

Top-3 Queries of the Project

1. Users that got discount in subscription payment: -

- It finds the user that have got any kind of discount on payment of the subscription plan. We just join the payment_details_table and the table having information about the different subscription plans with user table, and we just find out the tuples where the paid charges by user is less than the charge of a particular subscription plan

```
select spd.userid, u.user_name, spd.tier, spd.paid_amount,  
st.charges from
```

```
subscription_payment_details spd natural join  
subscription_tier st join users u on
```

```
spd.userid = u.userid where spd.paid_amount < st.charges;
```

	userid character varying (10) 🔒	user_name character varying (30) 🔒	tier character varying (10) 🔒	paid_amount integer 🔒	charges integer 🔒
1	U003	Charlie Lee	Gold	270	300
2	U026	Zara Ahmed	Gold	270	300
3	U033	Grace Lee	Gold	270	300
4	U058	Felipe Morales	Gold	270	300

2. List active subscribers whose subscription will expire within the next 50 days.

- The query finds the users with the corresponding important details whose subscription is going to expire within the next 50 days by accumulating the days remaining from the subscription start date stored in subscription_payment_details_table and joins it with the category of subscription from subscription_tier table to find out the duration of validity for corresponding plan, that if the accumulated duration is withing next 50 days then show those users.

```
SELECT spd.UserID, u.User_Name, spd.Subscription_Start_Date,  
st.Duration_In_Months,
```

```
(spd.Subscription_Start_Date + (st.Duration_In_Months *  
INTERVAL '1 month')) AS
```

```
Expiry_Date FROM Subscription_Payment_Details spd JOIN  
Subscription_Tier st ON
```

```

spd.Tier = st.Tier JOIN Users u ON u.UserID = spd.UserID WHERE
(spd.Subscription_Start_Date + (st.Duration_In_Months *
INTERVAL '1 month')) BETWEEN

CURRENT_DATE AND CURRENT_DATE + INTERVAL '50 day';

```

	userid character varying (10)	user_name character varying (30)	subscription_start_date date	duration_in_months integer	expiry_date timestamp without time zone
1	U015	Oscar Ruiz	2024-05-18	12	2025-05-18 00:00:00
2	U048	Victor Hugo	2024-11-03	6	2025-05-03 00:00:00

3. Search in general: -

- It allows us to search the given keyword from some set of attributes which may have the same value in them. For example, we can think of searching a keyword “baby john”, then it can be any movie or TV shows name, it can be person’s name, etc. So, the query will give all the relevant results as output.

```

SELECT DISTINCT C.Title, SC.Episode_Name FROM Contents C LEFT
JOIN

```

```

SubContent SC ON C.ContentID = SC.ContentID LEFT JOIN Genre G
ON C.ContentID =

```

```

G.ContentID LEFT JOIN Casts CS ON C.ContentID = CS.ContentID
AND SC.SubContentID =

```

```

CS.SubContentID WHERE C.Title ILIKE '%Carlos Mendes%' OR
SC.Episode_Name ILIKE '%Carlos

```

```

Mendes%' OR G.Genrename ILIKE '%Carlos Mendes%' OR CS.CastName
ILIKE '%carlos Mendes%'

```

	title character varying (20)	episode_name character varying (50)
1	Whispers	Whispers: Voices Beyond
2	Voices Rising	Voices Amplified
3	Vision Tales	Clarity Found
4	Chat Lounge	Welcome Chat
5	Wave Walkers	The Ripple
6	Signal Storm	Signal Lost
7	The Final Frame	Blurry Truth
8	Stream Force	Force Unleashed
9	Game Zone	Level Up
10	Stream Force	Force Awakens
11	Mind Matters	Inside Logic
12	Darkstream	Dark Beginnings
13	Bass Booster	Bass Drops
14	The Final Frame	Snap Start
15	Blade Shift	Edge of Blade
16	Wave Walkers	The Deep Dive
17	Echo Chamber	Echo Chamber: The Feedback Loop
18	Vision Tales	Vision Start
19	SciWorld	Beyond Reality
20	Byte Beats	Byte Beats: The Surge
21	Darkstream	Dark Depths