Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОННИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Лабораторная работа №5

Реализация SQL-запросов на простую выборку данных

Студент: Губаревич А.В.

Преподаватель: Силич С.С.

МИНСК 2025

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc161576262)

[1 SQL-ЗАПРОСЫ 4](#_Toc161576263)

[1.1 WHERE 4](#_Toc161576264)

[1.2 WHERE (ORDER BY) 5](#_Toc161576265)

[1.3 ORDER BY 6](#_Toc161576266)

[1.4 SELECT 8](#_Toc161576267)

[1.5 SELECT (WHERE) 10](#_Toc161576268)

[1.6 CROSS JOIN (WHERE AND) 11](#_Toc161576269)

[1.7 CROSS JOIN (WHERE OR) 12](#_Toc161576270)

[1.8 INNER JOIN 14](#_Toc161576271)

[1.9 INNER JOIN (WHERE) 16](#_Toc161576272)

[1.10 LEFT OUTER JOIN (WHERE) 17](#_Toc161576273)

[1.11 LEFT OUTER JOIN 19](#_Toc161576274)

[1.12 RIGHT OUTER JOIN 21](#_Toc161576275)

[1.13 RIGHT OUTER JOIN (WHERE) 22](#_Toc161576276)

[1.14 FULL OUTER JOIN (1) 23](#_Toc161576277)

[1.15 FULL OUTER JOIN (2) 24](#_Toc161576278)

[ЗАКЛЮЧЕНИЕ 25](#_Toc161576279)

# ВВЕДЕНИЕ

В лабораторной работе выполняется создание простых запросов на выборку данных на языке SQL с использованием предложений SELECT, FROM (JOINS), WHERE и ORDER BY оператора SELECT. В работе также требуется рассмотреть использование скалярных функций.

1) Получить у преподавателя задания по вашей собственной схеме данных, созданной в лабораторной работе №2 и реализованной в виде таблиц в СУБД в лабораторной работе №3. Создать запросы по заданиям (по одному запросу на каждое задание).

2) Правила выполнения заданий:

для каждого задания создать реализацию в виде одного оператора выборки, в котором НЕЛЬЗЯ использовать подзапросы и группировку данных (это еще будет в другой лабораторной работе);

при использовании соединений нескольких таблиц обратить внимание на условие задания и сделать выбор между внутренним и внешним соединениями и их вариантами реализации;

перед запуском запроса на выполнение, изучить данные в используемых запросом таблицах, и, если требуется, добавить в вашу схему необходимые новые данные, чтобы результат выборки был контролируемым и не пустым;

выполнить запрос и проанализировать его результат – если есть расхождения между ожидаемыми данными и результатом запроса, то есть повод задуматься о проверке правильности выполнения этого задания.

3) Оформить отчет

# SQL-ЗАПРОСЫ

* 1. WHERE

Вывести исполнителей с количеством участников больше 4.



Рисунок 1.1.0 – Исходная таблица

На рисунке 1.1.1 показан код SQL-скрипта и результат его работы

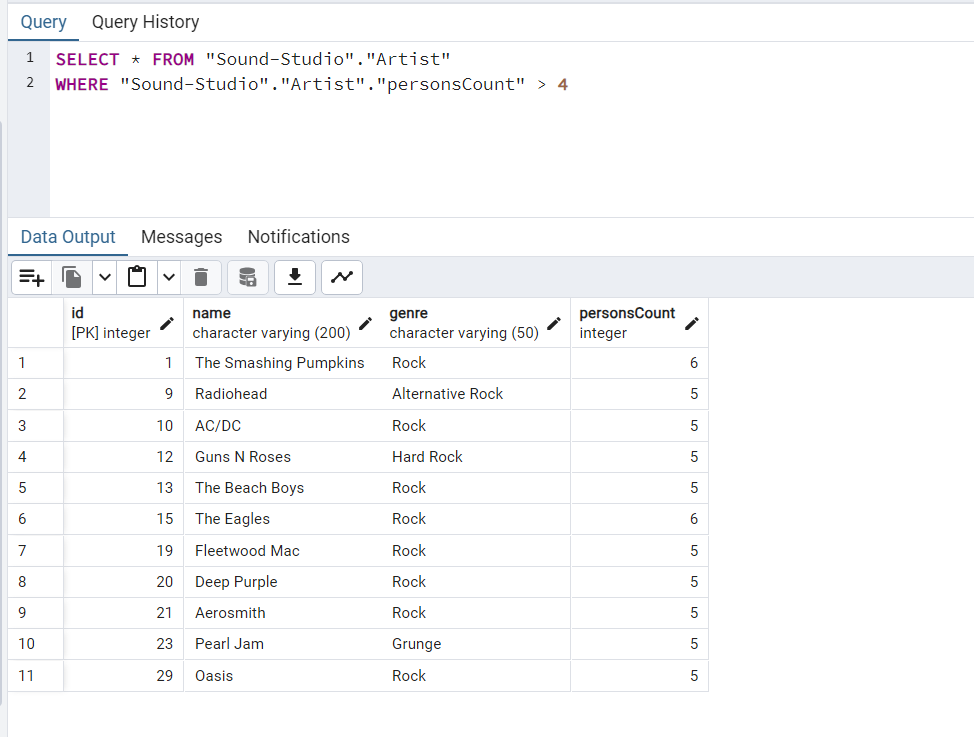


Рисунок 1.1.1 – SQL-cкрипт и результат выполнения

* 1. WHERE (ORDER BY)

Вывести исполнителей в жанре рок с числом участников больше меньше 5 и сгрупировать по алфавиту.



Рисунок 1.2.0 – Исходная таблица

На рисунке 1.2.1 показан код SQL-скрипта и результат его работы

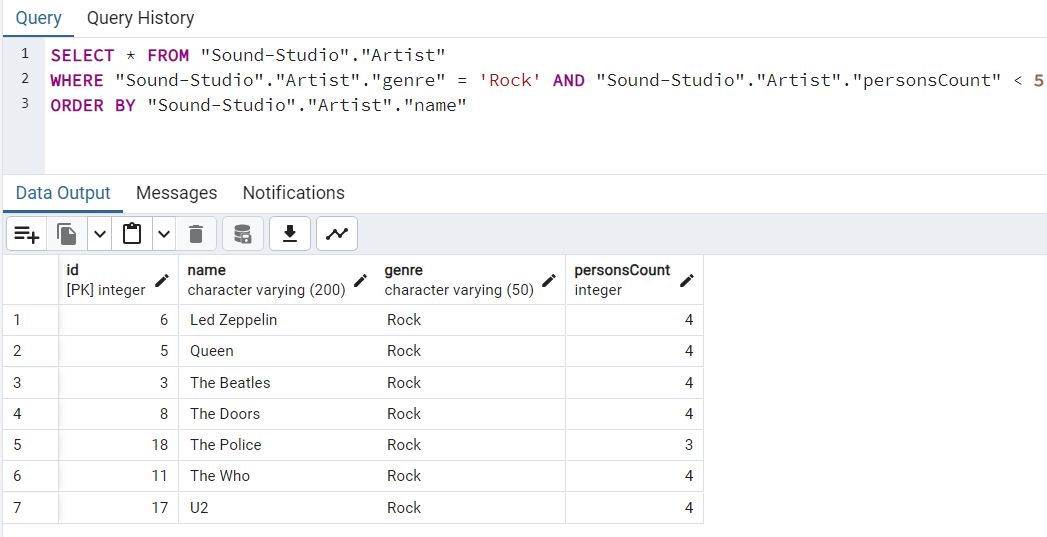


Рисунок 1.2.1 – SQL-cкрипт и результат выполнения

* 1. ORDER BY

Вывести всех сотрудников и сгрупировать по рабочей смене.



Рисунок 1.3.0 – Исходная таблица

На рисунке 1.3.1 показан код SQL-скрипта и результат его работы

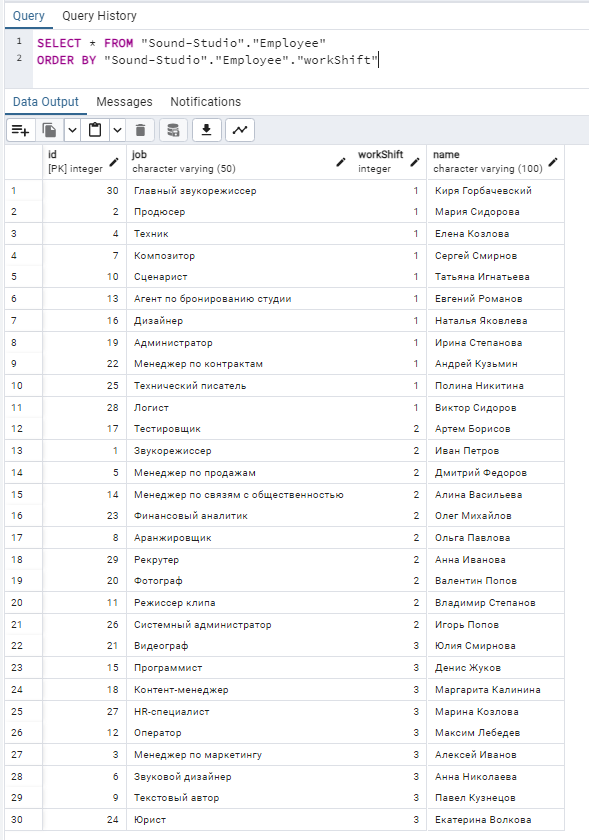


Рисунок 1.3.1 – SQL-cкрипт и результат выполнения

* 1. SELECT

Вывести названия и стоимость аренды всех инструментов.

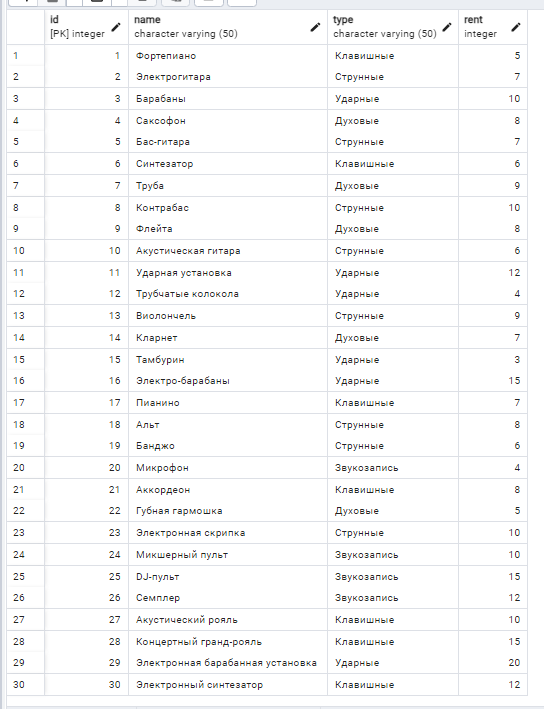


Рисунок 1.4.0 – Исходная таблица

На рисунке 1.4.1 показан код SQL-скрипта и результат его работы



Рисунок 1.4.1 – SQL-cкрипт и результат выполнения

* 1. SELECT (WHERE)

Вывести комнаты, площадь которых больше 30.

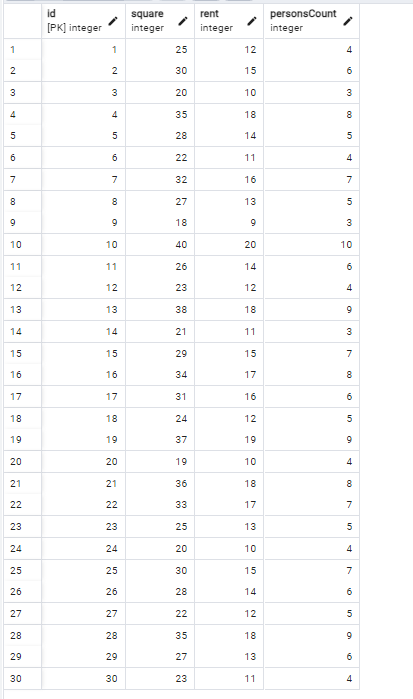


Рисунок 1.5.0 – Исходная таблица Room

На рисунке 1.5.1 показан код SQL-скрипта и результат его работы

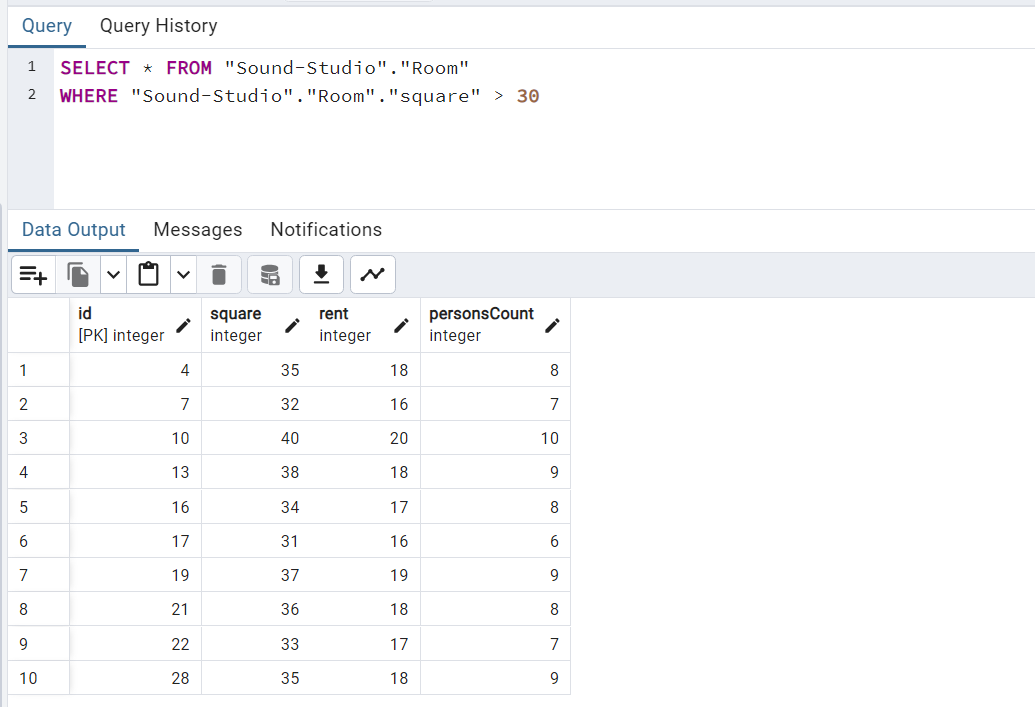


Рисунок 1.5.1 – SQL-cкрипт и результат выполнения

* 1. CROSS JOIN (WHERE AND)

Объединить таблицы Room и Instruments, создав новую таблицу, где стоимость аренды меньше 17, а тип инструмента – ударные

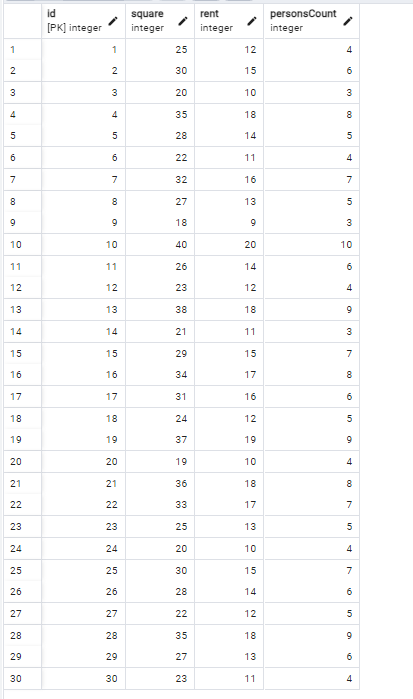


Рисунок 1.6.0 – Исходная таблица Room

На рисунке 1.6.1 показан код SQL-скрипта и результат его работы

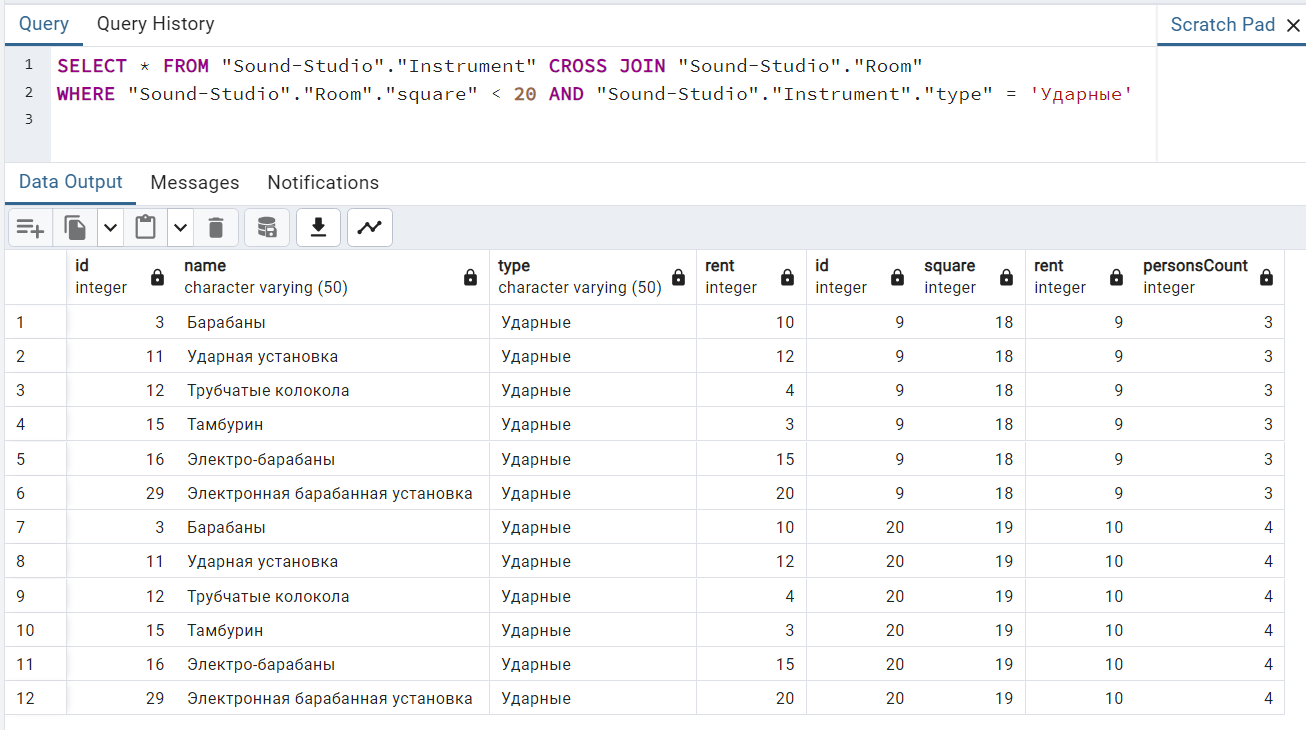


Рисунок 1.6.1 – SQL-cкрипт и результат выполнения

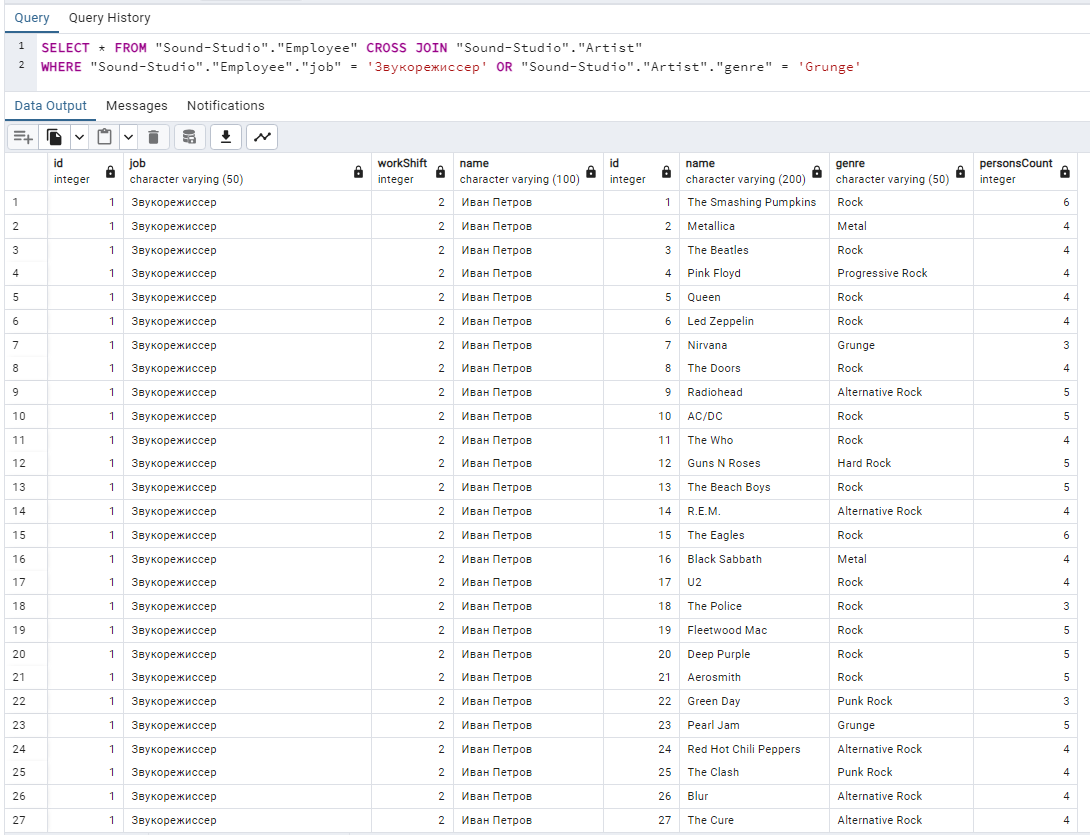
* 1. CROSS JOIN (WHERE OR)

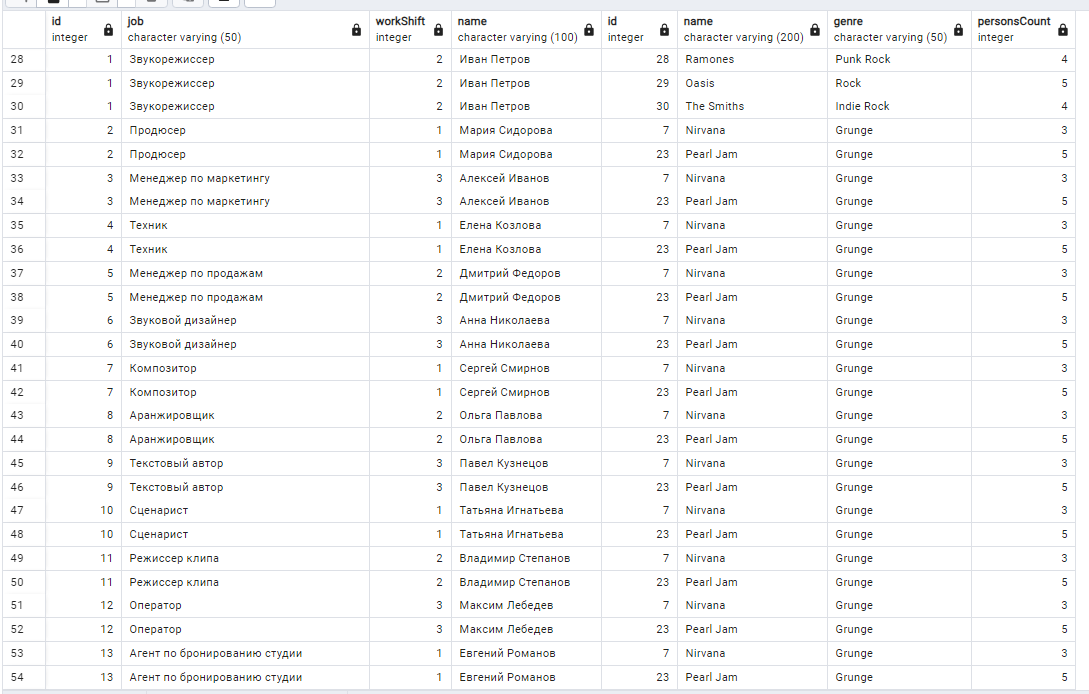
Объединить таблицу Employee и Artist, так чтобы в итоговую таблицу попали сотрудники звукорежиссёры или артисты в жанре «Grunge».



Рисунок 1.7.0 – Исходная таблица Employee

На рисунке 1.7.1 показан код SQL-скрипта и результат его работы





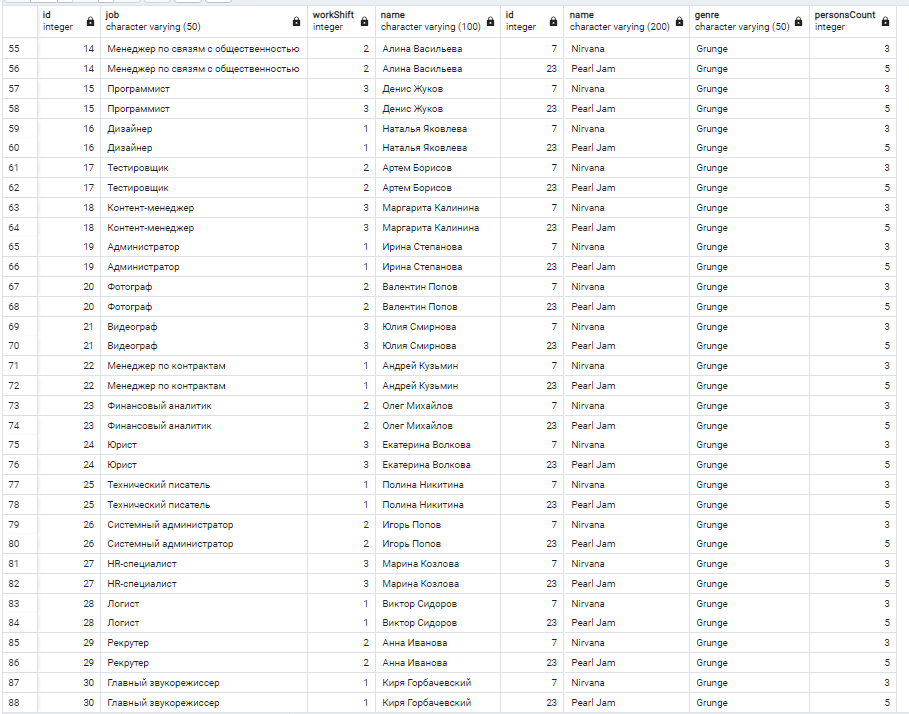


Рисунок 1.7.1 – SQL-cкрипт и результат выполнения

* 1. INNER JOIN

Объединить таблицы Instrument и Room так, чтобы цена аренды инструмента и комнаты были одинаковы.

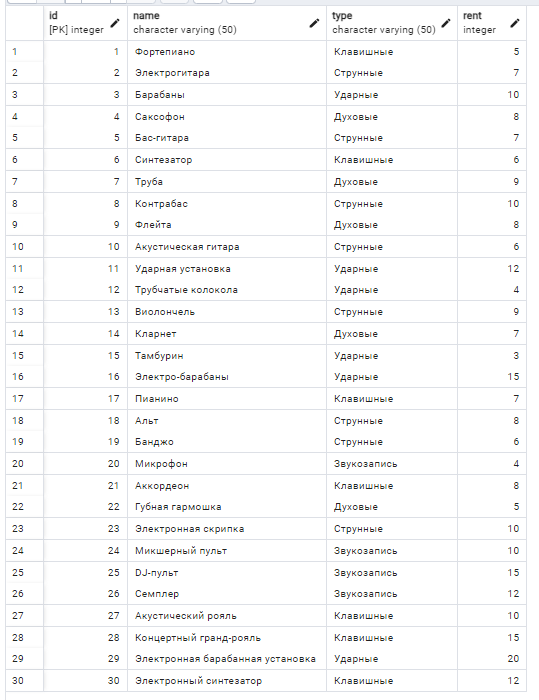


Рисунок 1.8.0 – Исходная таблица Instrument

На рисунке 1.8.1 показан код SQL-скрипта и результат его работы

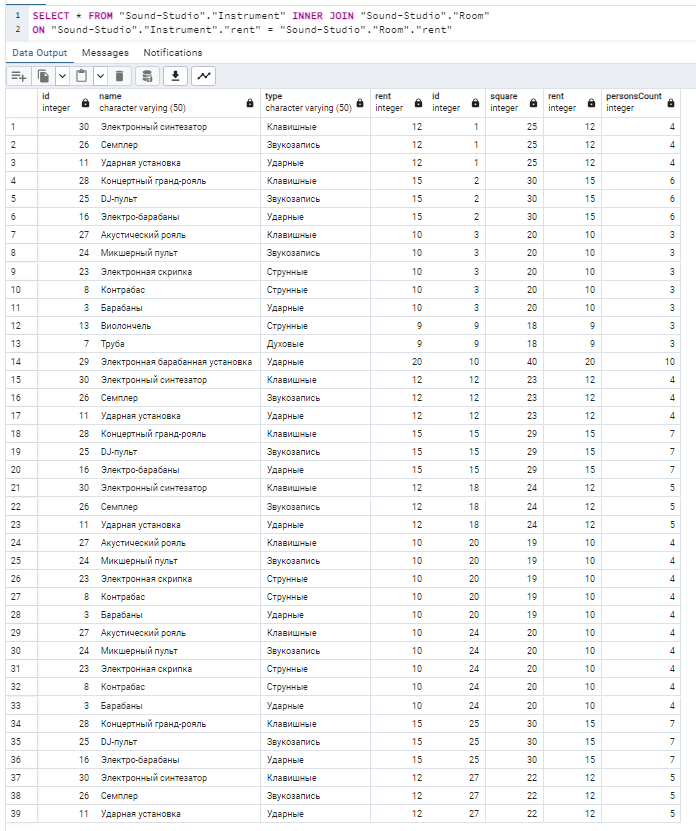


Рисунок 1.8.1 – SQL-cкрипт и результат выполнения

* 1. INNER JOIN (WHERE)

Объединить записи таблиц Instrument и RecordType, где инструмент записи – «Бас-гитара».

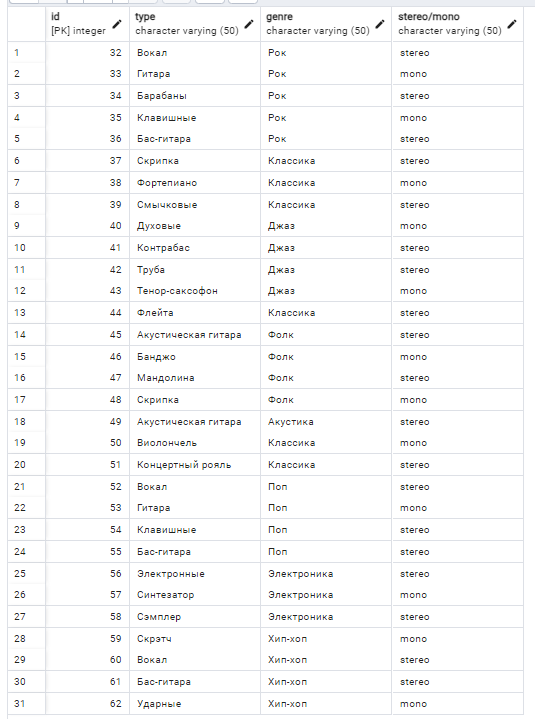


Рисунок 1.9.0 – Исходная таблица RecordType

На рисунке 1.9.1 показан код SQL-скрипта и результат его работы

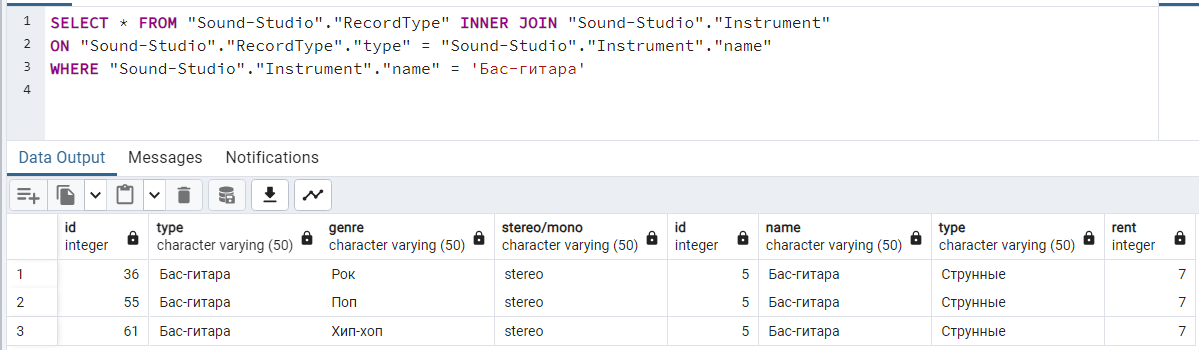


Рисунок 1.9.1 – SQL-cкрипт и результат выполнения

* 1. LEFT OUTER JOIN (WHERE)

Объединить записи таблиц Instrument и Room, c одинаковой арендой, меньшей 12.

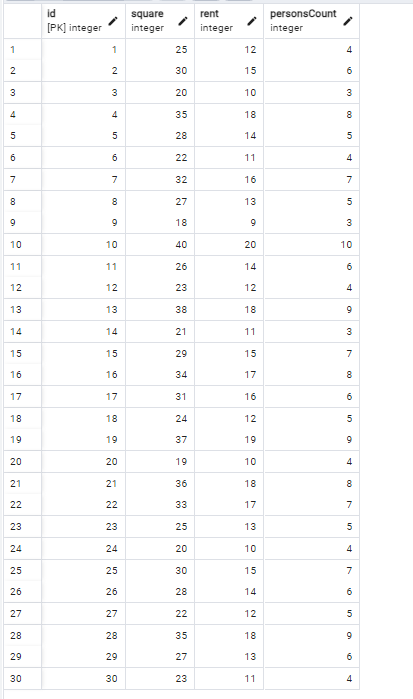


Рисунок 1.10.0 – Исходная таблица Room

На рисунке 1.10.1 показан код SQL-скрипта и результат его работы

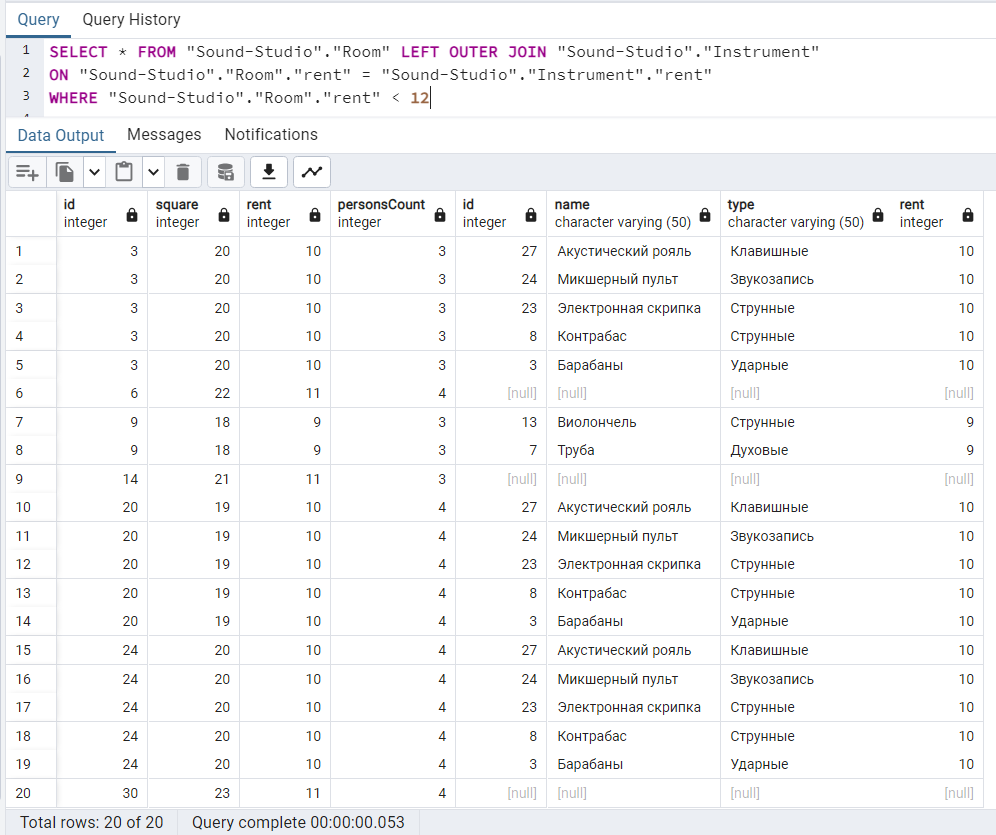


Рисунок 1.10.1 – SQL-cкрипт и результат выполнения

* 1. LEFT OUTER JOIN

Объединить записи таблиц Employee и Record, c одинаковым id сотрудника.



Рисунок 1.11.0 – Исходная таблица Employee

На рисунке 1.11.1 показан код SQL-скрипта и результат его работы

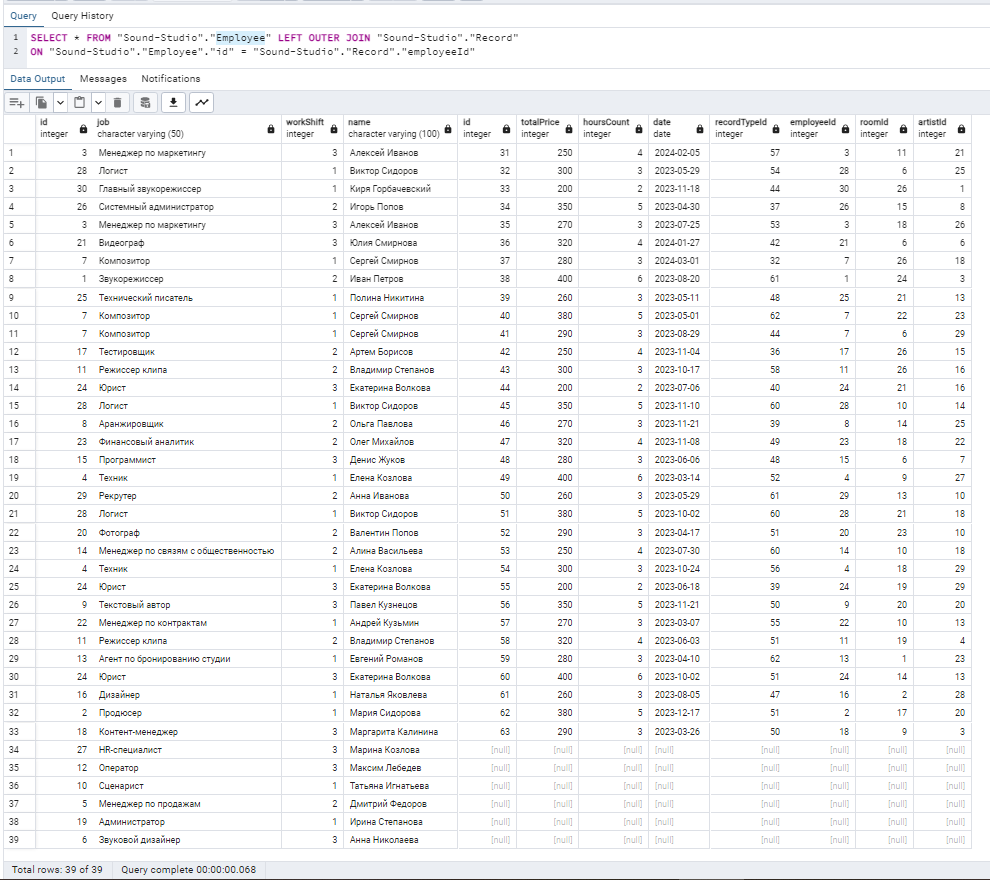


Рисунок 1.11.1 – SQL-cкрипт и результат выполнения

* 1. RIGHT OUTER JOIN

Вывести все инструменты и добавить их тип.

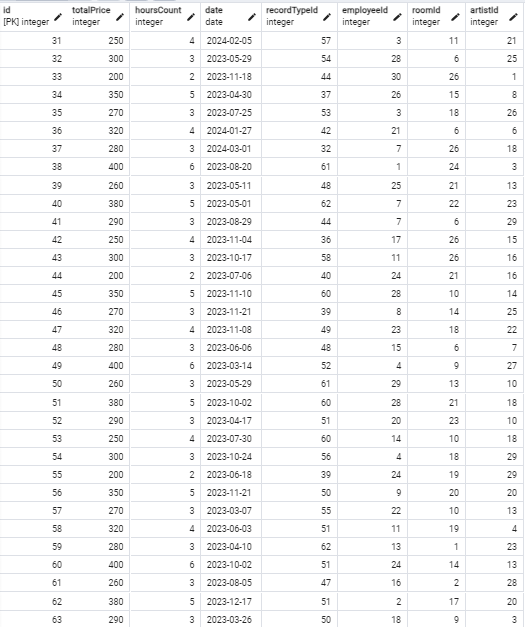


Рисунок 1.12.1 – Исходная таблица Record



Рисунок 1.12.2 – SQL-cкрипт и результат выполнения

* 1. RIGHT OUTER JOIN (WHERE)

Запрос на объединение таблицы *Record* и *Artist*

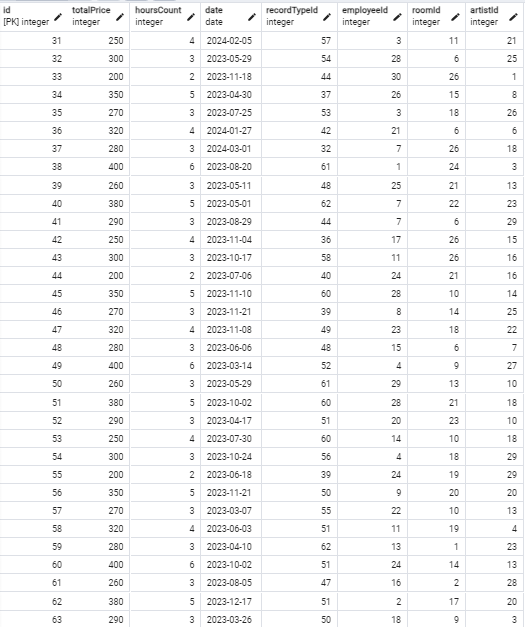


Рисунок 1.13.0 – Исходная таблица Record

На рисунке 1.13.1 показан код SQL-скрипта и результат его работы

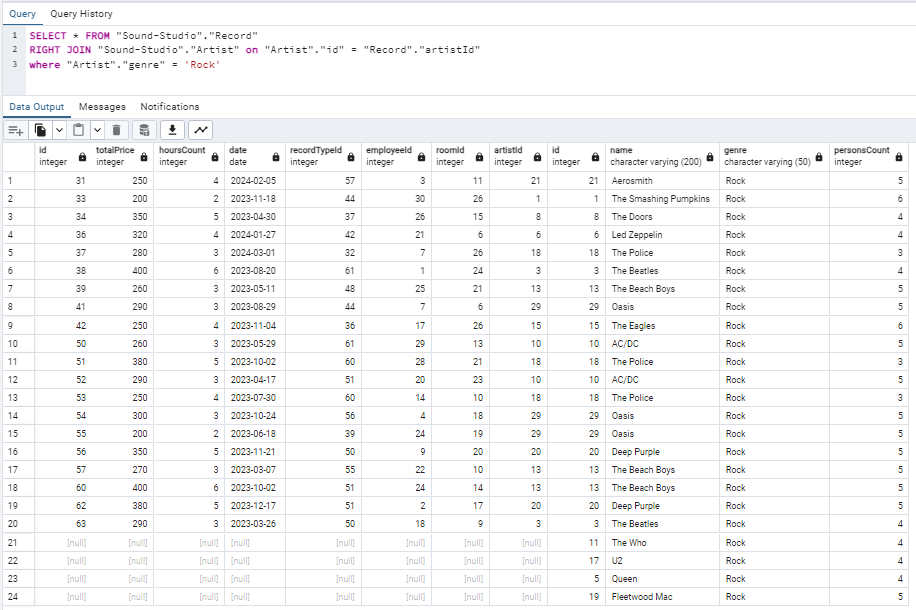
**

Рисунок 1.13.1 – SQL-cкрипт и результат выполнения

* 1. FULL OUTER JOIN (1)

Вывести все записи с присоединением комнаты, если данная запись не состоит ни в какой команте или данная комната не участвует ни в одной записи, на этих местах будет NULL.

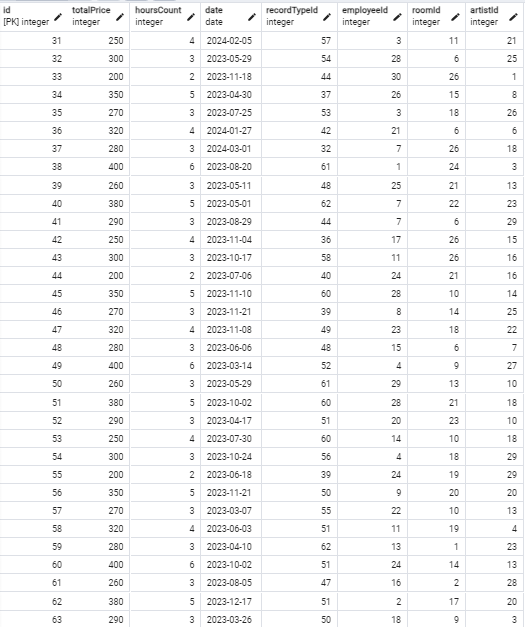


Рисунок 1.14.1 – Исходная таблица Record

На рисунке 1.14.1 показан код SQL-скрипта и результат его работы

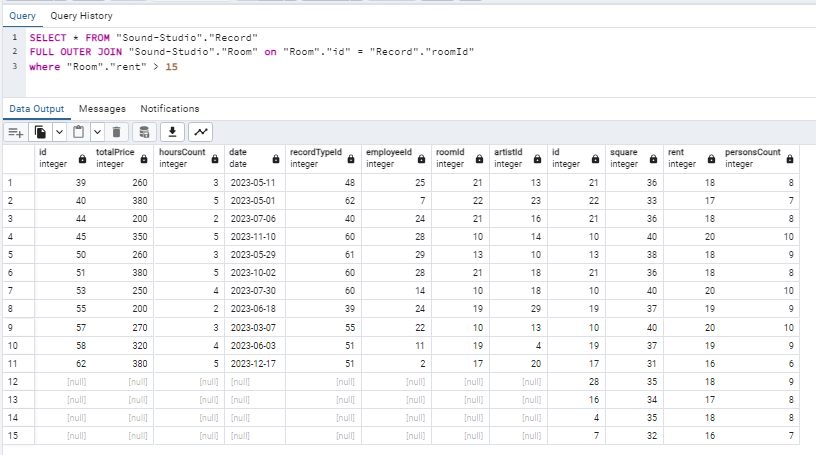


Рисунок 1.14.1 – SQL-cкрипт и результат выполнения

* 1. FULL OUTER JOIN (2)

Вывести все записи (Record) с присоединением сотрудников, если данную запись не обслуживает ни один сотрудник или сотрудник не обслуживает ни одну запись на этих местах будет NULL.

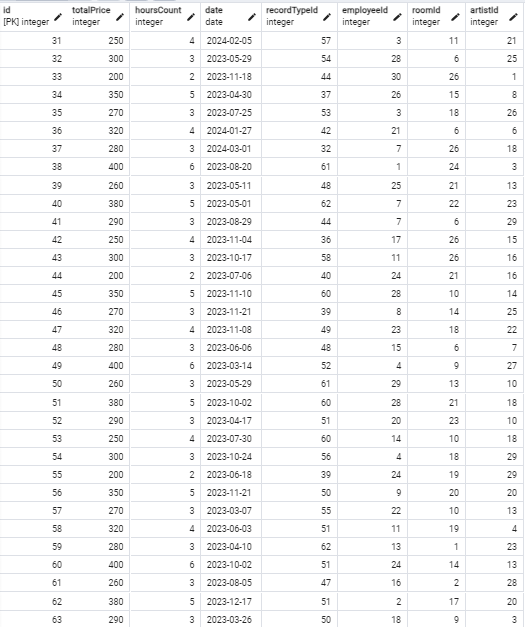


Рисунок 1.15.0 – Исходная таблица *Record*

На рисунке 1.15.1 показан код SQL-скрипта и результат его работы



Рисунок 1.15.1 – SQL-cкрипт и результат выполнения

# ЗАКЛЮЧЕНИЕ

В ходе данной лабораторной работы были изучены дополнительные возможности SQL-запросов в Postgres на базе «PgAdmin 4».

Изучено формирование SQL-операторов для выборки данных из таблиц с использованием многотабличных запросов, подзапросов и секций WHERE, ORDER BY и других.