Spotify Dataset SQL Analysis Cheat Sheet
==========

1. Listening Behavior & Popularity Trends

A. Trending Over Time

```
SELECT most_playedon, COUNT(*) AS Songs_Played
FROM spotify_data
GROUP BY most_playedon
ORDER BY most_playedon;
```

B. Hidden Gems: High Likes, Low Views

```
SELECT Track, Artist, Views, Likes, ROUND(Likes/Views * 100, 2) AS Like_to_View_Percentage
FROM spotify_data
WHERE Views > 1000
ORDER BY Like_to_View_Percentage DESC
LIMIT 10;
```

C. Engagement by Comments

```
SELECT Track, Artist, Views, Comments,
      ROUND(Comments / Views * 100, 2) AS Comment_Engagement
FROM spotify_data
WHERE Views > 10000
ORDER BY Comment_Engagement DESC
LIMIT 10;
```

2. Audio Feature Analysis

A. Most Danceable Tracks

```
SELECT Track, Artist, Danceability
FROM spotify_data
ORDER BY Danceability DESC
LIMIT 10;
```

B. Party Tracks: High Energy, Low Acousticness

```
SELECT Track, Artist, Energy, Acousticness
FROM spotify_data
WHERE Energy > 0.8 AND Acousticness < 0.3
ORDER BY Energy DESC
LIMIT 10;
```

C. Instrumental vs Vocal

```
SELECT
  CASE
    WHEN Instrumentalness > 0.7 THEN 'Instrumental'
    ELSE 'Vocal'
  END AS Track_Type,
  COUNT(*) AS Count
FROM spotify_data
GROUP BY Track_Type;
```

In-Depth Spotify Dataset SQL Analysis
==========

1. Distribution of Track Popularity Metrics

A. Views Distribution (Binned)

```
SELECT
  CASE
    WHEN Views < 1000 THEN '<1K'
    WHEN Views BETWEEN 1000 AND 10000 THEN '1K-10K'
    WHEN Views BETWEEN 10001 AND 100000 THEN '10K-100K'
    WHEN Views BETWEEN 100001 AND 1000000 THEN '100K-1M'
    ELSE '>1M'
  END AS View_Range,
  COUNT(*) AS Song_Count
FROM spotify_data
GROUP BY View_Range;
```

B. Likes to Views Correlation Buckets

```
SELECT
  CASE
    WHEN Likes / Views BETWEEN 0 AND 0.01 THEN '0-1%'
    WHEN Likes / Views BETWEEN 0.01 AND 0.05 THEN '1-5%'
    WHEN Likes / Views BETWEEN 0.05 AND 0.10 THEN '5-10%'
    ELSE '>10%'
  END AS Like_Rate_Bracket,
  COUNT(*) AS Song_Count
FROM spotify_data
WHERE Views > 0;
```

2. Audio Feature Clustering Indicators

A. Average Features per Album Type

```
SELECT Album_type,
       ROUND(AVG(Energy), 2) AS Avg_Energy,
       ROUND(AVG(Danceability), 2) AS Avg_Danceability,
       ROUND(AVG(Valence), 2) AS Avg_Valence,
       ROUND(AVG(Tempo), 2) AS Avg_Tempo
FROM spotify_data
GROUP BY Album_type;
```

B. Mood Category Distribution

```
SELECT
  CASE
    WHEN Valence >= 0.7 AND Energy >= 0.7 THEN 'Happy & Energetic'
    WHEN Valence < 0.7 AND Energy >= 0.7 THEN 'Energetic but Sad'
    WHEN Valence >= 0.7 AND Energy < 0.7 THEN 'Chill but Happy'
    ELSE 'Low Energy & Sad'
  END AS Mood_Category,
  COUNT(*) AS Track_Count
FROM spotify_data
```

```
GROUP BY Mood_Category;

3. Time-Based Trends
--------------------

A. Monthly Trends in Views
SELECT DATE_FORMAT(most_playedon, '%Y-%m') AS Month,
      SUM(Views) AS Total_Views
FROM spotify_data
GROUP BY Month
ORDER BY Month;

B. Weekday Engagement Patterns
SELECT DAYNAME(most_playedon) AS Weekday,
      ROUND(AVG(Likes), 2) AS Avg_Likes,
      ROUND(AVG(Views), 2) AS Avg_Views
FROM spotify_data
GROUP BY Weekday;

4. Instrumental & Live Song Insights
------------------------------------

A. Live Track Popularity
SELECT Track, Artist, Liveness, Views
FROM spotify_data
WHERE Liveness > 0.8
ORDER BY Views DESC
LIMIT 10;

B. Instrumental Focused Creators
SELECT Artist,
      COUNT(*) AS Track_Count,
      ROUND(AVG(Instrumentalness), 2) AS Avg_Instrumentalness
FROM spotify_data
GROUP BY Artist
HAVING Track_Count > 2 AND Avg_Instrumentalness > 0.7
ORDER BY Avg_Instrumentalness DESC;
```