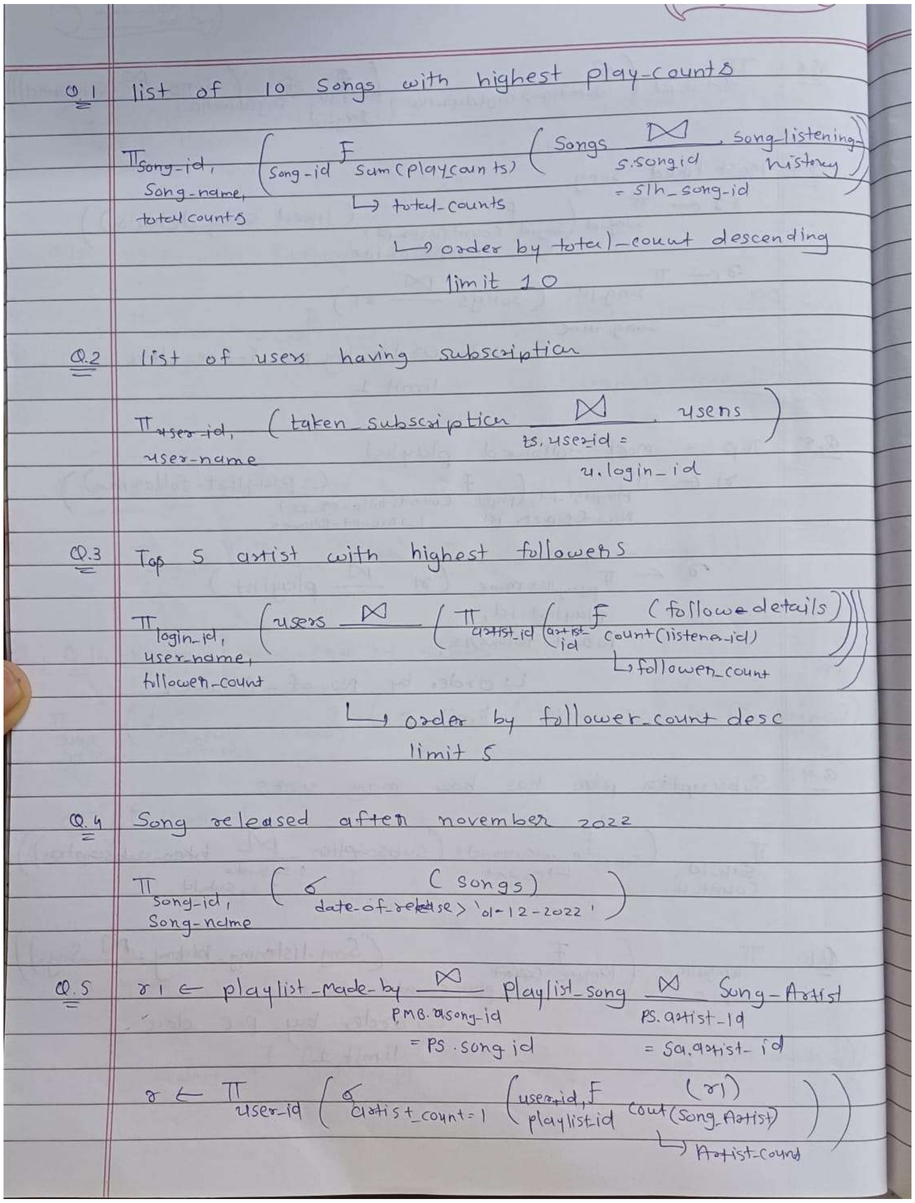
Group 3 DBMS project

Music_Player_system

List of Queries:

- **Relational Algebra**



Q.1 list of 10 Songs with highest play-counts

$$\Pi_{Song\_id,\ Song\_name,\ total\ counts}\left({}_{Song\_id}\mathcal{F}_{sum(playcounts) \hookrightarrow total\_counts}\left(Songs \bowtie_{s.songid = slh\_song\_id} song\_listening\_history\right)\right)$$

↳ order by total_count descending
limit 10

Q.2 list of users having subscription

$$\Pi_{~~user\_id~~,\ user\_name}\left(taken\_subscription \bowtie_{ts.userid = u.login\_id} users\right)$$

Q.3 Top 5 artist with highest followers

$$\Pi_{login\_id,\ user\_name,\ follower\_count}\left(users \bowtie \left(\Pi_{artist\_id}\left({}_{artist\_id}\mathcal{F}_{count(listener\_id) \hookrightarrow follower\_count}(followdetails)\right)\right)\right)$$

↳ order by follower_count desc
limit 5

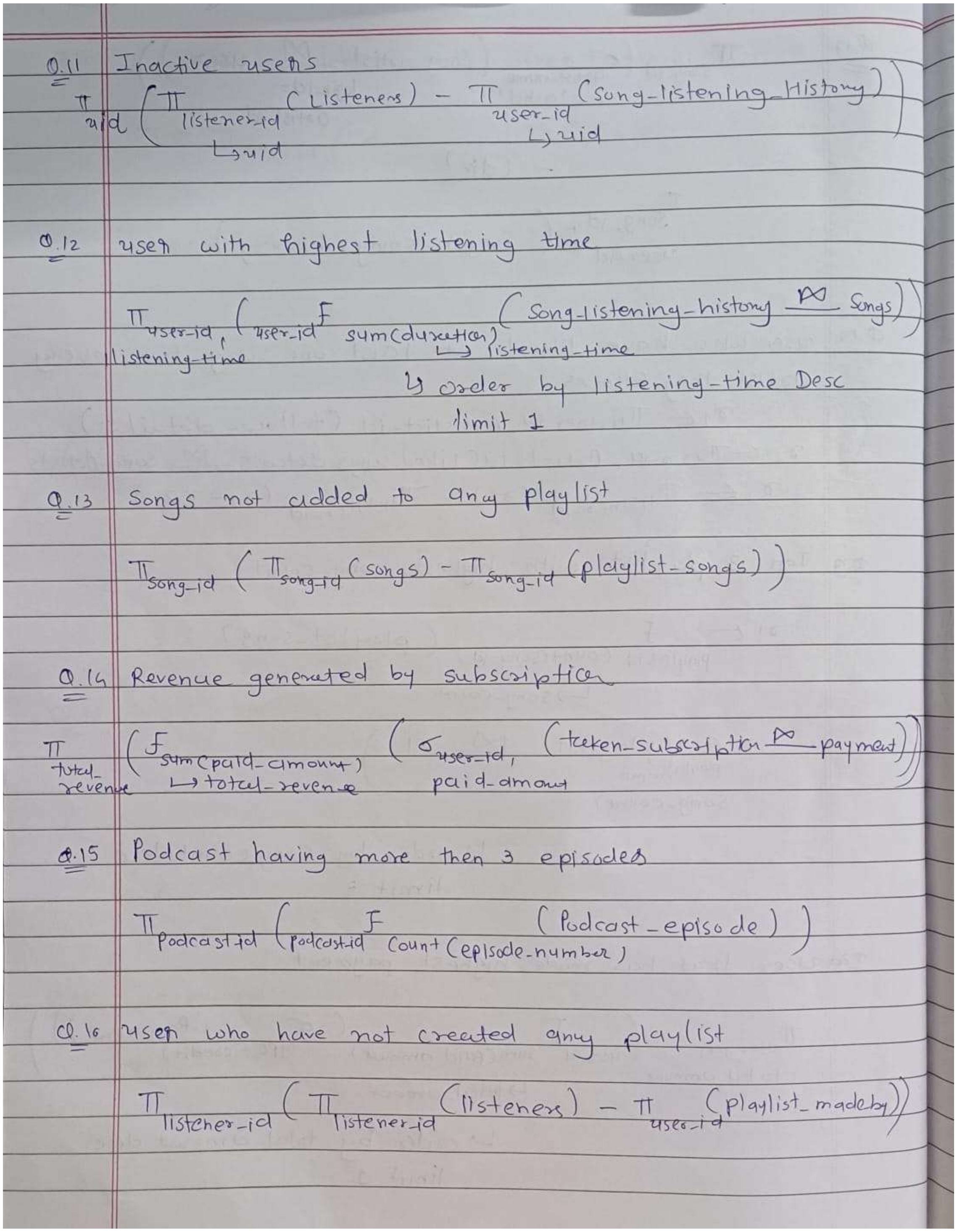
Q.4 Song released after november 2022

$$\Pi_{song\_id,\ Song\_name}\left(\sigma_{date\_of\_release > '01-12-2022'}(songs)\right)$$

Q.5

$$r1 \leftarrow playlist\_made\_by \bowtie_{PMB.song\_id = PS.song\ id} Playlist\_song \bowtie_{PS.artist\_id = Sa.artist\_id} Song\_Artist$$

$$r \leftarrow \Pi_{user\_id}\left(\sigma_{artist\_count=1}\left({}_{user\_id,\ playlist\_id}\mathcal{F}_{count(Song\_Artist) \hookrightarrow Artist\_count}(r1)\right)\right)$$



Q.11 Inactive users

$$\pi_{uid}\left(\pi_{\underset{\hookrightarrow uid}{\text{listener\_id}}}(\text{Listeners}) - \pi_{\underset{\hookrightarrow uid}{\text{user\_id}}}(\text{Song-listening\_History})\right)$$

Q.12 user with highest listening time

$$\pi_{\text{user\_id},\ \text{listening-time}}\left({}_{\text{user\_id}}\mathcal{F}_{\underset{\hookrightarrow \text{listening\_time}}{\text{sum(duration)}}}(\text{Song-listening-history} \bowtie \text{Songs})\right)$$

$\hookrightarrow$ order by listening-time Desc

limit 1

Q.13 Songs not added to any playlist

$$\pi_{\text{song\_id}}\left(\pi_{\text{song\_id}}(\text{songs}) - \pi_{\text{song\_id}}(\text{playlist\_songs})\right)$$

Q.14 Revenue generated by subscription

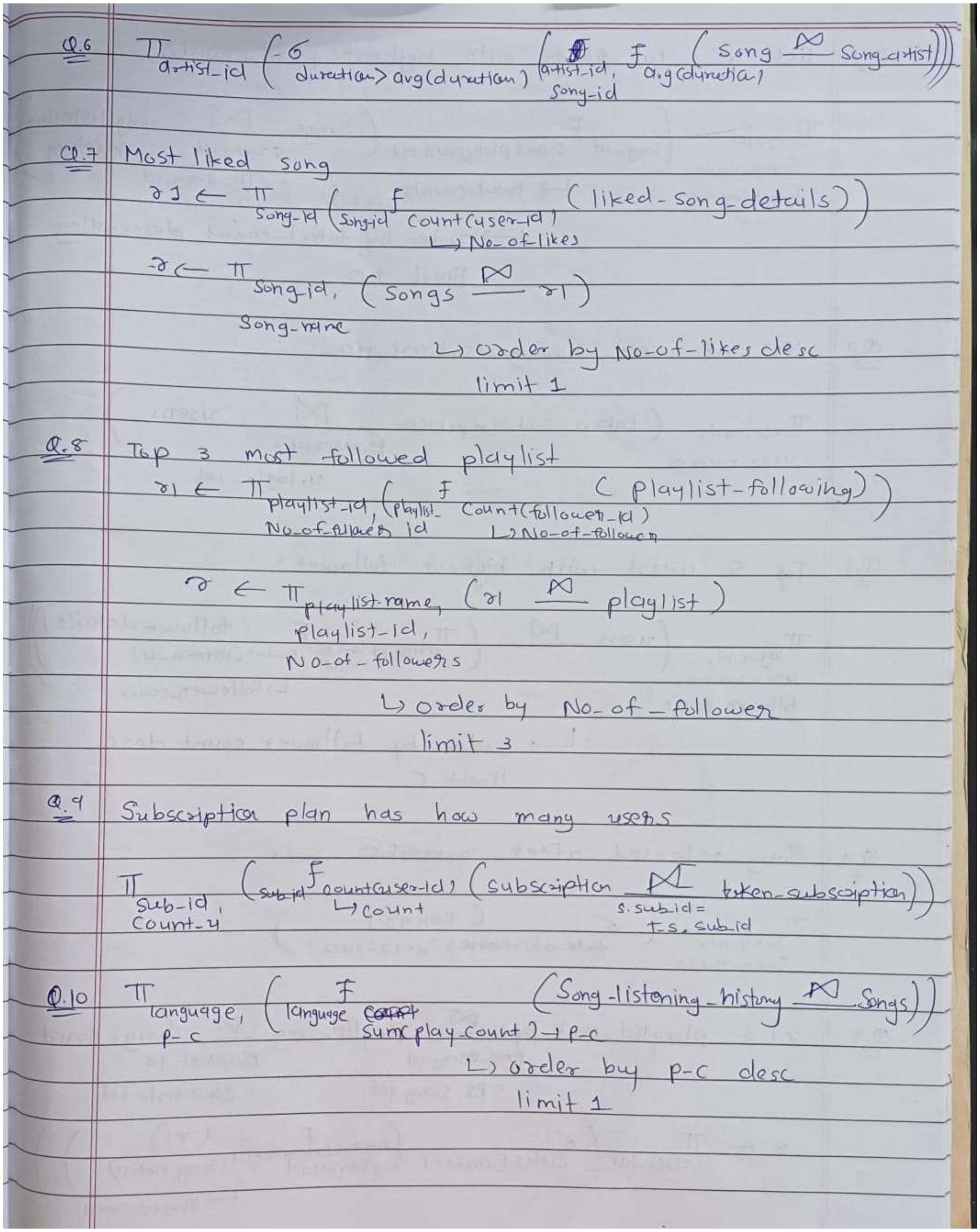
$$\pi_{\text{total\_revenue}}\left(\mathcal{F}_{\underset{\hookrightarrow \text{total\_revenue}}{\text{sum(paid\_amount)}}}\left(\sigma_{\text{user\_id},\ \text{paid\_amount}}(\text{taken-subscription} \bowtie \text{payment})\right)\right)$$

Q.15 Podcast having more then 3 episodes

$$\pi_{\text{Podcast\_id}}\left({}_{\text{podcast\_id}}\mathcal{F}_{\text{Count(episode\_number)}}(\text{Podcast\_episode})\right)$$

Q.16 user who have not created any playlist

$$\pi_{\text{listener\_id}}\left(\pi_{\text{listener\_id}}(\text{listeners}) - \pi_{\text{user\_id}}(\text{playlist\_madeby})\right)$$



Q.6 $\pi_{\text{artist\_id}} \left(\sigma_{\text{duration} > \text{avg(duration)}} \left({}_{\text{artist\_id, song\_id}} \mathcal{F}_{\text{avg(duration)}} \left(\text{song} \bowtie \text{Song\_artist} \right) \right) \right)$

Q.7 Most liked song

$r1 \leftarrow \pi_{\text{Song\_id}} \left({}_{\text{Song\_id}} \mathcal{F}_{\text{count(user\_id)} \to \text{No\_of\_likes}} \left(\text{liked\_song\_details} \right) \right)$

$r \leftarrow \pi_{\text{Song\_id, Song\_name}} \left(\text{Songs} \bowtie r1 \right)$

↳ order by No-of-likes desc
limit 1

Q.8 Top 3 most followed playlist

$r1 \leftarrow \pi_{\text{playlist\_id, No\_of\_followers}} \left({}_{\text{playlist\_id}} \mathcal{F}_{\text{count(follower\_id)} \to \text{No\_of\_follower}} \left(\text{playlist\_following} \right) \right)$

$r \leftarrow \pi_{\text{playlist\_name, playlist\_id, No\_of\_followers}} \left(r1 \bowtie \text{playlist} \right)$

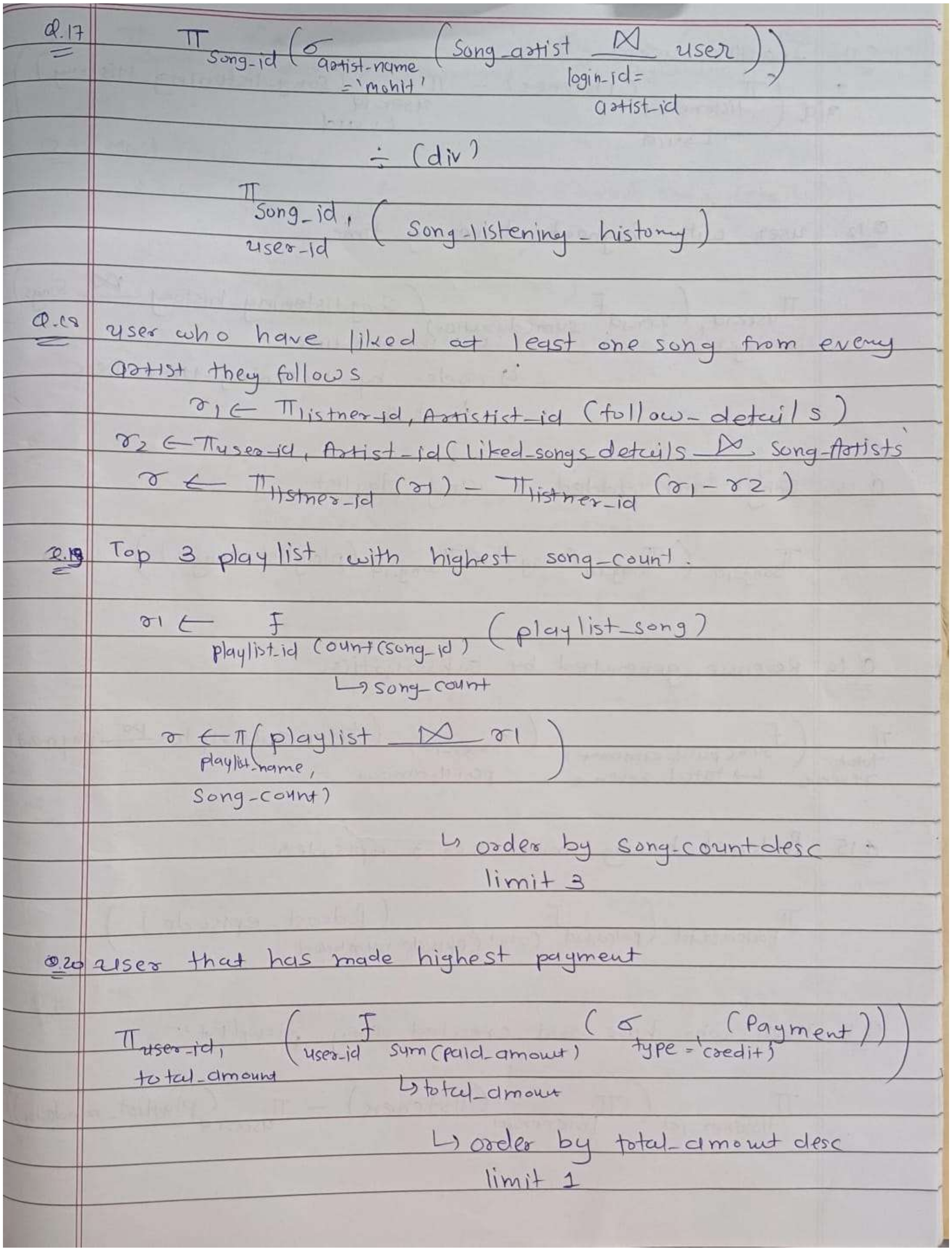
↳ order by No-of-follower
limit 3

Q.9 Subscription plan has how many users

$\pi_{\text{sub\_id, count\_u}} \left({}_{\text{sub\_id}} \mathcal{F}_{\text{count(user\_id)} \to \text{count}} \left(\text{subscription} \bowtie_{\text{s.sub\_id = t.s.sub\_id}} \text{taken\_subscription} \right) \right)$

Q.10 $\pi_{\text{language, p\_c}} \left({}_{\text{language}} \mathcal{F}_{\cancel{\text{count}}\ \text{sum(play\_count)} \to \text{p\_c}} \left(\text{Song\_listening\_history} \bowtie \text{Songs} \right) \right)$

↳ order by p-c desc
limit 1



Q.17

$$\pi_{Song\_id}\left(\sigma_{artist\text{-}name='mohit'}\left(Song\_artist \underset{login\_id=artist\_id}{\bowtie} user\right)\right)$$

$$\div \text{ (div)}$$

$$\pi_{Song\_id,\ user\_id}\left(Song\_listening\_history\right)$$

Q.18 user who have liked at least one song from every artist they follows

$$r_1 \leftarrow \pi_{listner\_id,\ Artistist\_id}(follow\_details)$$

$$r_2 \leftarrow \pi_{user\_id,\ Artist\_id}(Liked\_songs\_details \bowtie Song\_Artists$$

$$r \leftarrow \pi_{listner\_id}(r_1) - \pi_{listner\_id}(r_1 - r_2)$$

Q.19 Top 3 playlist with highest song-count.

$$r_1 \leftarrow {}_{playlist\_id}\mathcal{F}_{\underbrace{count(song\_id)}_{\hookrightarrow song\_count}}(playlist\_song)$$

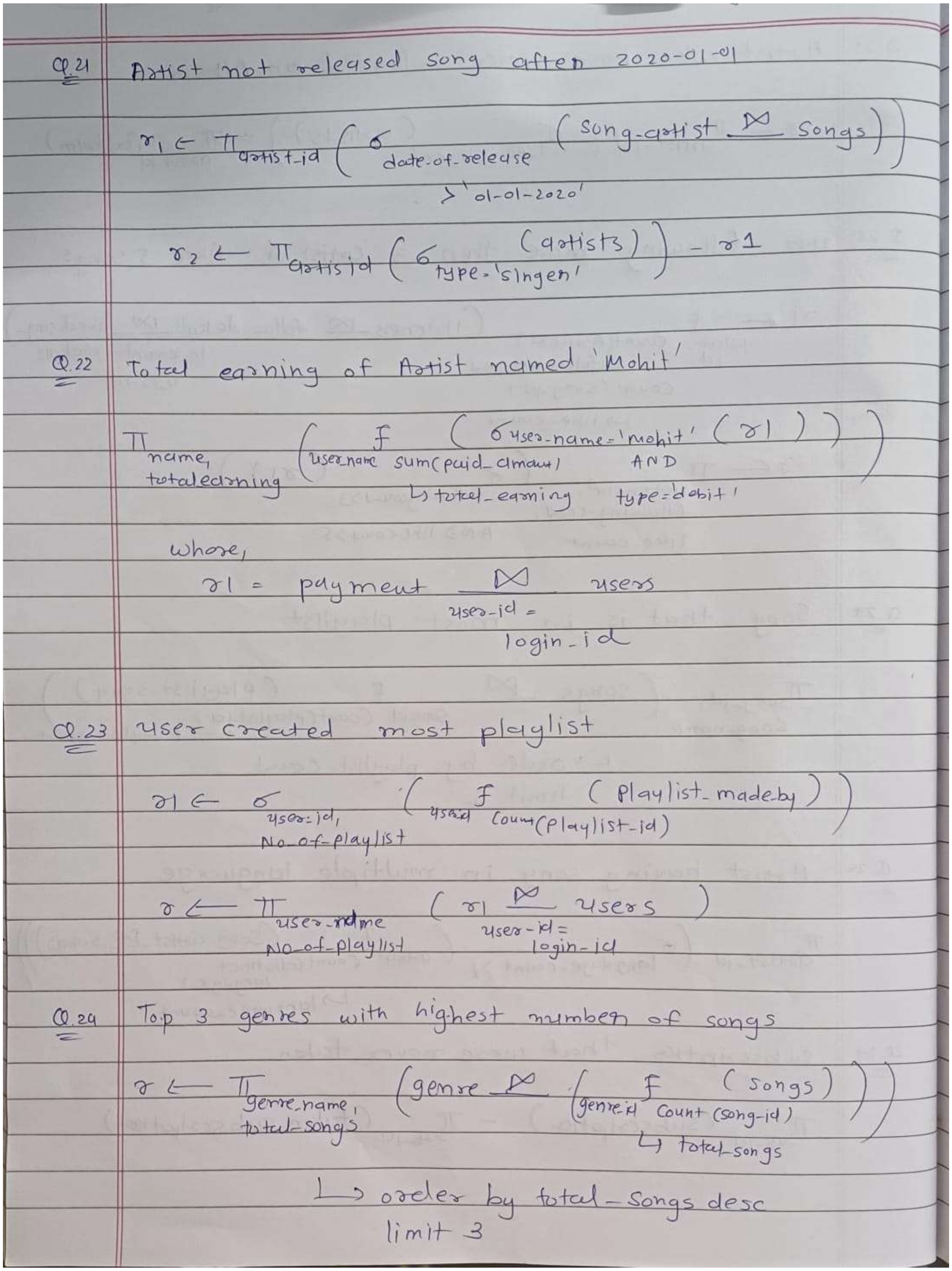
$$r \leftarrow \pi_{playlist\_name,\ Song\_count}(playlist \bowtie r_1)$$

↳ order by song-count desc
limit 3

Q.20 user that has made highest payment

$$\pi_{user\_id,\ total\_amount}\left({}_{user\_id}\mathcal{F}_{\underbrace{sum(paid\_amount)}_{\hookrightarrow total\_amount}}\left(\sigma_{type='credit'}(Payment)\right)\right)$$

↳ order by total_amount desc
limit 1



Q.21 Artist not released song after 2020-01-01

$r_1 \leftarrow \Pi_{\text{artist\_id}}\left(\sigma_{\text{date-of-release} > '01-01-2020'}(\text{song-artist} \bowtie \text{songs})\right)$

$r_2 \leftarrow \Pi_{\text{artisid}}\left(\sigma_{\text{type='singer'}}(\text{artists})\right) - r1$

Q.22 Total earning of Artist named 'Mohit'

$\Pi_{\text{name, totalearning}}\left({}_{\text{user\_name}}\mathcal{F}_{\text{sum(paid\_amount)} \rightarrow \text{total\_earning}}\left(\sigma_{\text{user\_name='mohit' AND type='debit'}}(r1)\right)\right)$

where,

$r1 = \text{payment} \bowtie_{\text{user\_id = login\_id}} \text{users}$

Q.23 user created most playlist

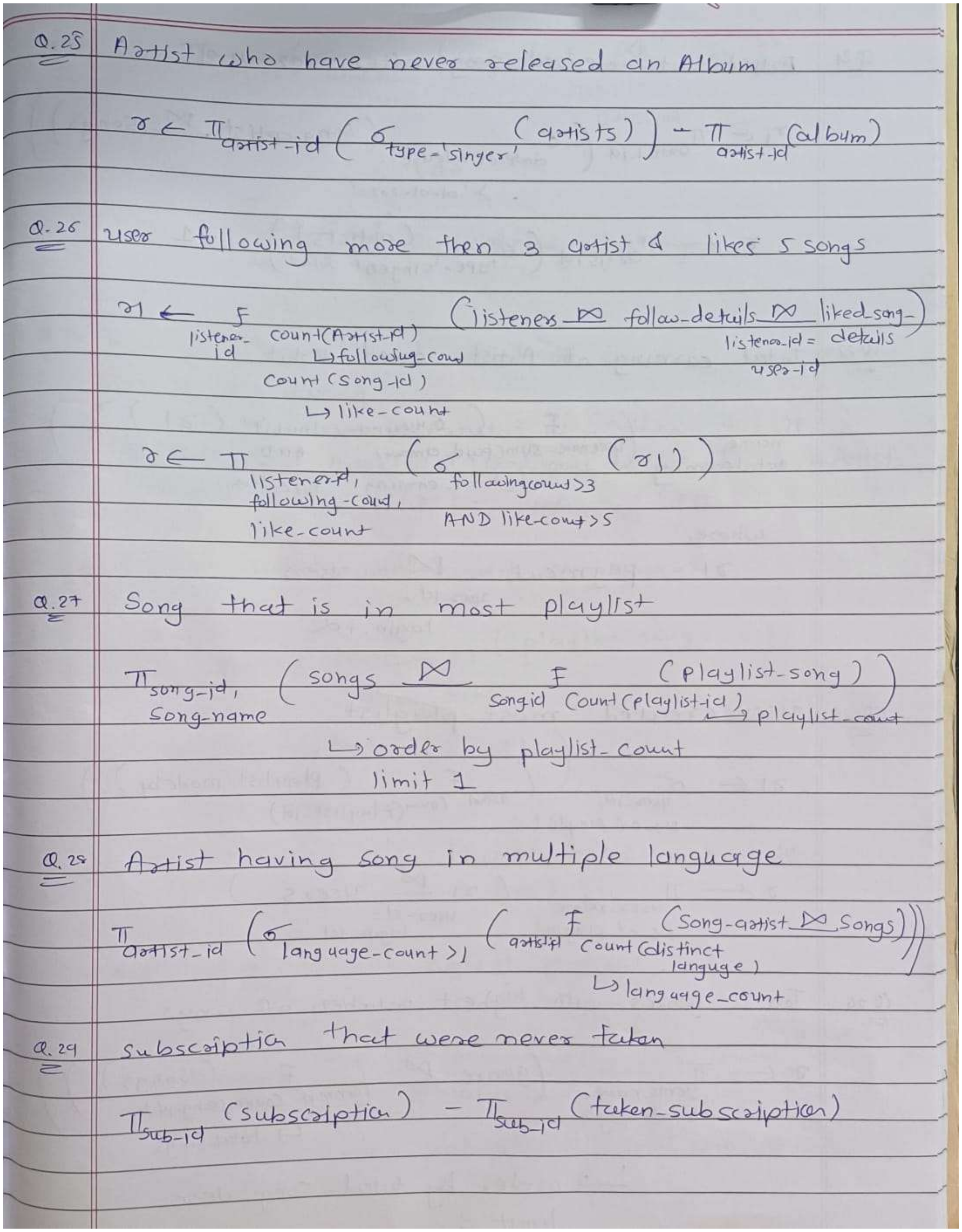
$r1 \leftarrow \sigma_{\text{user\_id, No\_of\_playlist}}\left({}_{\text{userid}}\mathcal{F}_{\text{count(playlist\_id)}}(\text{playlist\_madeby})\right)$

$r \leftarrow \Pi_{\text{user\_name, No\_of\_playlist}}\left(r1 \bowtie_{\text{user\_id = login\_id}} \text{users}\right)$

Q.24 Top 3 genres with highest number of songs

$r \leftarrow \Pi_{\text{genre\_name, total\_songs}}\left(\text{genre} \bowtie \left({}_{\text{genre\_id}}\mathcal{F}_{\text{count(song\_id)} \rightarrow \text{total\_songs}}(\text{songs})\right)\right)$

↳ order by total_songs desc limit 3



Q.25 Artist who have never released an Album

$$r \leftarrow \pi_{artist\text{-}id}\left(\sigma_{type='singer'}(artists)\right) - \pi_{artist\text{-}id}(album)$$

Q.26 user following more then 3 artist & likes 5 songs

$$r1 \leftarrow {}_{listener\_id}\mathcal{F}_{count(Artist\text{-}id) \rightarrow following\text{-}count,\ count(song\text{-}id) \rightarrow like\text{-}count}\left(listeners \bowtie follow\text{-}details \bowtie_{listener\_id = details.user\text{-}id} liked\_song\right)$$

$$r \leftarrow \pi_{listener\_id,\ following\text{-}count,\ like\text{-}count}\left(\sigma_{following count>3 \text{ AND } like\text{-}count>5}(r1)\right)$$

Q.27 Song that is in most playlist

$$\pi_{song\_id,\ song\text{-}name}\left(songs \bowtie {}_{songid}\mathcal{F}_{Count(playlist\text{-}id) \rightarrow playlist\_count}(playlist\text{-}song)\right)$$

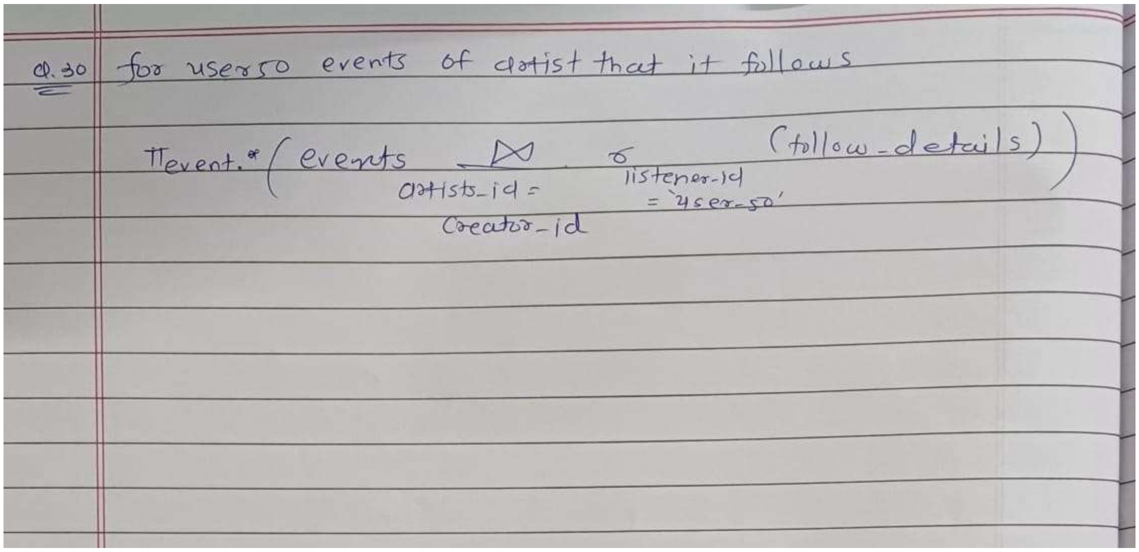
↳ order by playlist_count
limit 1

Q.28 Artist having song in multiple language

$$\pi_{artist\_id}\left(\sigma_{language\text{-}count>1}\left({}_{artistid}\mathcal{F}_{Count(distinct\ languge) \rightarrow language\_count}(Song\text{-}artist \bowtie Songs)\right)\right)$$

Q.29 subscription that were never taken

$$\pi_{sub\text{-}id}(subscription) - \pi_{sub\_id}(taken\text{-}subscription)$$



Q.30 for user50 events of artist that it follows

$\pi_{event.*}\left(events \bowtie_{artists\_id = creator\_id} \sigma_{listener\text{-}id = 'user\text{-}50'} (follow\_details) \right)$

SQL :

1.list of songs with total playcounts

SELECT S.SONG_ID,S.SONG_NAME,SUM(PLAY_COUNT) AS TOTAL_COUNT

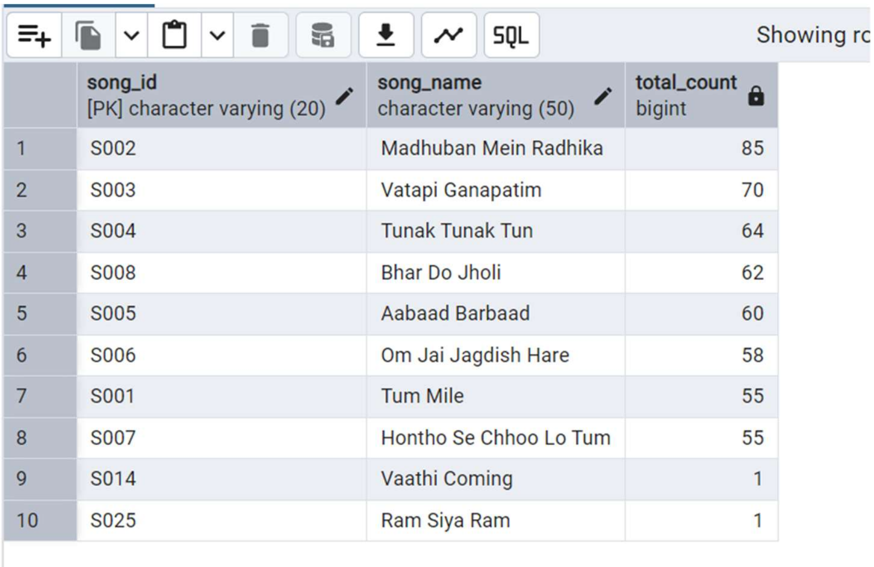
FROM SONGS AS S

NATURAL JOIN SONG_LISTENING_HISTORY AS SLH

GROUP BY SONG_ID

ORDER BY TOTAL_COUNT DESC

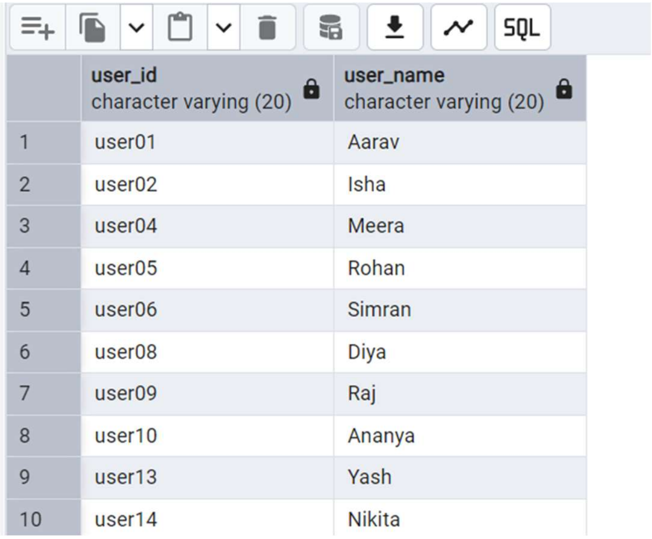
LIMIT 10;



	song_id [PK] character varying (20)	song_name character varying (50)	total_count bigint
1	S002	Madhuban Mein Radhika	85
2	S003	Vatapi Ganapatim	70
3	S004	Tunak Tunak Tun	64
4	S008	Bhar Do Jholi	62
5	S005	Aabaad Barbaad	60
6	S006	Om Jai Jagdish Hare	58
7	S001	Tum Mile	55
8	S007	Hontho Se Chhoo Lo Tum	55
9	S014	Vaathi Coming	1
10	S025	Ram Siya Ram	1

2. List of users having subscription

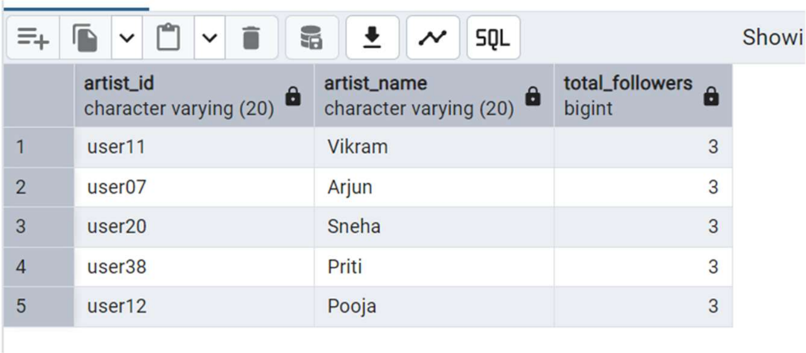
```
SELECT USER_ID,USER_NAME

FROM TAKEN_SUBSCRIPTION AS TS

JOIN USERS AS U

ON TS.USER_ID = U.LOGIN_ID;
```



	user_id character varying (20)	user_name character varying (20)
1	user01	Aarav
2	user02	Isha
3	user04	Meera
4	user05	Rohan
5	user06	Simran
6	user08	Diya
7	user09	Raj
8	user10	Ananya
9	user13	Yash
10	user14	Nikita

3. Top 5 artists with highest followers.

```
SELECT LOGIN_ID,USER_NAME, FOLLOWER_COUNT

FROM USERS AS U JOIN

(SELECT ARTIST_ID,COUNT(LISTENER_ID) AS FOLLOWER_COUNT

FROM FOLLOW_DETAILS

GROUP BY ARTIST_ID) AS R

ON R.ARTIST_ID = U.LOGIN_ID

ORDER BY FOLLOWER_COUNT DESC

LIMIT 5;
```



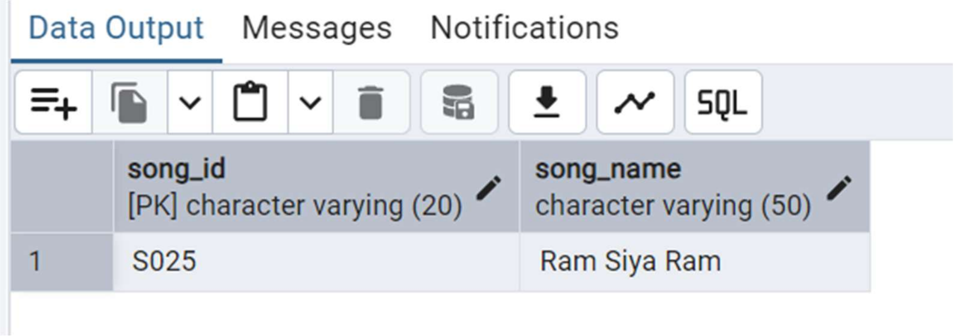
	artist_id character varying (20)	artist_name character varying (20)	total_followers bigint
1	user11	Vikram	3
2	user07	Arjun	3
3	user20	Sneha	3
4	user38	Priti	3
5	user12	Pooja	3

4. Song released after november 2022

SELECT SONG_ID,SONG_NAME

FROM SONGS

WHERE DATE_OF_RELEASE >= '01-12-2022';



	song_id [PK] character varying (20)	song_name character varying (50)
1	S025	Ram Siya Ram

5. Give the name of the user having a playlist with only a single singer's song.

SELECT USER_ID

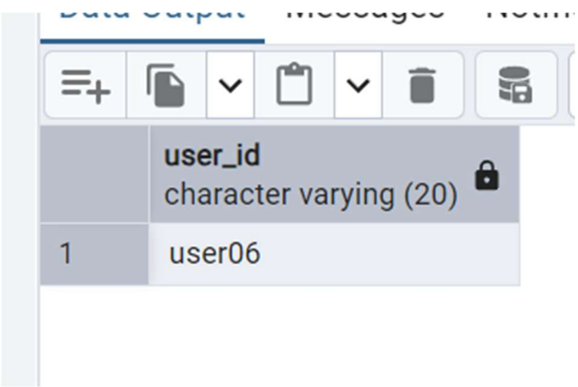
FROM PLAYLIST_MADE_BY AS PMB

JOIN PLAYLIST_SONG AS PS ON PMB.PLAYLIST_ID = PS.PLAYLIST_ID

JOIN SONG_ARTISTS AS SA ON PS.SONG_ID = SA.SONG_ID

GROUP BY USER_ID, PS.PLAYLIST_ID

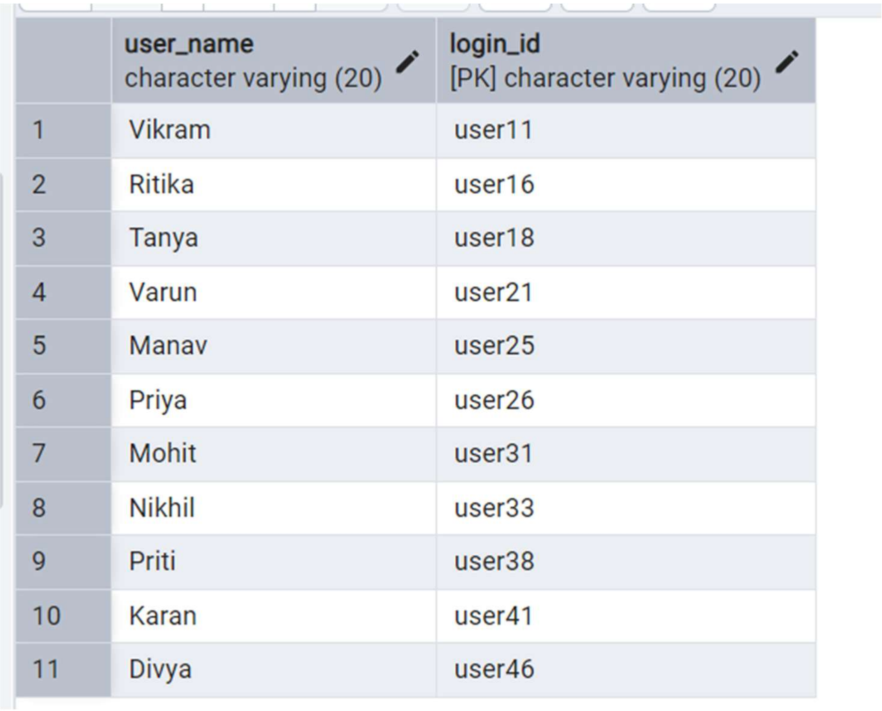
HAVING COUNT(DISTINCT ARTIST_ID) = 1;



	user_id character varying (20)
1	user06

6.Artist with longer song duration than average song duration

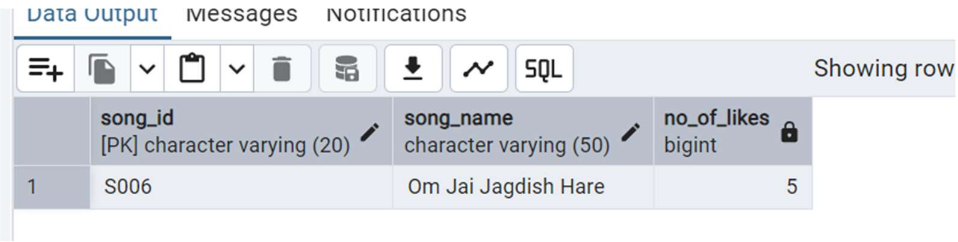
```
SELECT USER_NAME,LOGIN_ID

FROM USERS AS U JOIN

(SELECT DISTINCT ARTIST_ID

FROM SONGS NATURAL JOIN SONG_ARTISTS

GROUP BY ARTIST_ID,SONG_ID

HAVING DURATION > (SELECT AVG(DURATION) FROM SONGS)) AS R

ON U.LOGIN_ID = R.ARTIST_ID;
```



	user_name character varying (20)	login_id [PK] character varying (20)
1	Vikram	user11
2	Ritika	user16
3	Tanya	user18
4	Varun	user21
5	Manav	user25
6	Priya	user26
7	Mohit	user31
8	Nikhil	user33
9	Priti	user38
10	Karan	user41
11	Divya	user46

7.Most liked song

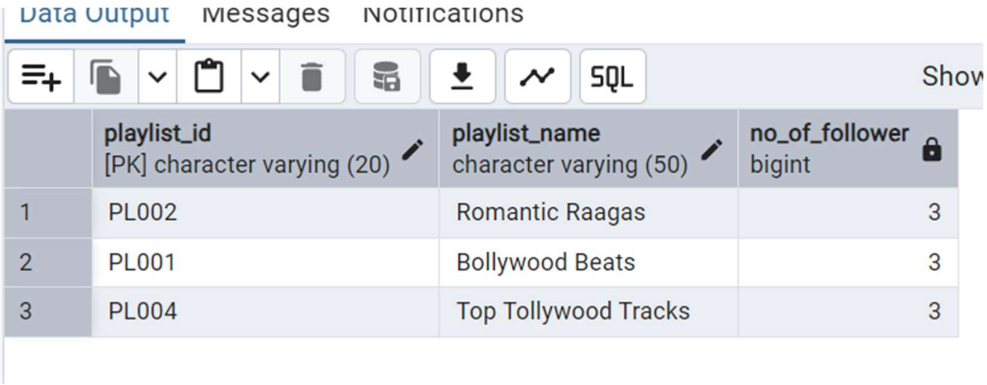
```
SELECT SONG_ID,SONG_NAME,NO_OF_LIKES

FROM SONGS NATURAL JOIN

(SELECT SONG_ID,COUNT(USER_ID) AS NO_OF_LIKES

FROM LIKED_SONGS_DETAILS

GROUP BY SONG_ID

ORDER BY NO_OF_LIKES DESC

LIMIT 1) AS R;
```



	song_id [PK] character varying (20)	song_name character varying (50)	no_of_likes bigint
1	S006	Om Jai Jagdish Hare	5

8. Top 3 most followed playlist

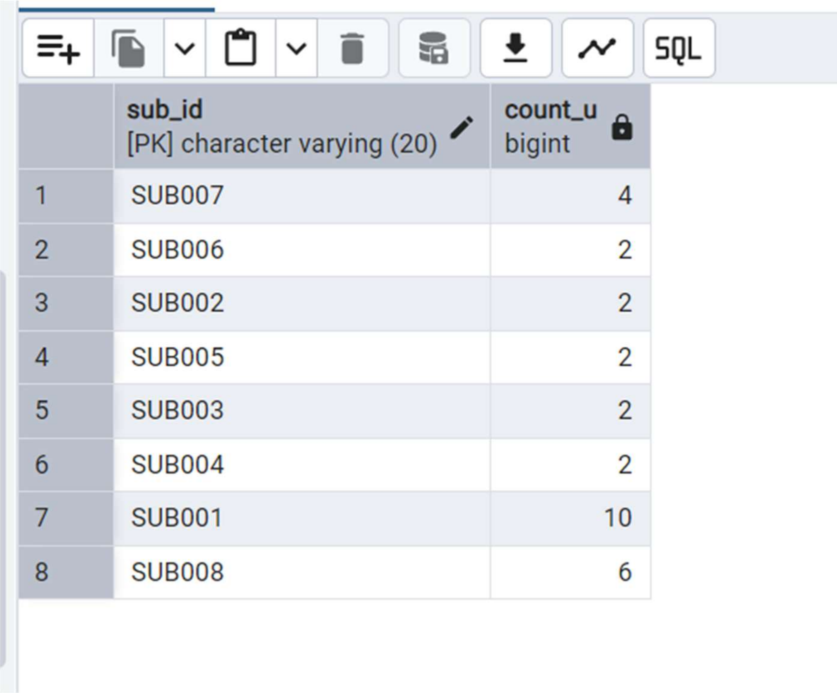
```
SELECT PLAYLIST_ID,PLAYLIST_NAME,NO_OF_FOLLOWER

FROM PLAYLIST NATURAL JOIN

(SELECT PLAYLIST_ID, COUNT(FOLLOWER_ID) AS NO_OF_FOLLOWER

FROM PLAYLIST_FOLLOWING

GROUP BY PLAYLIST_ID) AS R

ORDER BY NO_OF_FOLLOWER DESC

LIMIT 3;
```



	playlist_id [PK] character varying (20)	playlist_name character varying (50)	no_of_follower bigint
1	PL002	Romantic Raagas	3
2	PL001	Bollywood Beats	3
3	PL004	Top Tollywood Tracks	3

9.List of each Subscription plan with number of users

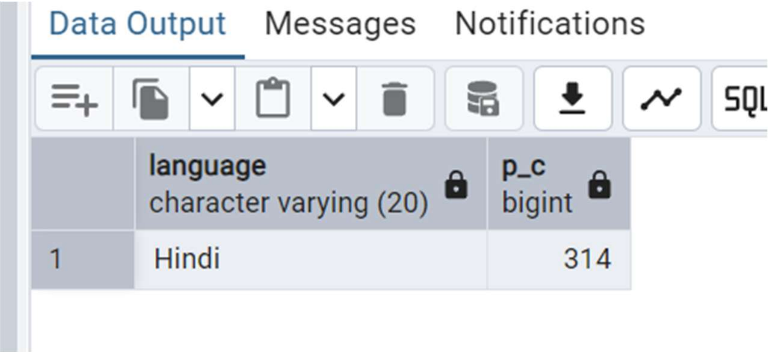
```
SELECT S.SUB_ID,COUNT(USER_ID) AS COUNT_U

FROM SUBSCRIPTION AS S LEFT JOIN

TAKEN_SUBSCRIPTION AS TS

ON S.SUB_ID = TS.SUB_ID

GROUP BY S.SUB_ID;
```



	sub_id [PK] character varying (20)	count_u bigint
1	SUB007	4
2	SUB006	2
3	SUB002	2
4	SUB005	2
5	SUB003	2
6	SUB004	2
7	SUB001	10
8	SUB008	6

10.Most played language (by song_count)

```
SELECT S.LANGUAGE,SUM(PLAY_COUNT) AS P_C

FROM SONG_LISTENING_HISTORY

NATURAL JOIN SONGS AS S

GROUP BY S.LANGUAGE

ORDER BY P_C DESC

LIMIT 1;
```



	language character varying (20)	p_c bigint
1	Hindi	314

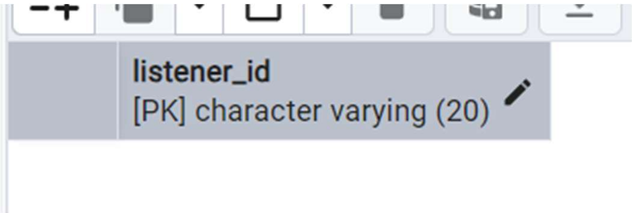
11.**Inactive user**

SELECT LISTENER_ID

FROM LISTENERS WHERE LISTENER_ID

NOT IN(SELECT DISTINCT USER_ID

FROM SONG_LISTENING_HISTORY);



	listener_id [PK] character varying (20)

12.User with highest listning time

SELECT USER_ID, SUM(DURATION) AS LISTNING_TIME

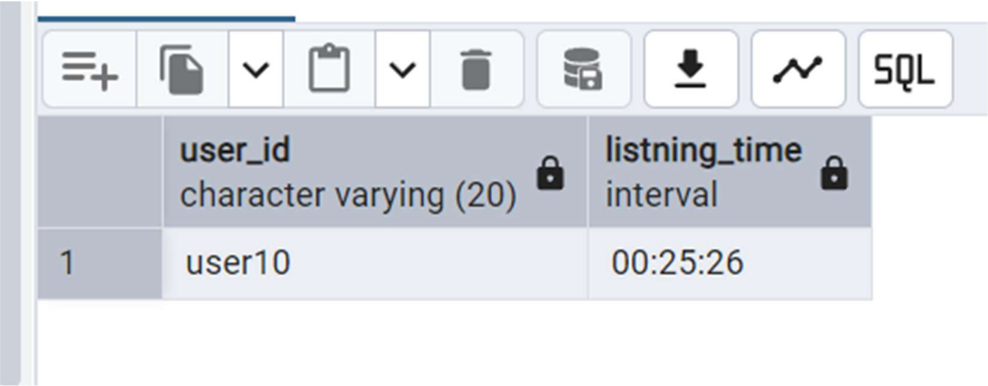
FROM SONG_LISTENING_HISTORY

NATURAL JOIN SONGS

GROUP BY USER_ID

ORDER BY LISTNING_TIME DESC

LIMIT 1;

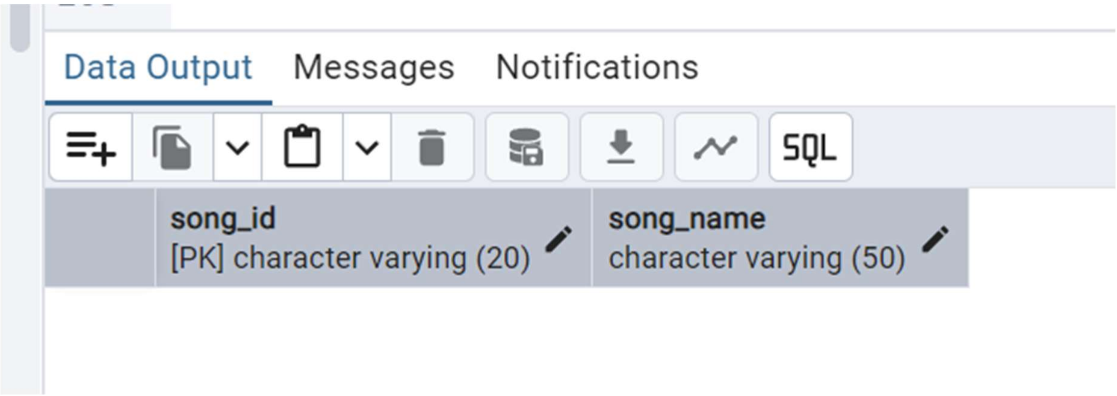


	user_id character varying (20)	listning_time interval
1	user10	00:25:26

13.Songs not added to any playlist

SELECT SONG_ID FROM SONGS

WHERE SONG_ID NOT IN(SELECT DISTINCT SONG_ID FROM PLAYLIST_SONG);



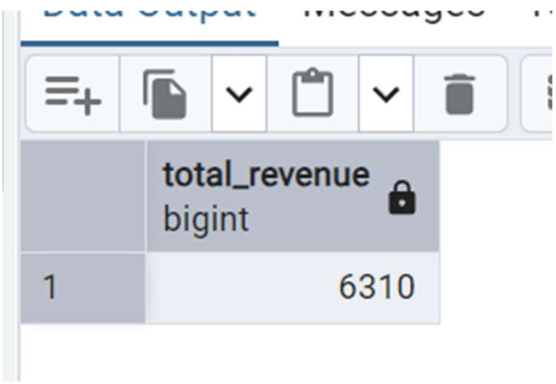
	song_id [PK] character varying (20)	song_name character varying (50)

14.Revenue generated by subscription

SELECT SUM(PAID_AMOUNT) AS TOTAL_REVENUE

FROM (SELECT USER_ID , PAID_AMOUNT

FROM TAKEN_SUBSCRIPTION NATURAL JOIN PAYMENT) AS R;



	total_revenue bigint
1	6310

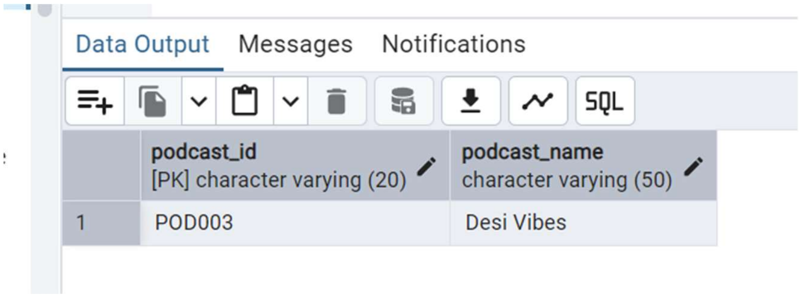
15.Podcast having more then 3 episodes

SELECT PODCAST_ID

FROM PODCAST_EPISODE

GROUP BY PODCAST_ID

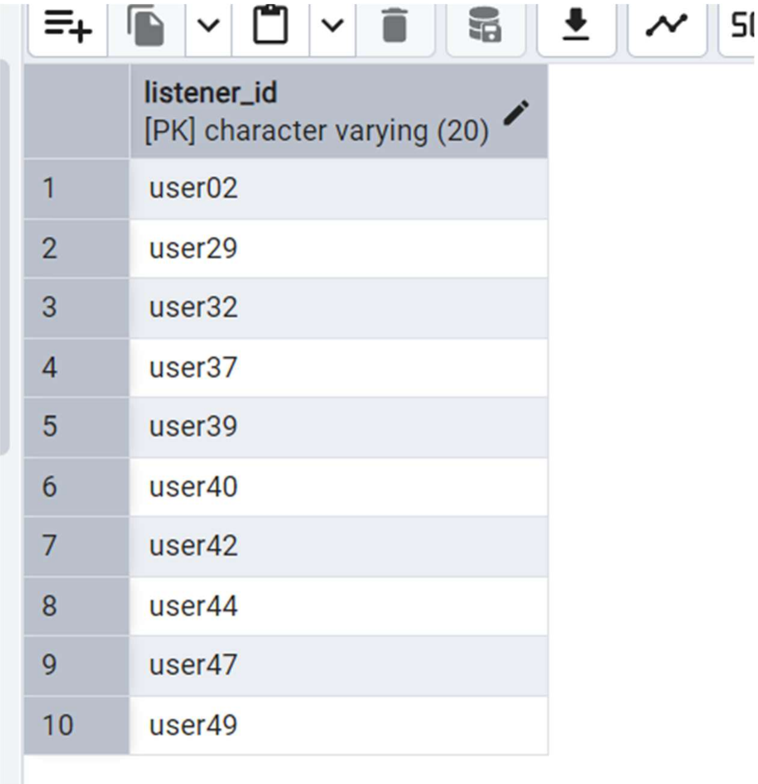
HAVING COUNT(EPISODE_NUMBER) > 3;



	podcast_id [PK] character varying (20)	podcast_name character varying (50)
1	POD003	Desi Vibes

16.Users who have not created any playlist

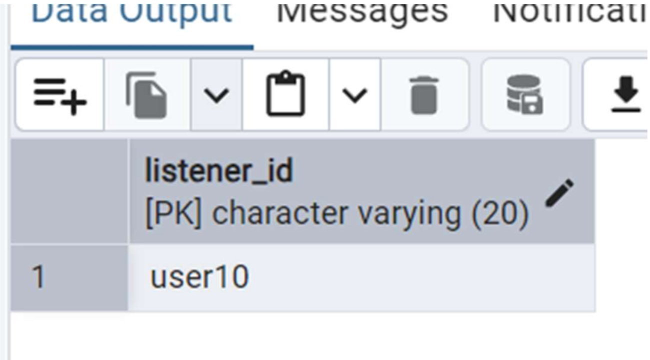
```
SELECT LISTENER_ID FROM LISTENERS

WHERE LISTENER_ID NOT IN(

SELECT DISTINCT USER_ID FROM PLAYLIST_MADE_BY);
```



	listener_id [PK] character varying (20)
1	user02
2	user29
3	user32
4	user37
5	user39
6	user40
7	user42
8	user44
9	user47
10	user49

17. User who has listned to every song of artist name mohit

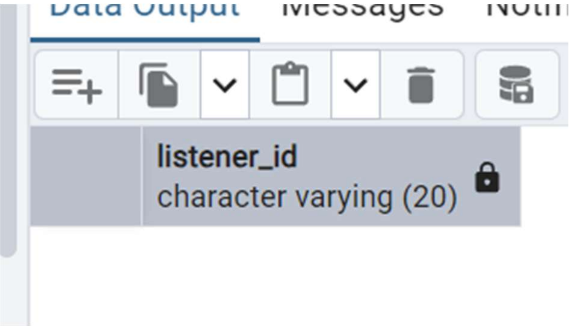
```
SELECT LISTENER_ID FROM LISTENERS AS L

WHERE NOT EXISTS(

SELECT SA.SONG_ID FROM

USERS AS U JOIN SONG_ARTISTS AS SA

ON U.LOGIN_ID = SA.ARTIST_ID

WHERE USER_NAME = 'Mohit'

EXCEPT

SELECT SONG_ID FROM

SONG_LISTENING_HISTORY AS SLH

WHERE SLH.USER_ID = L.LISTENER_ID);
```



	listener_id [PK] character varying (20)
1	user10

18. Users who have liked at least one song from every artist they follow

```
SELECT LISTENER_ID FROM FOLLOW_DETAILS AS FD
GROUP BY LISTENER_ID
HAVING COUNT(DISTINCT FD.ARTIST_ID) <=
(SELECT COUNT(DISTINCT SA.ARTIST_ID)
FROM LIKED_SONGS_DETAILS AS LSD
NATURAL JOIN SONG_ARTISTS AS SA
WHERE USER_ID = FD.LISTENER_ID
AND SA.ARTIST_ID IN (
SELECT ARTIST_ID FROM FOLLOW_DETAILS
WHERE LISTENER_ID = FD.LISTENER_ID));
```



	listener_id character varying (20)

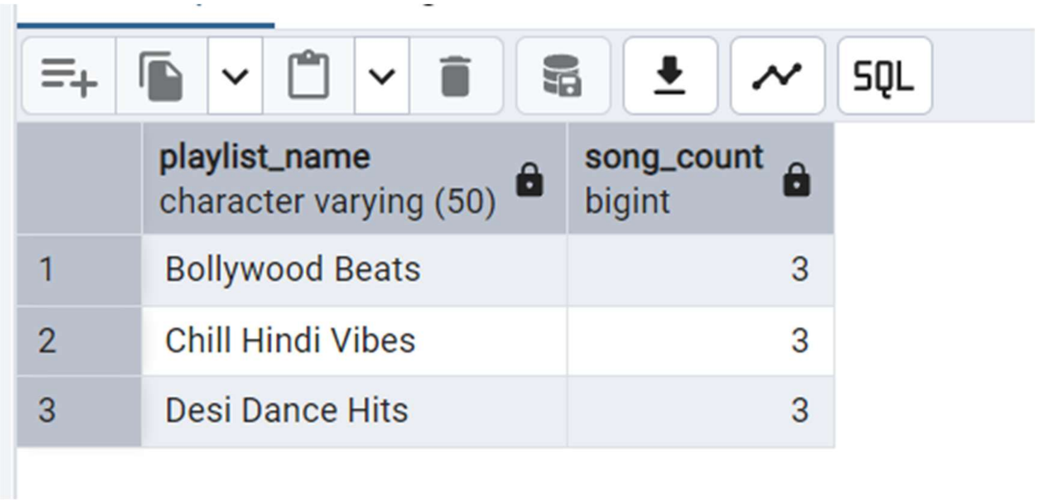
19.top 3 playlist with highest song_count

```
SELECT PLAYLIST_NAME,SONG_COUNT
FROM PLAYLIST NATURAL JOIN
(SELECT PLAYLIST_ID , COUNT(SONG_ID) AS SONG_COUNT
FROM PLAYLIST_SONG
```

GROUP BY PLAYLIST_ID) AS R

ORDER BY SONG_COUNT DESC

LIMIT 3;



	playlist_name character varying (50)	song_count bigint
1	Bollywood Beats	3
2	Chill Hindi Vibes	3
3	Desi Dance Hits	3

20. user with highest payment made

SELECT USER_ID , SUM(PAID_AMOUNT) AS TOTAL_AMOUNT

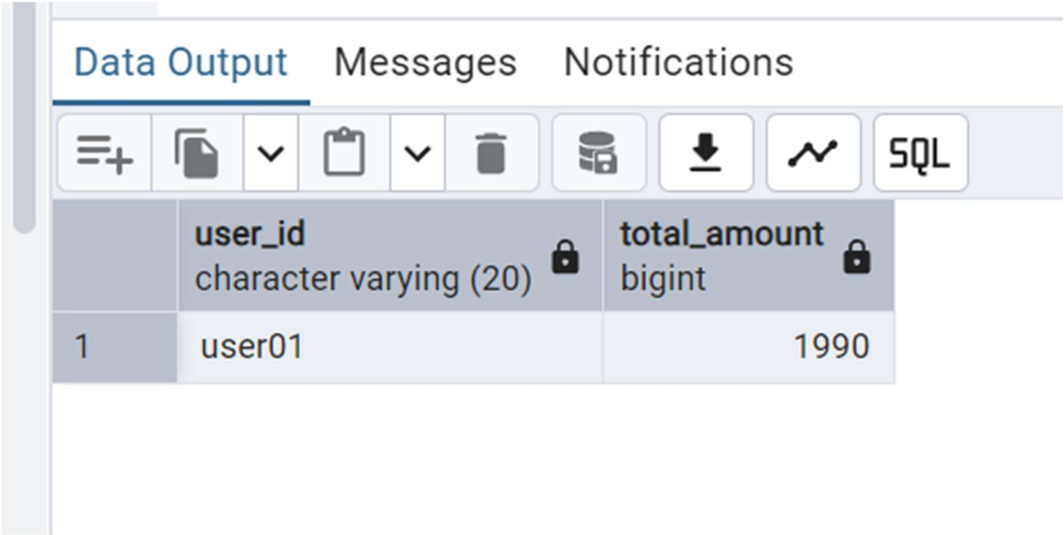
FROM PAYMENT AS P

WHERE P.TYPE = 'credit'

GROUP BY USER_ID

ORDER BY TOTAL_AMOUNT DESC

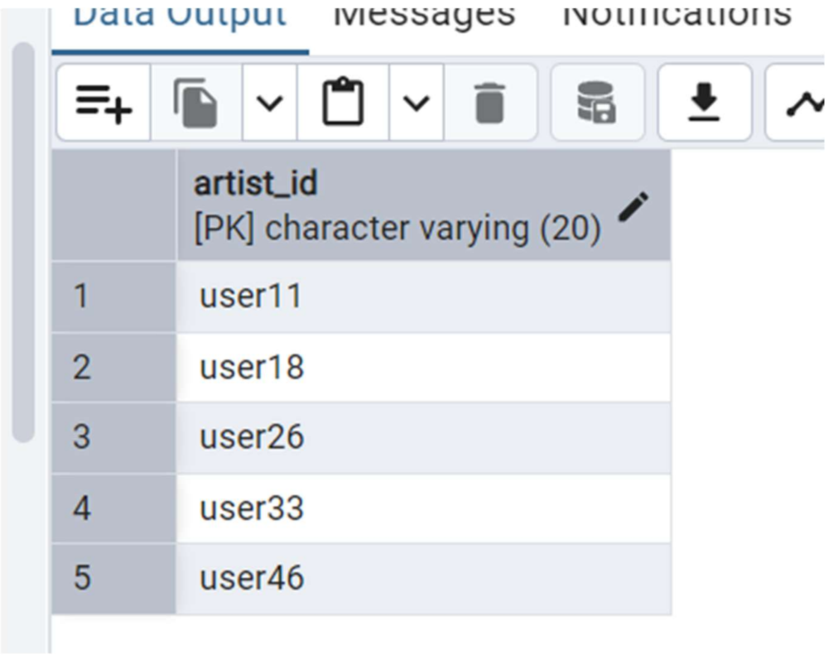
LIMIT 1;



	user_id character varying (20)	total_amount bigint
1	user01	1990

21. Find artists who have not released any songs after a particular date (e.g., 2020-01-01)

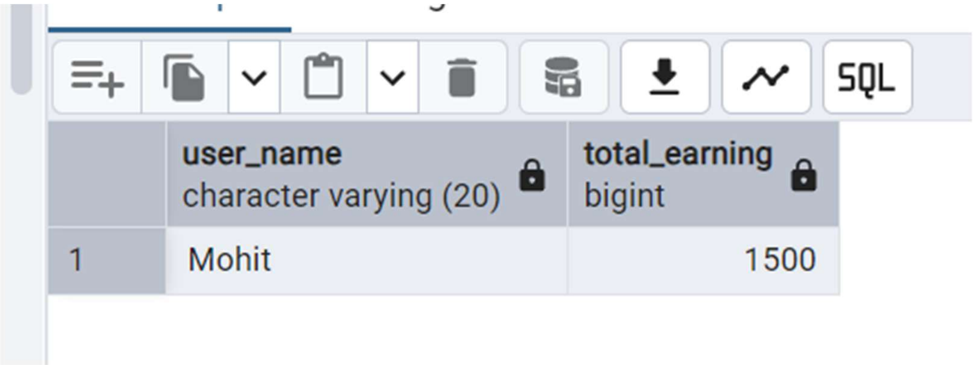
```
SELECT ARTIST_ID FROM ARTISTS
WHERE TYPE = 'singer'
AND ARTIST_ID NOT IN(
SELECT ARTIST_ID FROM SONG_ARTISTS
NATURAL JOIN SONGS
WHERE DATE_OF_RELEASE > '01-01-2020');
```



	artist_id [PK] character varying (20)
1	user11
2	user18
3	user26
4	user33
5	user46

22. Total earning of artist names ‘Mohit’

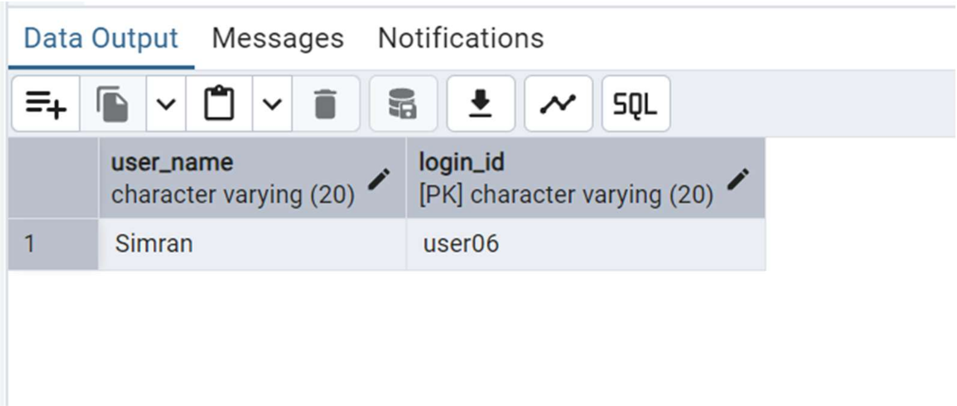
```
SELECT USER_NAME,SUM(PAID_AMOUNT) AS TOTAL_EARNING
FROM USERS JOIN PAYMENT
ON USER_ID = LOGIN_ID
WHERE USER_NAME = 'Mohit' AND TYPE = 'debit'
GROUP BY USER_NAME;
```



	user_name character varying (20)	total_earning bigint
1	Mohit	1500

23.User created most playlist

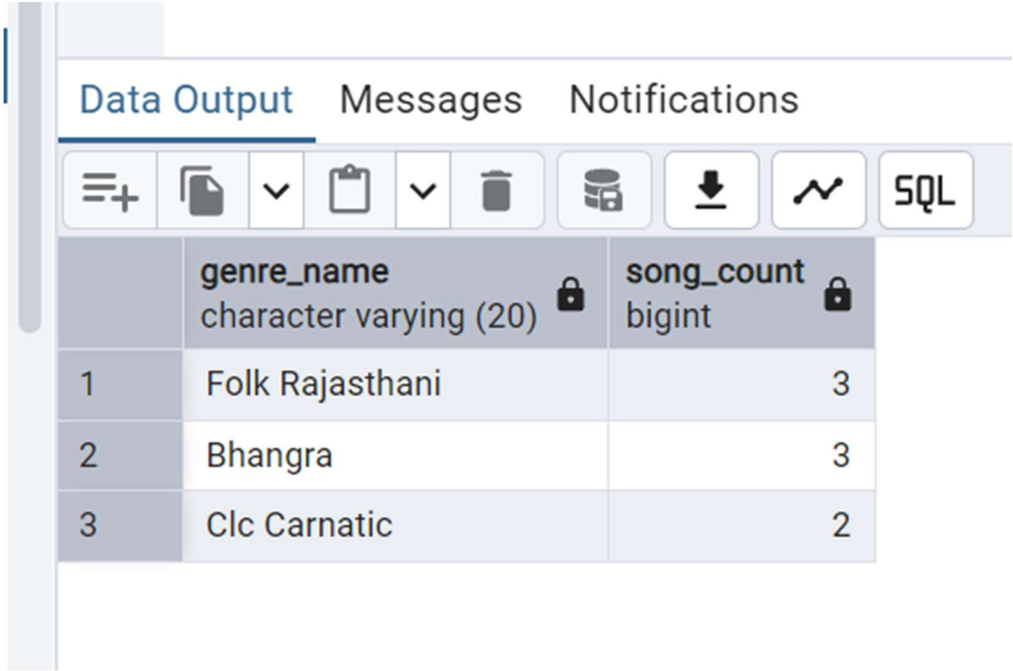
```
SELECT U.USER_NAME, U.LOGIN_ID FROM USERS AS U JOIN
(SELECT USER_ID ,COUNT(PLAYLIST_ID) AS No_of_playlist
FROM PLAYLIST_MADE_BY
GROUP BY USER_ID
ORDER BY No_of_playlist DESC
LIMIT 1) AS P
ON P.USER_ID = U.LOGIN_ID;
```



	user_name character varying (20)	login_id [PK] character varying (20)
1	Simran	user06

24.Top 3 genres with the highest number of songs

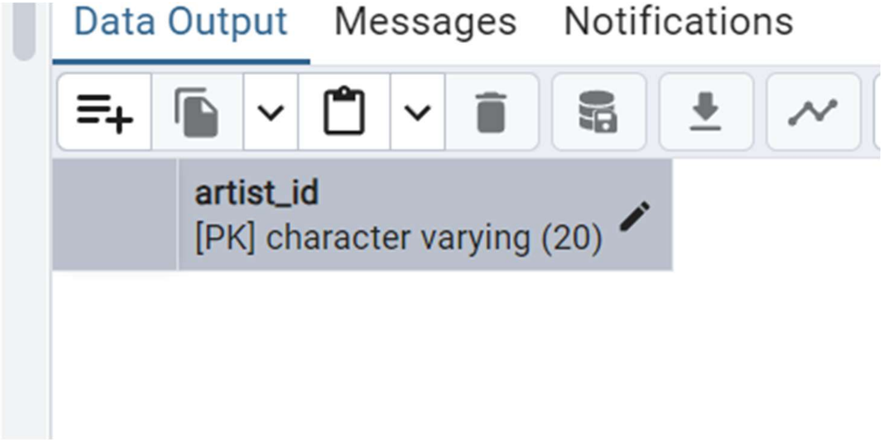
```
SELECT GENRE_NAME,TOTAL_SONGS FROM
GENRE NATURAL JOIN
(SELECT GENRE_ID,COUNT(SONG_ID) AS TOTAL_SONGS
FROM SONGS
GROUP BY GENRE_ID
ORDER BY TOTAL_SONGS DESC

LIMIT 3) AS R;
```



	genre_name character varying (20)	song_count bigint
1	Folk Rajasthani	3
2	Bhangra	3
3	Clc Carnatic	2

25. Artists who have never released an album

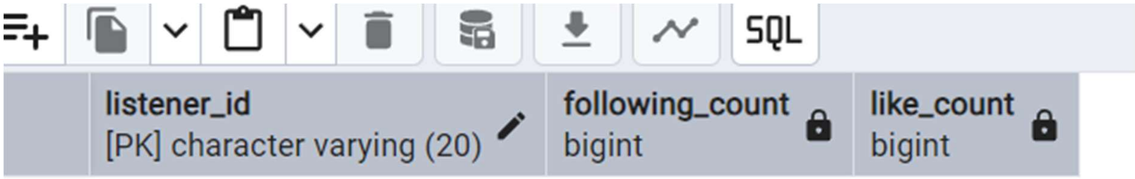
```
SELECT * FROM ARTISTS;

SELECT ARTIST_ID FROM

ARTISTS WHERE TYPE = 'singer' AND

(ARTIST_ID NOT IN

(SELECT DISTINCT ARTIST_ID

FROM ALBUM));
```



	artist_id [PK] character varying (20)

26. Users who follow more than 3 artists AND liked at least 5 songs

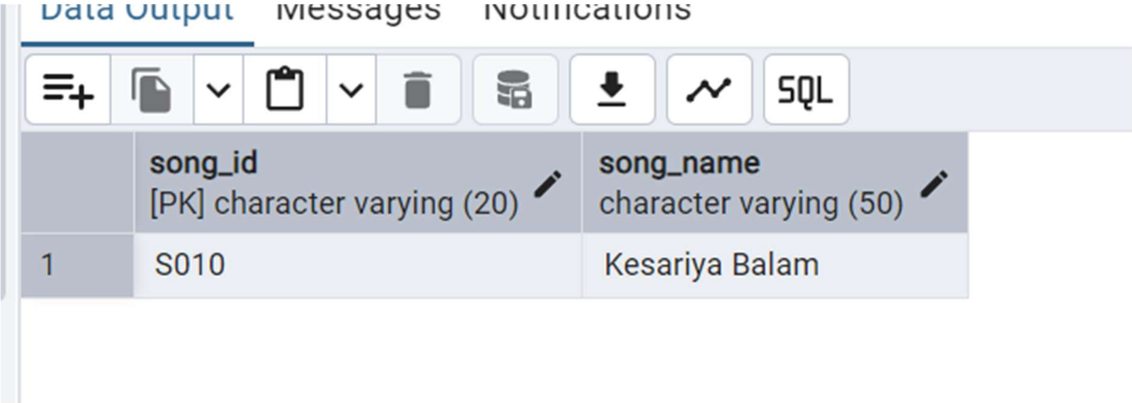
```
SELECT LISTENERS.LISTENER_ID, COUNT(DISTINCT ARTIST_ID) AS FOLLOWING_COUNT,
COUNT(DISTINCT SONG_ID) AS LIKE_COUNT
FROM LISTENERS
NATURAL JOIN FOLLOW_DETAILS
JOIN LIKED_SONGS_DETAILS ON LISTENER_ID = USER_ID
GROUP BY LISTENER_ID
HAVING COUNT(DISTINCT FOLLOW_DETAILS.ARTIST_ID) > 3
  AND COUNT(DISTINCT LIKED_SONGS_DETAILS.SONG_ID) > 5;
```



	listener_id [PK] character varying (20)	following_count bigint	like_count bigint

27.Song that is in most playlist

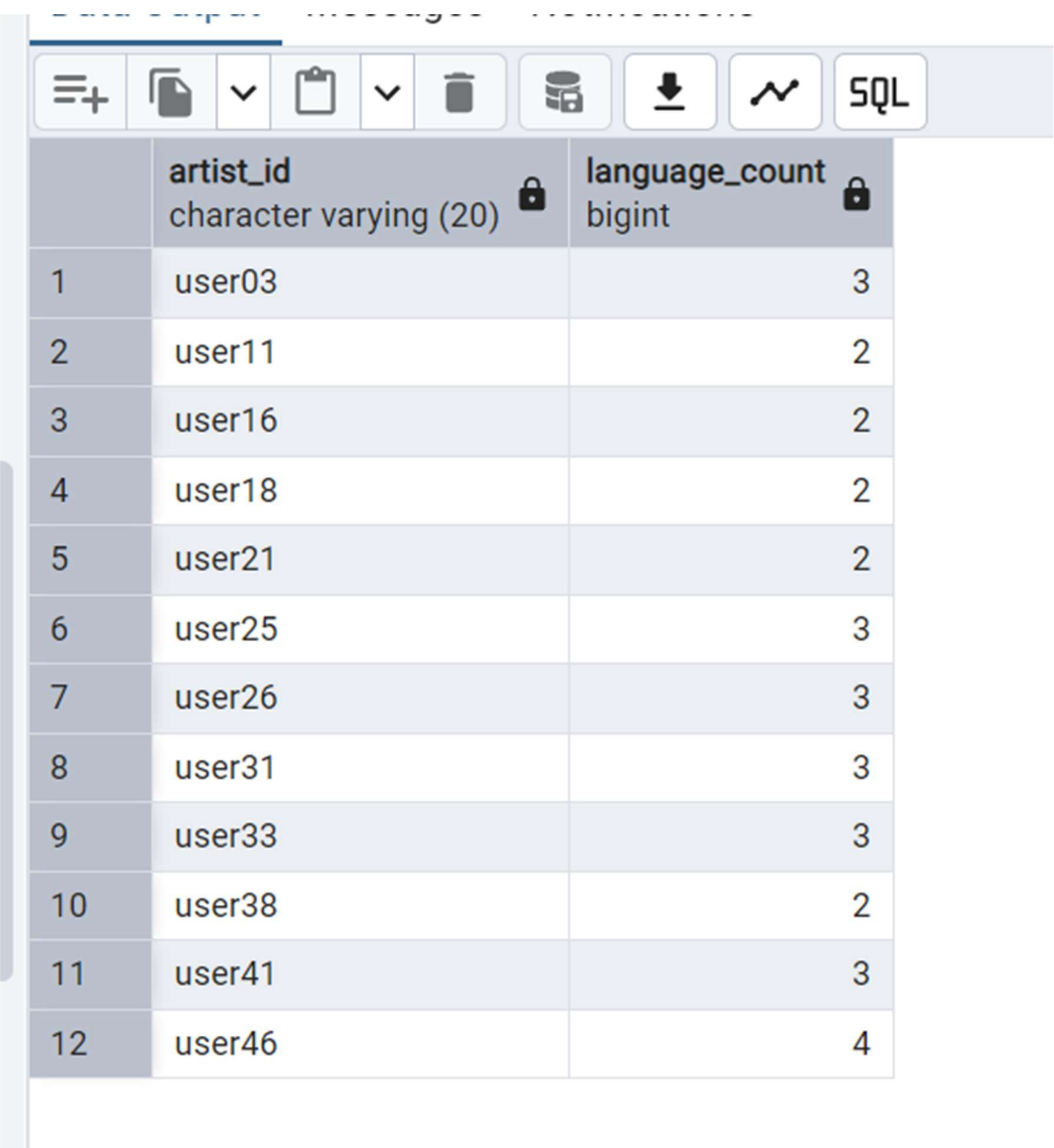
```
SELECT SONG_ID,SONG_NAME FROM
SONGS NATURAL JOIN
(SELECT SONG_ID,COUNT(PLAYLIST_ID) AS PLAYLIST_COUNT
FROM PLAYLIST_SONG
GROUP BY SONG_ID
ORDER BY PLAYLIST_COUNT DESC
LIMIT 1) AS R;
```



	song_id [PK] character varying (20)	song_name character varying (50)
1	S010	Kesariya Balam

28.Artist having song in multiple language

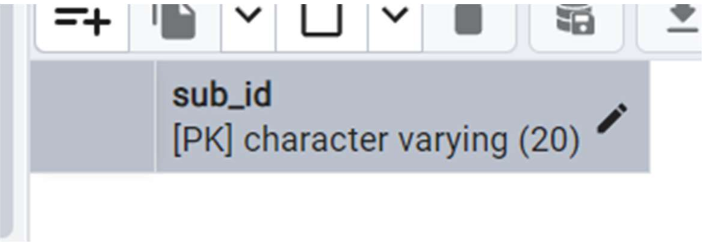
```
SELECT ARTIST_ID, COUNT(DISTINCT LANGUAGE) AS LANGUAGE_COUNT

FROM SONGS NATURAL JOIN SONG_ARTISTS

GROUP BY ARTIST_ID

HAVING COUNT(DISTINCT LANGUAGE) > 1;
```



	artist_id character varying (20)	language_count bigint
1	user03	3
2	user11	2
3	user16	2
4	user18	2
5	user21	2
6	user25	3
7	user26	3
8	user31	3
9	user33	3
10	user38	2
11	user41	3
12	user46	4

29. subscription that were never taken

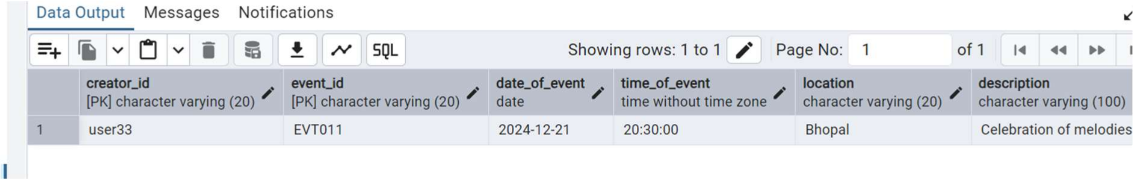
```
SELECT SUB_ID FROM SUBSCRIPTION

WHERE SUB_ID NOT IN

(SELECT DISTINCT SUB_ID FROM TAKEN_SUBSCRIPTION);
```



sub_id
[PK] character varying (20)

30. for given user Events of artist that user follows

```
SELECT EVENTS.* FROM EVENTS JOIN

(SELECT ARTIST_ID FROM FOLLOW_DETAILS

WHERE LISTENER_ID = 'user50') AS R

ON ARTIST_ID = CREATOR_ID;
```



	creator_id [PK] character varying (20)	event_id [PK] character varying (20)	date_of_event date	time_of_event time without time zone	location character varying (20)	description character varying (100)
1	user33	EVT011	2024-12-21	20:30:00	Bhopal	Celebration of melodies