	Debit Card	Book	Healthy Living

Total rows: 2 Query complete 00:00:00.169

5.Apps By Preferred Permission(Eg. camera and Storage):

```
SELECT
    A.App_ID,
    A.Name AS App_Name
FROM APPLICATION A
JOIN (
    SELECT App_ID
    FROM PERMISSION
    WHERE Permission IN ('Camera', 'Storage')
    GROUP BY App_ID
    HAVING COUNT(DISTINCT Permission) = 2
) AS FilteredApps ON A.App_ID = FilteredApps.App_ID
ORDER BY A.App_ID;
```

202301037/202301037@PgSQL Server_202301037

Open File
Ctrl O

Query History

```
1 SELECT
2     A.App_ID,
3     A.Name AS App_Name
4 FROM APPLICATION A
5 JOIN (
6     SELECT App_ID
7     FROM PERMISSION
8     WHERE Permission IN ('Camera', 'Storage')
9     GROUP BY App_ID
10    HAVING COUNT(DISTINCT Permission) = 2
11 ) AS FilteredApps ON A.App_ID = FilteredApps.App_ID
12 ORDER BY A.App_ID;
```

Data Output Messages Notifications

Showing rows: 1 to 4

	app_id [PK] integer	app_name character varying (100)
1	1	ChatMaster
2	4	FitnessPlus
3	7	WeatherNow
4	11	RecipeBook

6. Apps Updated in the Last 2 Years:

```
SELECT DISTINCT A.Name, U.Version, U.Release_Date
FROM APPLICATION A
JOIN UPDATES U ON A.App_ID = U.App_ID
WHERE U.Release_Date >= CURRENT_DATE - INTERVAL '2 years';
```

202301037/202301037@PgSQL Server_202301037

No limit

Query Query History

```
1 SELECT DISTINCT A.Name, U.Version, U.Release_Date
2 FROM APPLICATION A
3 JOIN UPDATES U ON A.App_ID = U.App_ID
4 WHERE U.Release_Date >= CURRENT_DATE - INTERVAL '2 years';
5
```

Data Output Messages Notifications

Showing rows

	name character varying (100)	version character varying (20)	release_date date
1	StudyHelper	1.1	2024-03-01
2	NewsDaily	2.0	2024-01-01
3	MusicStream	3.0	2023-06-01
4	MindRelax	1.2	2024-02-01
5	LanguageLearner	1.3	2023-11-01
6	BudgetBuddy	1.1	2023-10-01
7	WeatherNow	1.2	2023-08-01
8	RecipeBook	1.1	2023-12-01
9	FitnessPlus	1.0.1	2023-05-05
10	TravelGuide	2.0	2023-09-01
11	NoteKeeper	1.1	2023-07-01

Total rows: 11 Query complete 00:00:00.096

7. Most Used Offers:

```
SELECT
    O.Title AS Offer,
    COUNT(*) AS Times_Used
FROM OFFERS O
JOIN OFFERED_TO_1 OT ON O.Offer_ID = OT.Offer_ID
GROUP BY O.Offer_ID, O.Title
ORDER BY Times_Used DESC
LIMIT 5;
```

202301037/202301037@PgSQL Server_202301037

Query Query History

```
1 SELECT
2     O.Title AS Offer,
3     COUNT(*) AS Times_Used
4 FROM OFFERS O
5 JOIN OFFERED_TO_1 OT ON O.Offer_ID = OT.Offer_ID
6 GROUP BY O.Offer_ID, O.Title
7 ORDER BY Times_Used DESC
8 LIMIT 5;
9
```

Data Output Messages Notifications

Showing rows: 1 to

	offer character varying (100)	times_used bigint
1	Student Discount	4
2	Flash Sale	3
3	Cyber Monday	2
4	Summer Sale	1
5	Easter Sale	1

Total rows: 5 Query complete 00:00:00.390

8.Total Revenue earned by Developer:

```
SELECT
    D.Name AS Developer,
    D.Country,
    D.Description,
    SUM(P.Amount) AS Total_Revenue
FROM DEVELOPER D
JOIN CREATED_BY C ON D.Developer_ID = C.Developer_ID
JOIN BUY_APP B ON C.App_ID = B.App_ID
JOIN PURCHASE P ON B.Order_ID = P.Order_ID
GROUP BY D.Developer_ID, D.Name, D.Country, D.Description
ORDER BY Total_Revenue DESC;
```

202301037/202301037@PgSQL Server_202301037

Query Query History

```
1 SELECT
2     D.Name AS Developer,
3     D.Country,
4     D.Description,
5     SUM(P.Amount) AS Total_Revenue
6 FROM DEVELOPER D
7 JOIN CREATED_BY C ON D.Developer_ID = C.Developer_ID
8 JOIN BUY_APP B ON C.App_ID = B.App_ID
9 JOIN PURCHASE P ON B.Order_ID = P.Order_ID
10 GROUP BY D.Developer_ID, D.Name, D.Country, D.Description
11 ORDER BY Total_Revenue DESC;
```

Data Output Messages Notifications

Showing rows: 1 to 10

	developer character varying (100)	country character varying (50)	description text	total_revenue numeric
1	BetaSoft	Canada	Beta testing experts	19.99
2	TechSoft	UK	Innovative solutions	13.99
3	EpsilonApps	Ireland	Epsilon level apps	12.99
4	GammaTech	Australia	Gamma technology	9.99
5	SoftSolutions	Ireland	Reliable software	8.99
6	ThetaApps	Canada	Theta app developm...	7.99
7	DeltaDevs	India	Delta development	5.99
8	Innovatech	USA	Innovative tech	5.99
9	DevOne	USA	Leading app developer	5.99
10	AlphaApps	UK	Alpha quality apps	3.99

Total rows: 10 Query complete 00:00:00.336

9. Search App By Category:

```
SELECT
    A.App_ID,
    A.Name AS App_Name,
    A.Size,
    A.Price,
    A.Number_of_Downloads,
    ROUND(AVG(R.Rating), 2) AS Avg_Rating,
    C.Name AS Category_Name,
    O.Title AS Active_Offer
FROM APPLICATION A
JOIN BELONGS_TO_1 B ON A.App_ID = B.App_ID
JOIN CATEGORY C ON B.Category_ID = C.Category_ID
LEFT JOIN REVIEWS R ON A.App_ID = R.App_ID
LEFT JOIN OFFERED_TO_1 OT ON A.App_ID = OT.App_ID
LEFT JOIN OFFERS O ON OT.Offer_ID = O.Offer_ID AND O.Status = 'Active'
WHERE C.Name = 'Productivity'
GROUP BY A.App_ID, A.Name, A.Size, A.Price, A.Number_of_Downloads, C.Name, O.Title
ORDER BY Avg_Rating DESC NULLS LAST, A.Name;
```

202301037/202301037@PgSQL Server_202301037

Query Query History Scratch Pad x

```
1 SELECT
2     A.App_ID,
3     A.Name AS App_Name,
4     A.Size,
5     A.Price,
6     A.Number_of_Downloads,
7     ROUND(AVG(R.Rating), 2) AS Avg_Rating,
8     C.Name AS Category_Name,
9     O.Title AS Active_Offer
10 FROM APPLICATION A
11 JOIN BELONGS_TO_1 B ON A.App_ID = B.App_ID
12 JOIN CATEGORY C ON B.Category_ID = C.Category_ID
13 LEFT JOIN REVIEWS R ON A.App_ID = R.App_ID
14 LEFT JOIN OFFERED_TO_1 OT ON A.App_ID = OT.App_ID
15 LEFT JOIN OFFERS O ON OT.Offer_ID = O.Offer_ID AND O.Status = 'Active'
16 WHERE C.Name = 'Productivity'
17 GROUP BY A.App_ID, A.Name, A.Size, A.Price, A.Number_of_Downloads, C.Name, O.Title
18 ORDER BY Avg_Rating DESC NULLS LAST, A.Name;
```

Data Output Messages Notifications

Showing rows: 1 to 3 Page No: 1

	app_id integer	app_name character varying (100)	size numeric (10,2)	price numeric (10,2)	number_of_downloads integer	avg_rating numeric	category_name character varying (100)	active_offer character varying (100)
1	1	ChatMaster	50.50	0.00	1003	5.00	Productivity	[null]
2	6	NoteKeeper	30.00	0.00	602	5.00	Productivity	Flash Sale
3	3	GameZone	120.00	2.99	1502	3.00	Productivity	Flash Sale

✓ Successfully run. Total query runtime: 113 msec

Total rows: 3 Query complete 00:00:00.113

10. Search App By name:

```
SELECT * FROM APPLICATION A
WHERE A.name='GameZone' ;
```

202301037/202301037@PgSQL Server_202301037

Query Query History

```
1 SELECT * FROM APPLICATION A
2 WHERE A.name='GameZone' ;
```

Data Output Messages Notifications

Showing rows: 1 to 1

	app_id [PK] integer	name character varying (100)	release_date date	description text	size numeric (10,2)	price numeric (10,2)	number_of_downloads integer
1	3	GameZone	2023-03-15	Arcade games	120.00	2.99	1502

Total rows: 1 Query complete 00:00:00.182

11.Search Book By Title:

```
SELECT * FROM BOOKS B
WHERE B.title='Cooking for Beginners';
```

202301037/202301037@PgSQL Server_202301037

Query Query History

```
1 SELECT * FROM BOOKS B
2 WHERE B.title='Cooking for Beginners';
```

Data Output Messages Notifications

Showing rows: 1 to 1

	book_id [PK] integer	title character varying (100)	price numeric (10,2)	release_date date	publisher character varying (100)	number_of_sold_books integer
1	6	Cooking for Beginners	10.49	2020-06-10	Kitchen Kings	850

Total rows: 1 Query complete 00:00:00.213

✓ Successfully run. Total

***Note : Highlighted are top 3 queries**