**הנחות עבור הפרוייקט**

1.מוזיקאי יכול להקליט שירים

2.כל שיר משויך לאלבום או כמה אלבומים.

3.כל אלבום מיוצר על ידי אחד או יותר מוזיקאים

4.בכל קטע מוזיקלי יש אחד או יותר כלי נגינה

5.לאורך כל השנים = בכל הזמנים

6.לכל אלבום יש רשימת כלי נגינה שהשתמשו בכדי להקליט את האלבום

7. שאילתה 9: מוסיקאי ששיתף פעולה עם המספר הגדול ביותר של מוסיקאים אחרים כלומר כל שיתוף פעולה הוא הקלטת אלבום עם אחד או יותר מוסיקאים אחרים.

8.לשיר יש ז'אנר

**טבלת SQL עבור הפעולות**

הערה: עבור פעולות שדרשו יותר משלב אחד – הטבלא מכילה רק את המשפט המרכזי.

|  |  |
| --- | --- |
| **פעולה** | **SQL** |
| 1 | SELECT count(\*) as album\_count FROM albums where albumRecordingBegining > second\_date and albumRecordingEnd < second\_date; |
| 2 | SELECT count(\*) as num\_of\_songs FROM albums  inner join musician\_albums  on musician\_albums.albumId = albums.albumId  inner join musicians  on musicians.musicianId = musician\_albums.musicianId "  inner join album\_songs  on album\_songs.albumId = albums.albumId  where musicianName = musician\_name and albumRecordingBegining > first\_date and albumRecordingEnd < second\_date; |
| 3 | Select COUNT(\*) as count  FROM albums  INNER JOIN musician\_albums  ON albums.albumId = musician\_albums.albumId  INNER JOIN musicians  ON musicians.musicianId = musician\_albums.musicianId  WHERE(Select musicianId from musicians WHERE  musicians.musicianName = musician\_name) = musicians.musicianId and albumRecordingEnd > first\_date' and albumRecordingEnd < second\_date; |
| 4 | select instruments.instrumentName from musical\_numbers  inner join musical\_numbers\_instruments on musical\_numbers.musicalNumberId = musical\_numbers\_instruments.musicalNumberId  inner join instruments on instruments.instrumentId = musical\_numbers\_instruments.instruemntId  group by instruments.instrumentId  limit 1; |
| 5 | select instrumentName from instruments  inner join albums\_instruments on instruments.instrumentId = albums\_instruments.instrumentId  "where albums\_instruments.albumId = (select albums.albumId from albums where albums.albumName = album\_name); |
| 6 | Select producerName, max(reps) from  (Select producerName, count(albumId) as reps  FROM producers  INNER JOIN producer\_albums  ON producer\_albums.producerId = producers.producerId  group by producer\_albums.producerId ) as temp\_table; |
| 7 | select instruments.instrumentMaker from instruments  where instruments.instrumentName = (select instruments.instrumentName from musical\_numbers  inner join musical\_numbers\_instruments on musical\_numbers.musicalNumberId = musical\_numbers\_instruments.musicalNumberId  inner join instruments on instruments.instrumentId = musical\_numbers\_instruments.instruemntId  group by instruments.instrumentId  limit 1); |
| 8 | Select count(\*) as count1 from musicians |
| 9 | select musicianName, max(participants) from  (Select musicians.musicianName,albumId,count(\*) as participants from musicians  INNER JOIN musician\_albums  ON musicians.musicianId = musician\_albums.musicianId  group by albumId) as tablu; |
| 10 | select songGenre, max(rep) as rep from(Select\*, count(\*) as rep from songs  group by songGenre) as table; |
| 11 | SELECT\*, count(\*) as songs FROM songs  inner join recorders  on songs.recorderId = recorders.recorderId  inner join album\_songs  on album\_songs.songId = songs.songId  inner join albums  on albums.albumId = album\_songs.albumId  where albumRecordingBegining > first\_date and albumRecordingEnd < second\_date  group by recorderName  order by songs DESC  limit 1 |
| 12 | SELECT \* FROM albums order by albumRecordingEnd ASC; |
| 13 | Select\*, count(\*) as rep from album\_songs  inner join songs  on album\_songs.songId = songs.songId  group by songName  having rep > 1; |
| 14 | SELECT \*, count(\*) as number\_of\_tracks FROM recorders  inner join songs  on recorders.recorderId = songs.recorderId  inner join album\_songs  on album\_songs.songId = songs.songId  inner join albums  on albums.albumId = album\_songs.albumId  group by albums.albumId  having albums.albumNumberOfTracks = number\_of\_tracks; |
| 15 | select musicians.musicianName from musicians  inner join musician\_songs on musicians.musicianId = musician\_songs.musicianId  inner join songs on songs.songId = musician\_songs.songId  group by musicians.musicianName  order by count(songs.songGenre) desc  limit 1; |