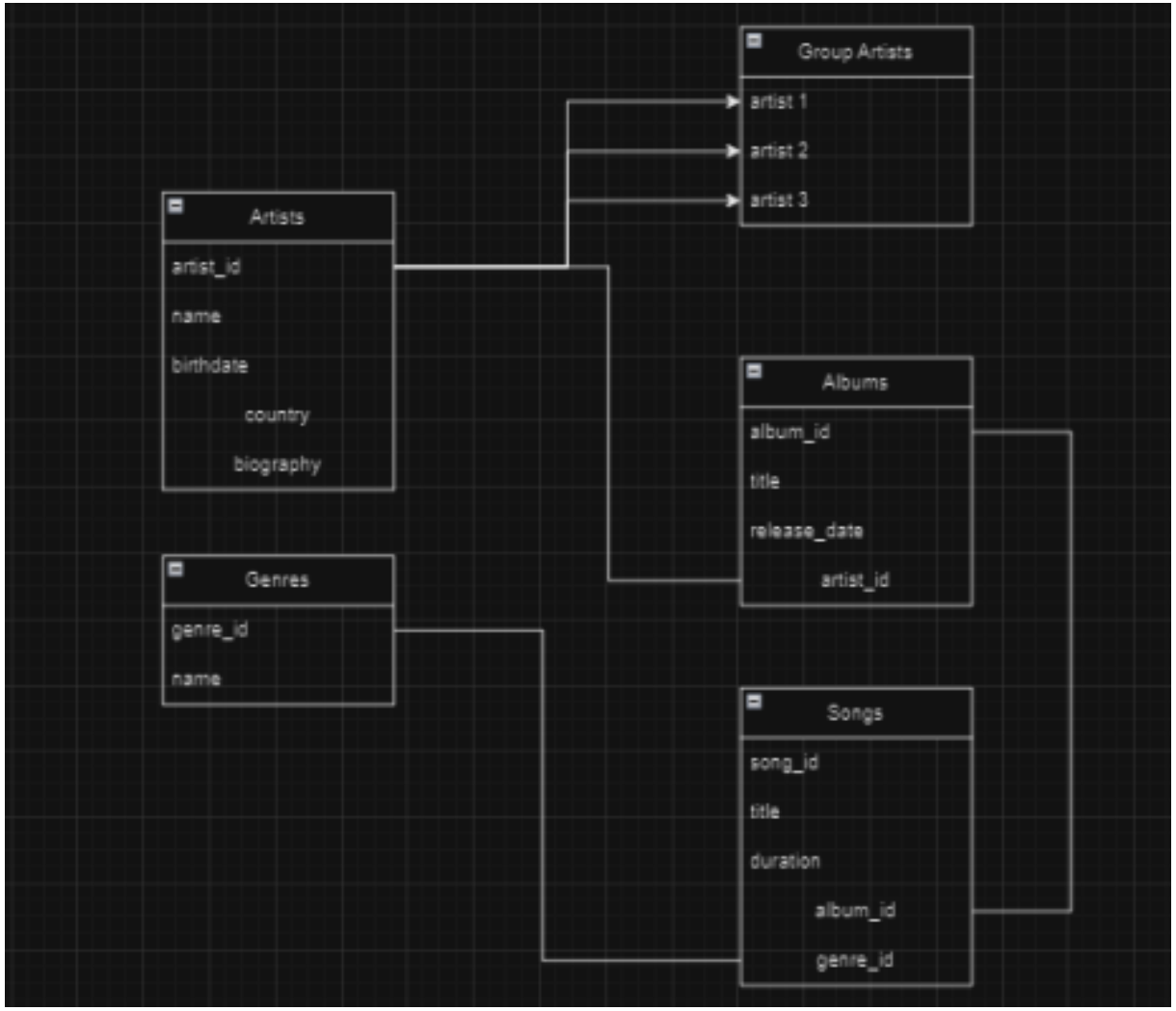
Лабораторная работа №9

Студенты выполнившие лабораторную работу: Бардин Владислав и Подковыров Роман
Группа: ПЗУ
Работали в программе: pgAdmin4
Цель: освоить процесс написания SQl-запросов для выборки из нескольких таблиц. Задача: для своей базы данных написать SELECT-запросы с использованием: - агрегирующих функций - вложенных запросов - оператора GROUP BY для группировки данных - оператора HAVING для фильтрации сгруппированных данных - объединения таблиц при помощи JOIN

Схема базы данных



Пример с агрегирующими функциями

SELECT AVG(duration) AS title, COUNT(*) AS album_id, genre_id

FROM songs

GROUP BY genre_id

```
1  SELECT AVG(duration) AS title, COUNT(*) AS album_id, genre_id
2  FROM songs
3  GROUP BY genre_id
```

Data Output | Messages | Notifications

	title interval	album_id bigint	genre_id integer
1	03:44:49	2	3
2	03:10:28.333333	3	2
3	02:28:00	1	1

Пример с вложенным запросом

SELECT *

FROM artists

WHERE birthdate = (SELECT MAX(birthdate)
FROM artists);

```
1  SELECT *
2  FROM artists
3  WHERE birthdate = (SELECT MAX(birthdate) FROM artists);
```

Data Output | Messages | Notifications

	id [PK] integer	name character varying (100)	birthdate date	country character varying (100)	biography text
1	3	Владислав	2004-12-20	Казахстан	Просто текст

Пример с GROUP BY и HAVING

SELECT genre_id

FROM songs

GROUP BY genre_id

HAVING COUNT(*) > 0 AND MAX(genre_id) > 0;

```
1  SELECT genre_id
2  FROM songs
3  GROUP BY genre_id
4  HAVING COUNT(*) > 0 AND MAX(genre_id) > 0;
```

Data Output | Messages | Notifications

	genre_id integer
1	3
2	2
3	1

объединения таблиц при помощи JOIN

SELECT songs.id, artists.name

FROM songs

INNER JOIN artists ON songs.genre_id = artists.id;

```
1  SELECT songs.id, artists.name
2  FROM songs
3  INNER JOIN artists ON songs.genre_id = artists.id
```

Data Output | Messages | Notifications

	id integer	name character varying (100)
1	2	Владислав
2	40	Владислав

Итог лабораторной работы

Листинг SQL-Запросов

```
Today - 13.11.2023
▶ SELECT songs.id, artists.name FROM songs INNER JOIN art…
11:43:57
▶ SELECT songs.id, artists.name FROM songs INNER JOIN art…
11:43:47
▶ SELECT songs.id, artists.name FROM songs INNER JOIN art…
11:43:25
▶ SELECT songs.id, artists.name FROM artists INNER JOIN a…
11:43:17
▶ SELECT songs.id, artists.name FROM orders INNER JOIN ar…
11:43:09
▶ SELECT genre_id FROM songs GROUP BY genre_id HAVING COU…
11:41:20
▶ SELECT album_id FROM songs GROUP BY album_id HAVING COU…
11:40:57
▶ SELECT album_id FROM songs GROUP BY album_id HAVING COU…
11:40:55
▶ SELECT album_id FROM songs GROUP BY album_id HAVING COU…
11:40:53
▶ SELECT album_id FROM songs GROUP BY album_id HAVING COU…
11:40:51
▶ SELECT album_id FROM songs GROUP BY album_id HAVING COU…
```

```
11:45:23
▶ DELETE FROM songs WHERE album_id = '3';
11:44:56
SELECT * FROM public.songs ORDER BY id ASC
11:44:32
▶ UPDATE songs SET album_id = '2' WHERE genre_id = 3;
11:44:11
▶ UPDATE songs SET album_id = '2' WHERE genre_id = 1;
11:44:04
SELECT * FROM public.songs ORDER BY id ASC
11:43:44
▶ INSERT INTO songs (duration, album_id, genre_id) VALUES…
11:43:20
▶ SELECT * FROM songs LIMIT 3;
11:42:34
▶ SELECT * FROM songs LIMIT 10;
11:42:30
▶ SELECT * FROM songs LIMIT 2;
11:42:25
```