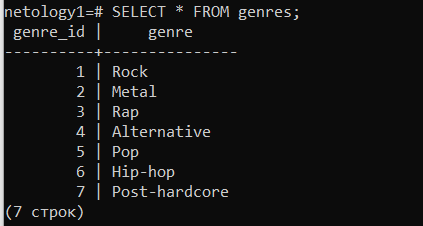
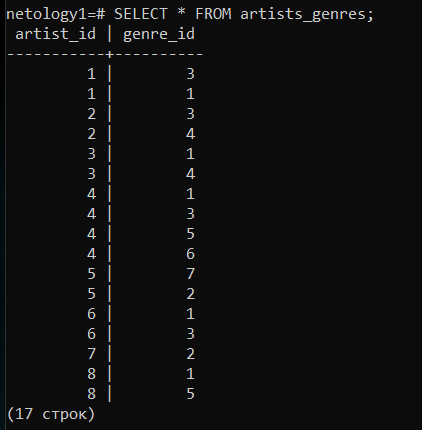
Группировки, выборки из нескольких таблиц

1. Количество исполнителей в каждом жанре

Список жанров – таблица genres



Связи жанров и артистов – таблица artists\_genres



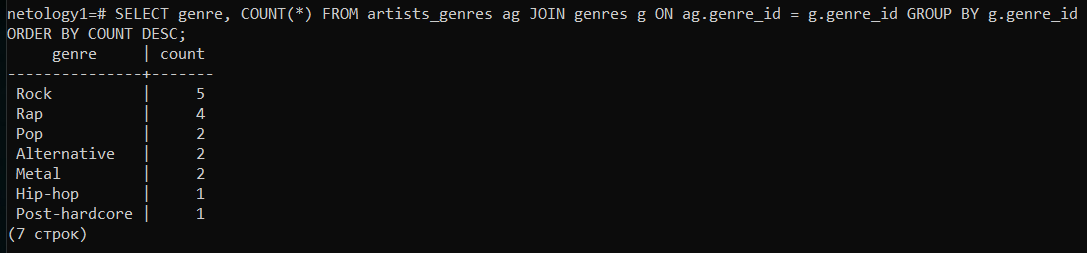
Количество исполнителей в каждом жанре:

SELECT genre, COUNT(\*) FROM artists\_genres ag

JOIN genres g ON ag.genre\_id = g.genre\_id

GROUP BY g.genre\_id

ORDER BY COUNT DESC;

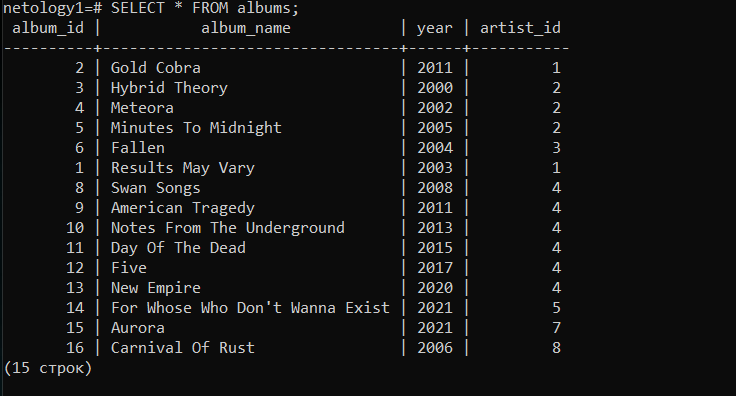


1. Количество треков, вошедших в альбомы 2019-2021 годов.

Альбомов 2019 у меня не оказалось, поэтому расширяю до 2021.

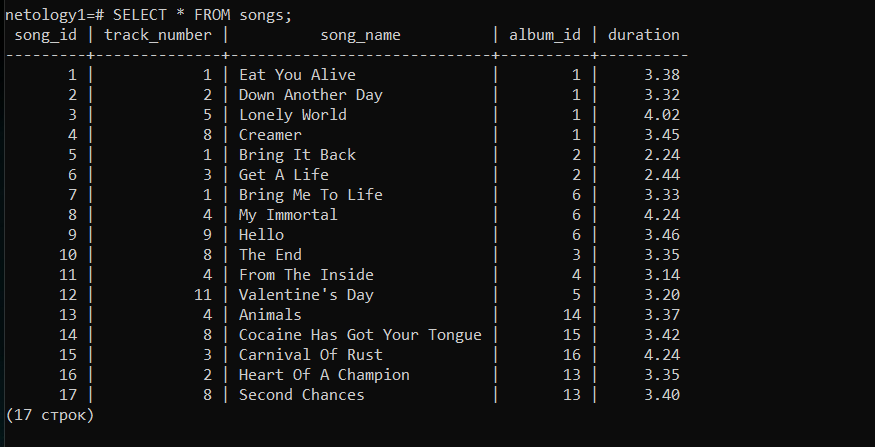
Добавил несколько песен.

Альбомы – таблица albums



Альбомов 2019-2021 всего 3, их id 13,14,15.

Песни – таблица songs



Из песен ссылки на 13,14,15 – всего у 4х, их id тут 16,17,13,14.

Сам запрос:

SELECT song\_name, year FROM songs s

JOIN albums a ON s.album\_id = a.album\_id

WHERE year BETWEEN 2019 AND 2021;

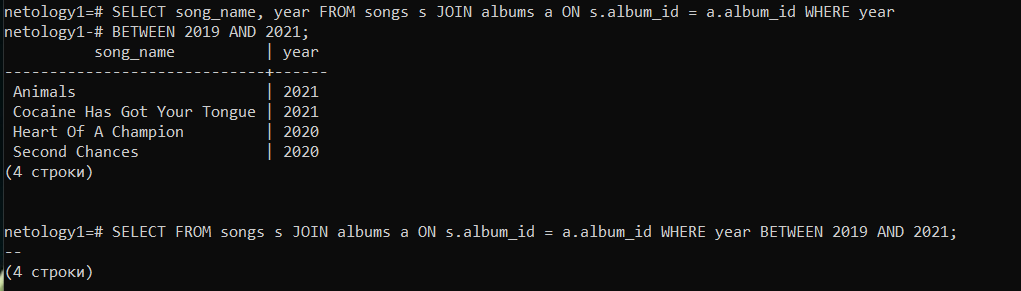
(первый вывод – именно сами треки).

SELECT FROM songs s

JOIN albums a ON s.album\_id = a.album\_id

WHERE year BETWEEN 2019 AND 2021;

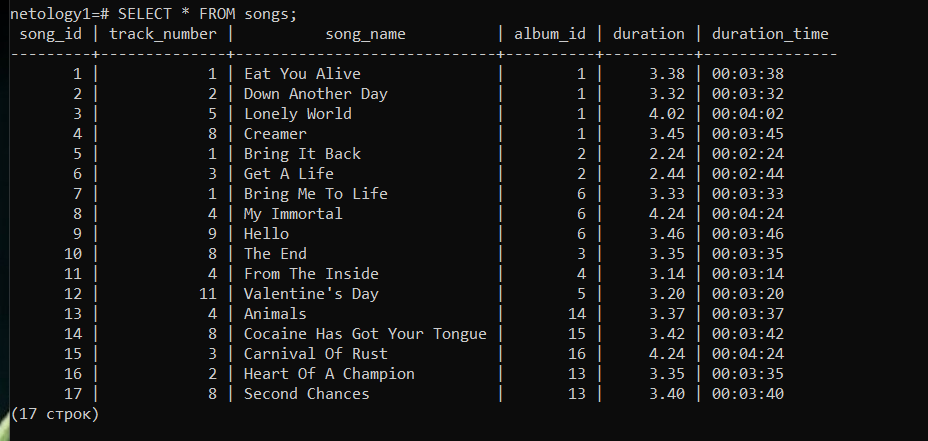
(второй вывод – просто количество строк в получившейся выборке)



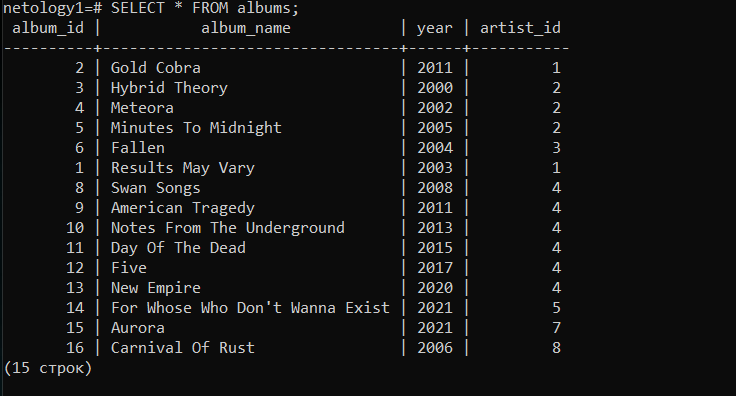
1. Средняя продолжительность песен по каждому альбому.

Добавил в таблицу песни колонку duration\_time формата time, потому что колонка duration формата numeric считалась неправильно.

Песни – таблица songs



Альбомы – таблица albums



В альбомах 8,9,10,11,12 – нет песен (не заполнено). Строка с id 7 – вообще отсутствует (видимо, я что-то добавлял, а потом удалял). Таким образом, из 15 строк должно остаться 10 (для пяти альбомов нет треков, они не войдут в расчёт).

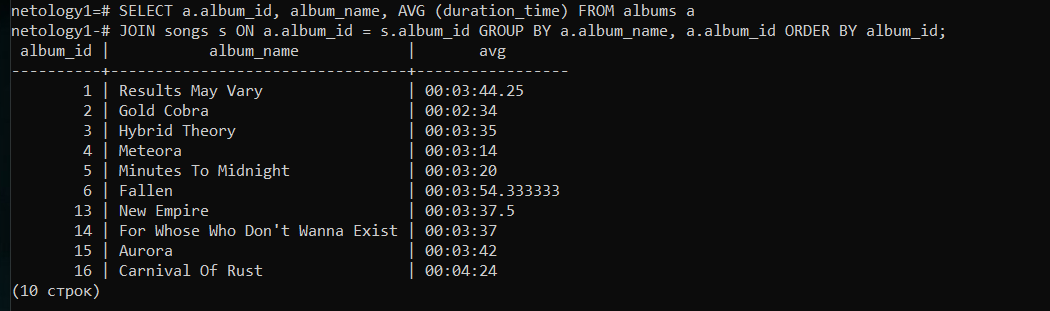
Сам запрос:

SELECT a.album\_id, album\_name, AVG (duration\_time) FROM albums a

JOIN songs s ON a.album\_id = s.album\_id

GROUP BY a.album\_name, a.album\_id

ORDER BY album\_id;

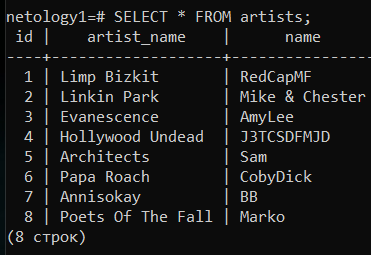


Легко проверить для альбома 2 – там 2 песни длительностью 2.44 и 2.24 – среднее 2.34. И для альбома 13 – там две песни 3.35 и 3.40, соответственно среднее 3.37.5.

Для колонки формата numeric считается неправильно)) Например, для песен 3.30 и 4.00 среднее покажет 3.65 (математически среднее), когда мы ожидаем 3.45.

1. Только те артисты, у кого нет альбома в 2020 году.

Артисты – таблица artists



Альбомы – таблица albums, есть выше на странице.

У меня только один исполнитель имел альбом в 2020 году (artist\_id=4), поэтому он не отобразится, а остальные все попадут в выборку.

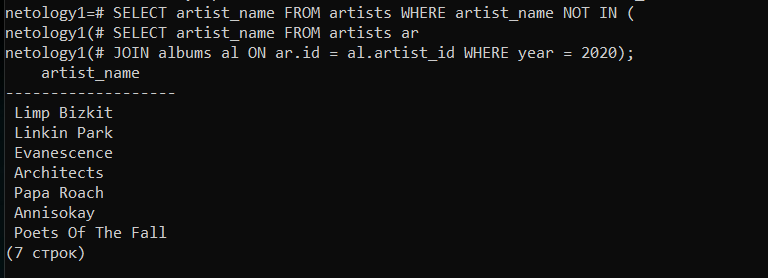
Запрос:

SELECT artist\_name FROM artists WHERE artist\_name NOT IN (

SELECT artist\_name FROM artists ar

JOIN albums al ON ar.id = al.artist\_id WHERE year = 2020

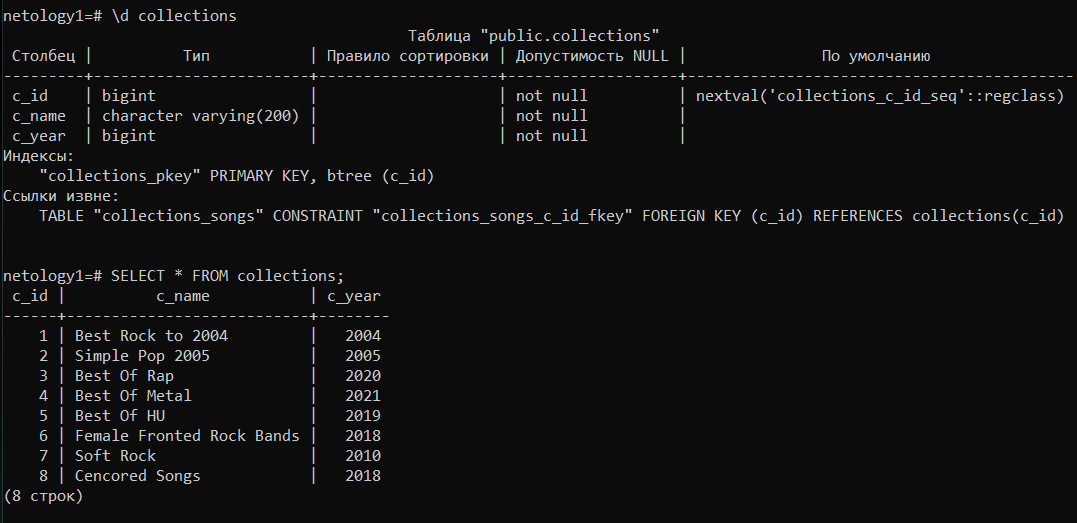
);



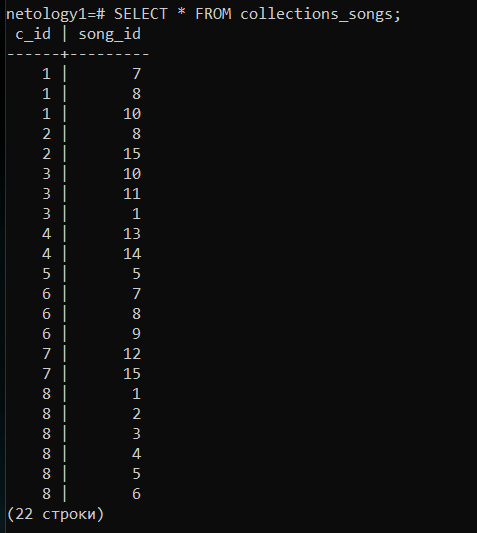
1. Такие сборники, в которых встречается конкретный исполнитель.

Берём исполнителя Linkin Park. В таблице артисты его id = 2. В таблице альбомов у него их 3 – с id = 3, 4, 5. В каждом этом альбоме по 1 песне (их id = 10, 11, 12.

Сборники – таблица collections



Связи сборников и песен – таблица collections\_songs



То есть, песни 10,11,12 встречаются в сборниках с с\_id 1,3,7.

Запрос:

SELECT DISTINCT c\_name FROM collections c

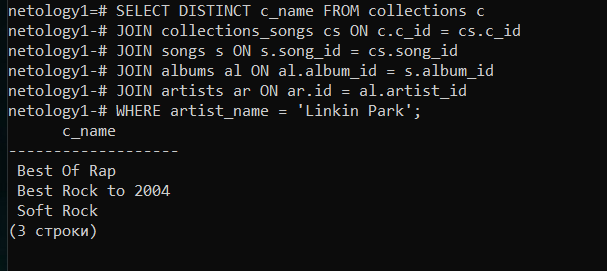
JOIN collections\_songs cs ON c.c\_id = cs.c\_id

JOIN songs s ON s.song\_id = cs.song\_id

JOIN albums al ON al.album\_id = s.album\_id

JOIN artists ar ON ar.id = al.artist\_id

WHERE artist\_name = 'Linkin Park';



1. Название альбомов, в которых присутствуют исполнители более 1 жанра

Таблица исполнители жанры есть на стр1. В ней видно, что только 7й исполнитель имеет 1 жанр, остальные больше. И у этого исполнителя есть только 1 альбом. Соответственно, ожидаю увидеть все альбомы, кроме этого одного.

Запрос:

SELECT album\_name FROM albums WHERE artist\_id IN (

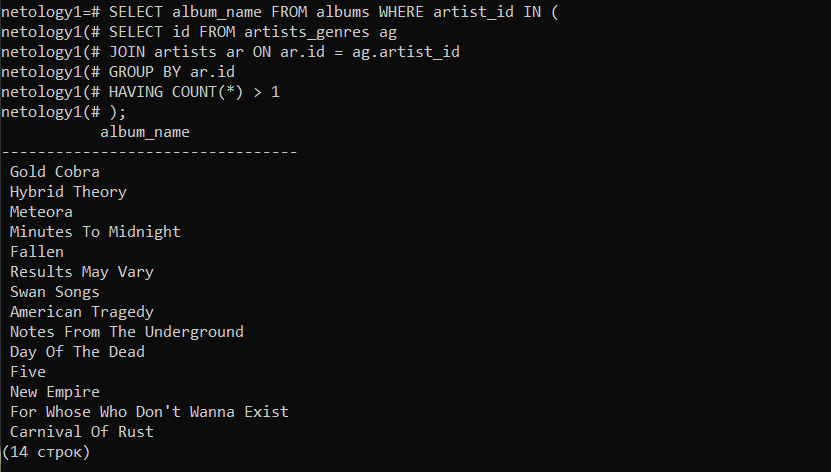
SELECT id FROM artists\_genres ag

JOIN artists ar ON ar.id = ag.artist\_id

GROUP BY ar.id

HAVING COUNT(\*) > 1

);



1. Только те песни, которые не входят ни в какие сборники.

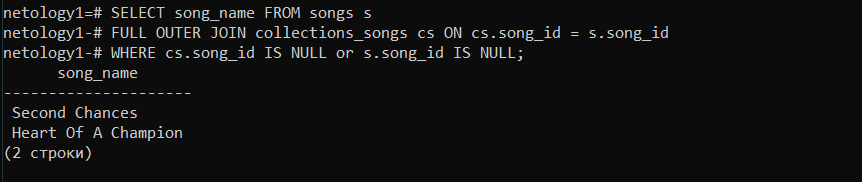
Из таблицы связей сборники-песни на стр7 и таблицы songs на стр4 видно, что только две песни не входят ни в какие сборники.

Запрос:

SELECT song\_name FROM songs s

FULL OUTER JOIN collections\_songs cs ON cs.song\_id = s.song\_id

WHERE cs.song\_id IS NULL or s.song\_id IS NULL;



1. Исполнитель самого короткого и длинного по продолжительности трека

SELECT artist\_name FROM artists ar

JOIN albums al ON ar.id = al.artist\_id

JOIN songs s ON s.album\_id = al.album\_id

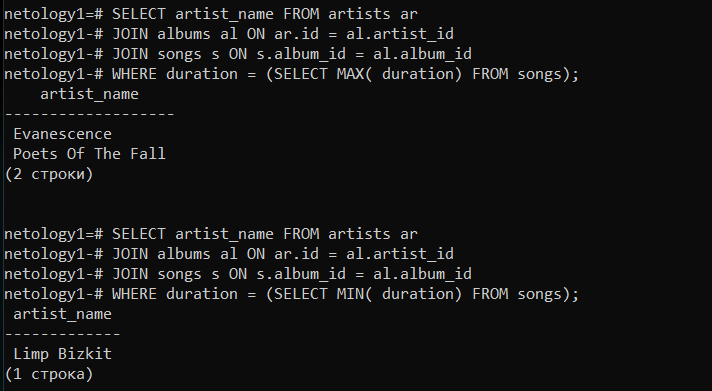
WHERE duration = (SELECT MAX( duration) FROM songs);

SELECT artist\_name FROM artists ar

JOIN albums al ON ar.id = al.artist\_id

JOIN songs s ON s.album\_id = al.album\_id

WHERE duration = (SELECT MIN( duration) FROM songs);



1. Названия альбомов, содержащих наименьшее и наибольшее количество треков

SELECT album\_name, COUNT(\*) FROM albums al

JOIN songs s ON al.album\_id = s.album\_id

GROUP BY al.album\_name HAVING COUNT (\*) =

(SELECT MAX(c) FROM (SELECT COUNT(\*) AS c FROM songs GROUP BY album\_id) AS Q);

SELECT album\_name, COUNT(\*) FROM albums al

JOIN songs s ON al.album\_id = s.album\_id

GROUP BY al.album\_name HAVING COUNT (\*) =

(SELECT MIN(c) FROM (SELECT COUNT(\*) AS c FROM songs GROUP BY album\_id) AS Q);

