Amazon

You are a Data Analyst on the Amazon Music Recommendation Team focused on understanding the impact of playlists on user engagement....

Question 1: The Amazon Music Recommendation Team wants to know which playlists have the least number of tracks. Can you find out the playlist with the minimum number of tracks?

```sql
SELECT playlist_name
FROM playlists
WHERE number_of_tracks = (
  SELECT MIN(number_of_tracks)
  FROM playlists
);
```

Amazon

You are a Data Analyst on the Amazon Music Recommendation Team focused on understanding the impact of playlists on user engagement....

Question 2: We are interested in understanding the engagement level of playlists. Specifically, we want to identify which playlist has the lowest average listening time per track. This means calculating the total listening time for each playlist in October 2024 and then normalizing it by the number of tracks in that playlist. Can you provide the name of the playlist with the lowest value based on this calculation?

```
WITH total AS (
  SELECT
    p.playlist_name,
    SUM(pe.listening_time_minutes) AS total_minutes,
    p.number_of_tracks
  FROM playlists p
  INNER JOIN playlist_engagement pe ON p.playlist_id = pe.play
list_id
  WHERE pe.engagement_date BETWEEN DATE '2024-10-01' AND DATE
'2024-10-31'
  GROUP BY p.playlist_name, p.number_of_tracks
),
calculate AS (
  SELECT
    playlist_name,
    total_minutes / number_of_tracks AS avg_minutes_per_track
  FROM total
)
SELECT playlist_name
FROM calculate
WHERE avg_minutes_per_track = (
  SELECT MIN(avg_minutes_per_track)
  FROM calculate
);
```

Amazon

You are a Data Analyst on the Amazon Music Recommendation Team focused on understanding the impact of playlists on user engagement....

Question 3: To optimize our recommendations, we need the average monthly listening time per listener for each playlist in October 2024. For readibility, please round down to the average listening time to the nearest whole number.

```sql
SELECT
  p.playlist_id,
  p.playlist_name,
  FLOOR(SUM(pe.listening_time_minutes) * 1.0 / COUNT(DISTINCT
pe.user_id)) AS avg_listening_time_per_user
FROM playlists p
JOIN playlist_engagement pe ON p.playlist_id = pe.playlist_id
WHERE pe.engagement_date BETWEEN DATE '2024-10-01' AND DATE '2
024-10-31'
GROUP BY p.playlist_id, p.playlist_name;
```