

## 1 Single-Table SQL

Write SQL queries to accomplish each task below. You will not need to join any tables. Assume you have access to tables with the following schemas, where each primary key is in ALL\_CAPS:

- `Songs (SONG_ID, song_name, album_id, weeks_in_top_40)`
- `Artists (ARTIST_ID, artist_name, first_yr_active)`
- `Albums (ALBUM_ID, album_name, artist_id, yr_released, genre)`

(a) Find the names of the 5 songs that spent the fewest weeks in the top 40, ordered from least to most. Break ties by song name in alphabetical order.

(b) Find the name and the first year active for every artist whose name starts with the letter 'B'.

(c) Find the total number of albums released per genre.

(d) Find the total number of albums released per genre. Don't include genres with a count less than 10.

(e) Find the genre for which the most albums were released in the year 2000. Assume there are no ties.

## 2 Multi-Table SQL

Write SQL queries to accomplish each task below. Use the same tables from the previous question (copied from the front page). You will need to use joins.

- `Songs (SONG_ID, song_name, album_id, weeks_in_top_40)`
- `Artists (ARTIST_ID, artist_name, first_yr_active)`
- `Albums (ALBUM_ID, album_name, artist_id, yr_released, genre)`

(a) Find the names of all artists who released a 'country' genre album in 2020.

(b) Find the name of the album with the song that spent the most weeks in the top 40. Assume there is only one such song.

(c) Find the artist name and the most weeks one of their songs spent in the top 40 for each artist. Include artists that have not released an album.