ФГБОУ ВО «Московский технологический университет»

Институт комплексной безопасности и специального приборостроения

Кафедра КБ-1 «Защита информации»

**Дисциплина:** «Безопасность систем баз данных»

**Отчет по лабораторной работе № 7**

**Тема:** «Создание SQL-запросов (Часть 2)»

**Вариант задания № 14**

**Выполнил:**

Студент группы БББО-05-20

Кутьин Захар Сергеевич

**Проверил:**

Войтенков Д.В.

**Учебная цель занятия:** Углубить теоретические знания и выработать практические умения в области настройки безопасности компонента Database Engine с помощью среды управления Management Studio и инструкций языка Transact-SQL.

**Место проведения занятия:** компьютерная аудитория.

**Учебно-материальное обеспечение:**

1. Методическая разработка.
2. Компьютерный класс с ПЭВМ.
3. SQL Server 2022 Developer или Enterprise.

**Задание № 1**

Создать и заполнить базу данных Музыкальных жанров.

**Ход выполнения задания**

Процесс создания базы данных Музыкальных жанров с использованием инструкции языка Transact-SQL показан на рисунке 1.

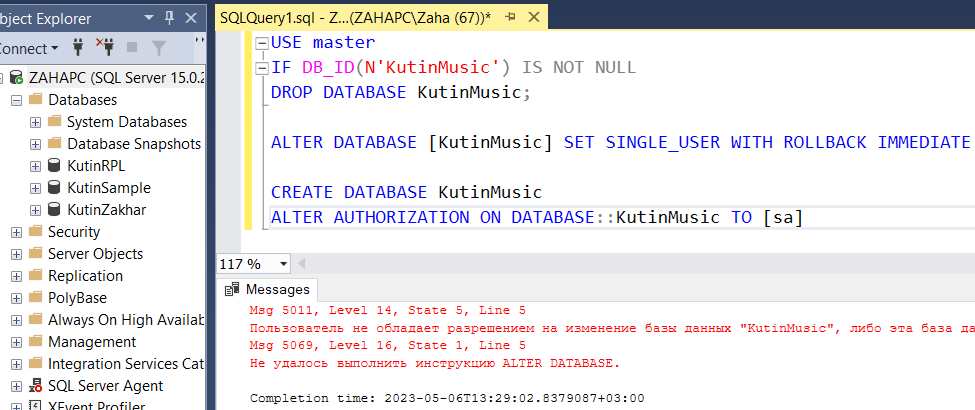


Рисунок 1 - Создание базы данных Музыкальных жанров.

Процесс заполнения базы данных Музыкальных жанров с использованием инструкции языка Transact-SQL показан на рисунках 2-.

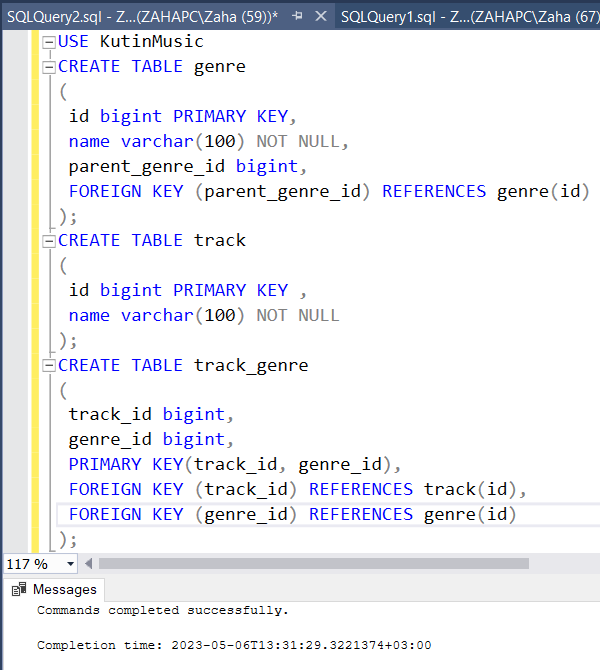


Рисунок 2 - Создание таблиц «Жанры», «Треки» и «Связь трека и жанра» в базе данных Музыкальных жанров.

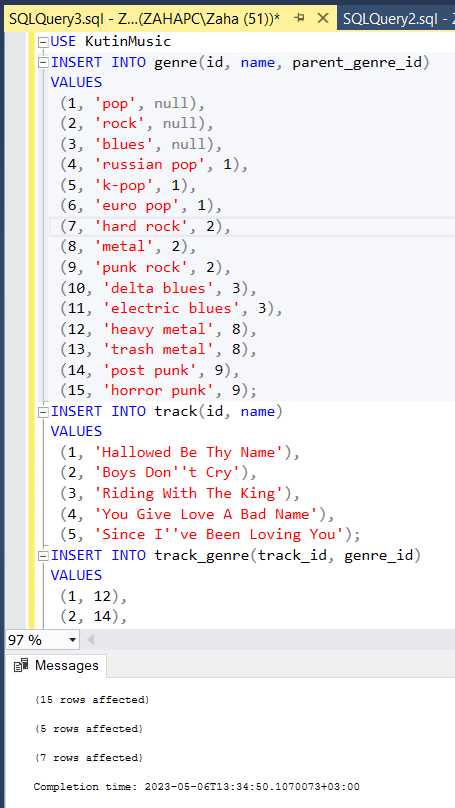


Рисунок 3 - Заполнение данными таблиц базы данных Музыкальных жанров

**Задание № 2**

Необходимо написать запрос, который для каждого трека возвращает все жанры, к которым этот трек принадлежит, в следующем формате:

track\_id — id трека

genre\_id — id жанра

track\_name — наименование трека

genre\_name — наименование жанра

Записи должны быть отсортированы по возрастанию id треков. При равенстве id треков записи необходимо отсортировать по возрастанию id жанров.

**Ход выполнения задания**

Процесс создания запроса с использованием инструкции языка Transact-SQL показан на рисунке 4.

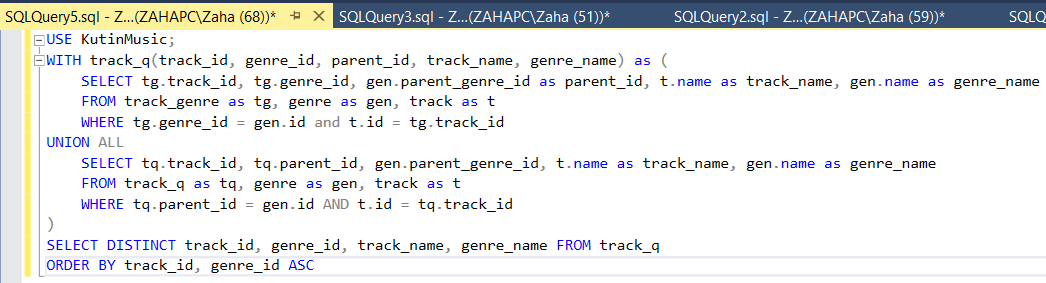


Рисунок 4 – Создание запроса согласно заданию.

Оператор **WITH** формирует CTE (обобщенное табличное выражение) **track\_q**. CTE включает в себя два объединенных подзапроса:

Первый осуществляет выборку данных из всех таблиц базы данных, где id трека в таблице **track** совпадает с id трека в таблице **track\_genre** и id жанра в таблице **genre** совпадает с id жанра в таблице **track\_genre**

Второй осуществляет выборку данных из **track\_q**, **genre** и **track**, где id родительского жанра из **track\_q** совпадает с id жанра из таблицы **genre** и id трека из таблицы **track** совпадает с id трека из **track\_q**.

Результирующий набор данных этих запросов объединен с помощью оператора **UNION ALL**, следовательно набор данных будет включать все абсолютно все строки, даже с дублями.

Затем из **track\_q** осуществляется выборка данных из столбцов **track\_id**, **genre\_id, track\_name, genre\_name** отсортированная по возрастанию по столбцу **track\_id**, при равенстве значений в столбце по столбцу **genre\_id.**

Результат выполнения запроса показан на рисунке 5.

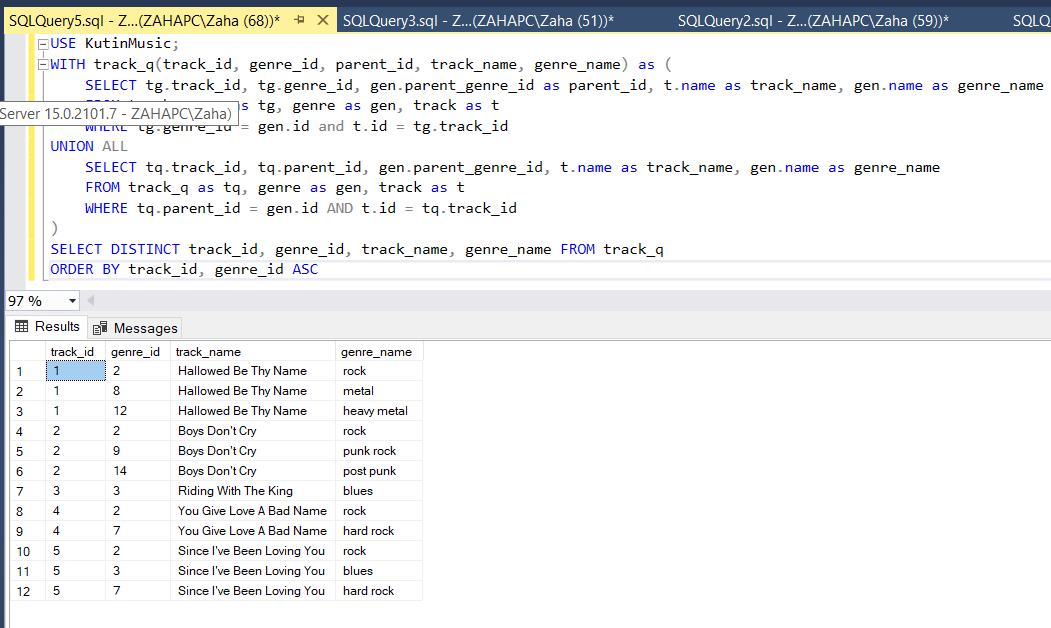


Рисунок 5 - Результат выполнения запроса.

Листинг запроса:

USE KutinMusic;

WITH track\_q(track\_id, genre\_id, parent\_id, track\_name, genre\_name) as (

SELECT tg.track\_id, tg.genre\_id, gen.parent\_genre\_id as parent\_id, t.name as track\_name, gen.name as genre\_name

FROM track\_genre as tg, genre as gen, track as t

WHERE tg.genre\_id = gen.id and t.id = tg.track\_id

UNION ALL

SELECT tq.track\_id, tq.parent\_id, gen.parent\_genre\_id, t.name as track\_name, gen.name as genre\_name

FROM track\_q as tq, genre as gen, track as t

WHERE tq.parent\_id = gen.id AND t.id = tq.track\_id

)

SELECT DISTINCT track\_id, genre\_id, track\_name, genre\_name FROM track\_q

ORDER BY track\_id, genre\_id ASC

**Выводы по проделанной работе**

В результате выполнения лабораторной работы были решены следующие задачи:

- углублены теоретические знания в области настройки безопасности компонента Database Engine;

- освоены приемы настройки безопасности компонента Database Engine с помощью среды управления Management Studio;

- выработаны практические умения настройки безопасности компонента Database Engine с помощью инструкций языка Transact-SQL.