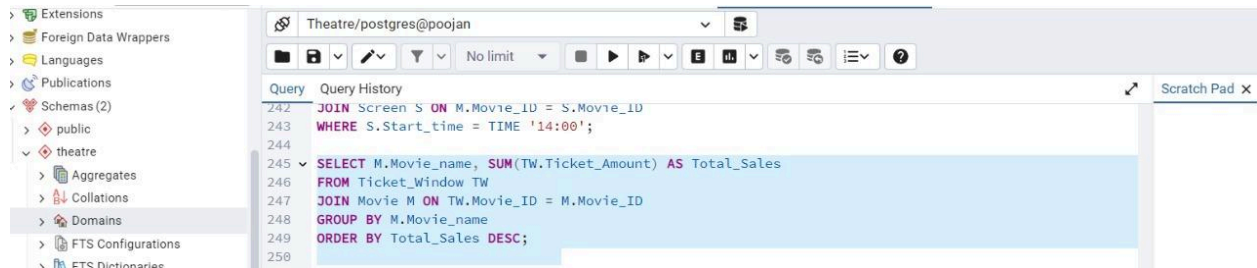


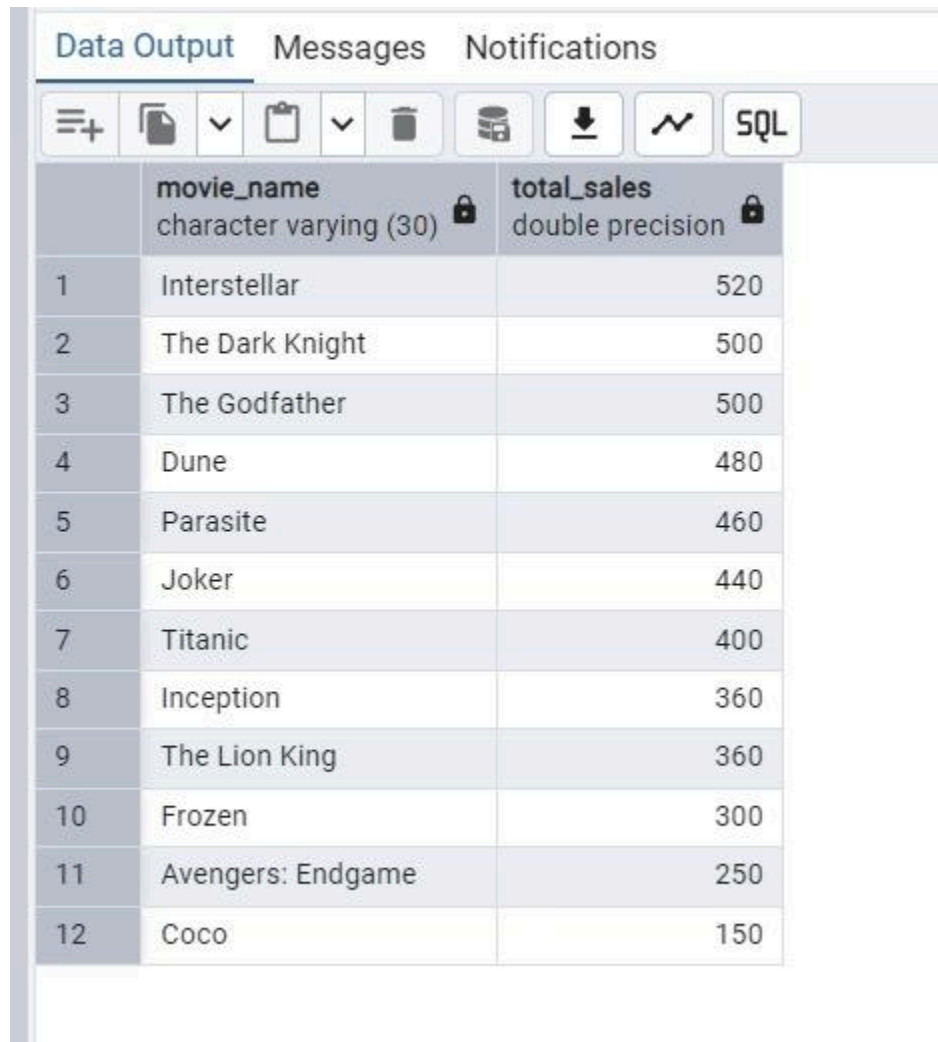
## SQL Queries

### 1. Top Movies Based on Total Ticket Sales:

This query will calculate the total ticket sales for each movie based on the Ticket\_Window table. It then sorts the movies in descending order of ticket sales.



```
242 JOIN Screen S ON M.Movie_ID = S.Movie_ID
243 WHERE S.Start_time = TIME '14:00';
244
245 SELECT M.Movie_name, SUM(TW.Ticket_Amount) AS Total_Sales
246 FROM Ticket_Window TW
247 JOIN Movie M ON TW.Movie_ID = M.Movie_ID
248 GROUP BY M.Movie_name
249 ORDER BY Total_Sales DESC;
250
```



	movie_name character varying (30)	total_sales double precision
1	Interstellar	520
2	The Dark Knight	500
3	The Godfather	500
4	Dune	480
5	Parasite	460
6	Joker	440
7	Titanic	400
8	Inception	360
9	The Lion King	360
10	Frozen	300
11	Avengers: Endgame	250
12	Coco	150

## 2. Canteen Product with the Most Sales (by Quantity):

This query calculates the product with the most sales by quantity from the Bill\_Invoice and Canteen tables.

```
248 GROUP BY M.Movie_name
249 ORDER BY Total_Sales DESC;
250
251 SELECT C.Product_Name, SUM(B.Product_Quantity) AS Total_Sold
252 FROM Bill_Invoice B
253 JOIN Canteen C ON B.Product_ID = C.Product_ID
254 GROUP BY C.Product_Name
255 ORDER BY Total_Sold DESC
256 LIMIT 1;
257
258
```

	product_name character varying (30)	total_sold bigint
1	Popcorn	2

## 3. Top 5 Movies With Highest Ratings and Their Duration:

This query will display the top 5 highest-rated movies along with their durations.

```
257
258 SELECT M.Movie_name, M.Rating, M.Duration
259 FROM Movie M
260 ORDER BY M.Rating DESC
261 LIMIT 5;
262
```

	Data Output	Messages	Notifications
	<div> <div>≡</div> <div>+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🗄️</div> <div>⬇️</div> <div>📈</div> <div>SQL</div> </div>		
	movie_name character varying (30) 🔒	rating numeric 🔒	duration integer 🔒
1	Avengers: Endgame	9.5	181
2	The Godfather	9.2	175
3	The Dark Knight	9.1	152
4	Inception	9.0	148
5	Coco	9.0	105

#### 4. Average Ticket Price for Each Screen:

This query calculates the average ticket price for each screen.

```

260 ORDER BY M.Rating DESC
261 LIMIT 5;
262 ✓ SELECT S.Screen_Name, AVG(TW.Ticket_Amount) AS Average_Ticket_Price
263 FROM Ticket_Window TW
264 JOIN Screen S ON TW.Screen_No = S.Screen_No
265 GROUP BY S.Screen_Name;
266
267

```

Data Output Messages Notifications		
<div> <div>≡ +</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🗄️</div> <div>⬇️</div> <div>📈</div> <div>SQL</div> </div>		
	screen_name character varying (30) 🔒	average_ticket_price double precision 🔒
1	sky View	235
2	Ocean View	220
3	Kids Corner	162
4	Main Hall	254
5	Dream Zone	205

### 5. Seats Occupancy Rate per Screen:

This query will calculate the seat occupancy rate per screen by comparing the total number of booked seats to the total number of seats available.

```

1 (324, 214, 5, 270.0, '2025-04-15', 505);
2 ✓ SELECT S.Screen_Name,
3       (SUM(Seats.Booked_seats) * 100.0 / SUM(Seats.Total_Seats)) AS Occupancy_Rate
4 FROM Seats
5 JOIN Screen S ON Seats.Screen_No = S.Screen_No
6 GROUP BY S.Screen_Name;
7

```

Data Output Messages Notifications		
<div> <div>≡ +</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🗄️</div> <div>⬇️</div> <div>📈</div> <div>SQL</div> </div>		
	screen_name character varying (30) 🔒	occupancy_rate numeric 🔒
1	Kids Corner	85.000000000000000000
2	Main Hall	81.5151515151515152

