Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОННИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Лабораторная работа №4

 Реализация SQL-запросов на простую выборку данных

Студент: В.С. Шевцов   
Преподаватель: Д.В. Куприянова

МИНСК 2024

СОДЕРЖАНИЕ

[1 SELECT FROM 3](#_Toc161571643)

[2 SELECT FROM WHERE 4](#_Toc161571644)

[3 SELECT FROM ORDER BY 5](#_Toc161571645)

[4 CROSS JOIN 8](#_Toc161571646)

[5 INNER JOIN 11](#_Toc161571647)

[6 LEFT OUTER JOIN 15](#_Toc161571648)

[7 RIGTH OUTER JOIN 18](#_Toc161571649)

[8 FULL OUTER JOIN 21](#_Toc161571650)

# SELECT FROM

Представляет собой простейшую выборку без ограничений. Например извлечем все данные из таблицы *Places.*

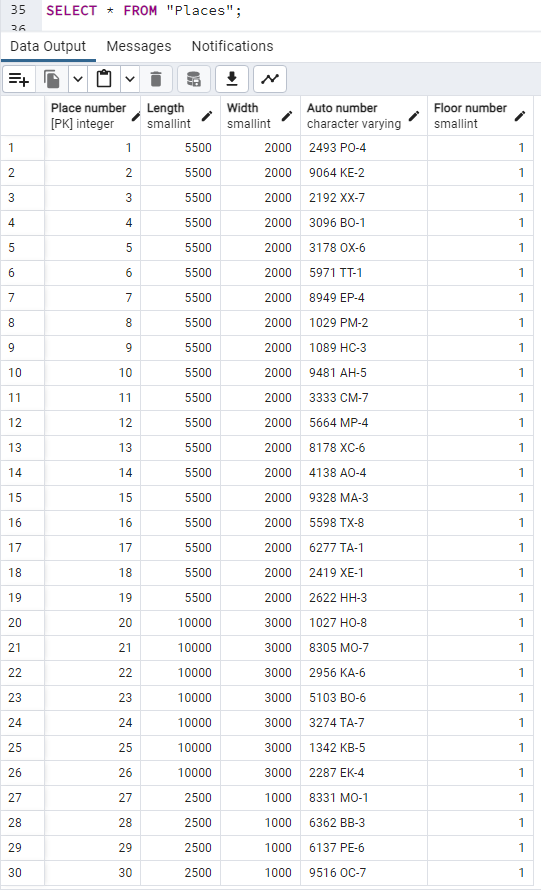


Рисунок 1.1 – Выборка из таблицы *Places*

# SELECT FROM WHERE

C помощью оператора WHERE можно ограничить выборку по некому условию. Например напишем запрос чтоб получить все автомобили массой больше 5 тонн.

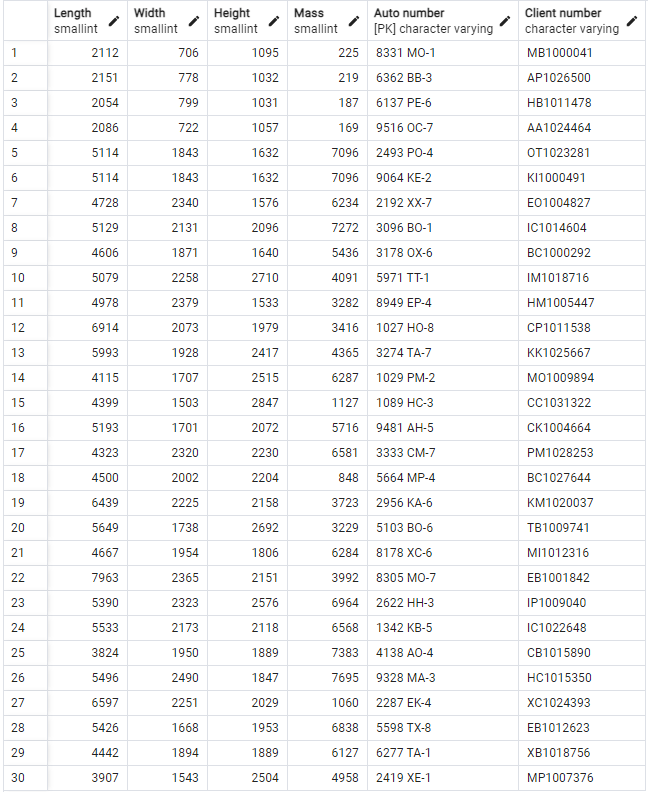


Рисунок 2.3 – Выборка из таблицы *Autos* без условия

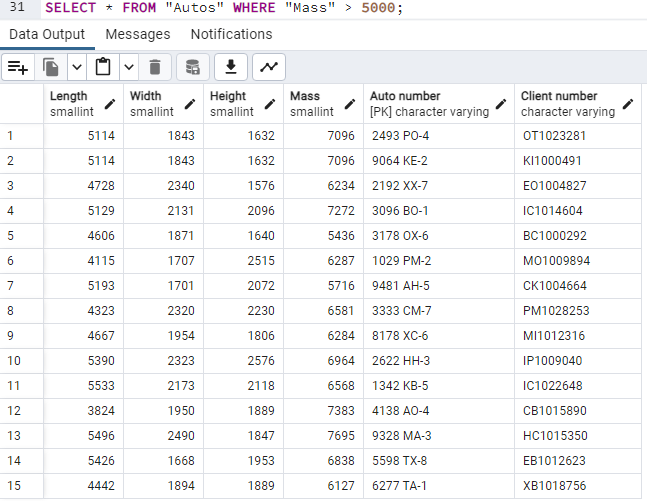


Рисунок 2.4 – Выборка из таблицы *Autos* с условием

# 3 SELECT FROM ORDER BY

Для упорядочивания результата по определенной колонке используется оператор ORDER BY. Напишем запрос который выведет нам всех клиентов со временм парковки больше 15 и отсортируем по возрастанию номера паспорта.

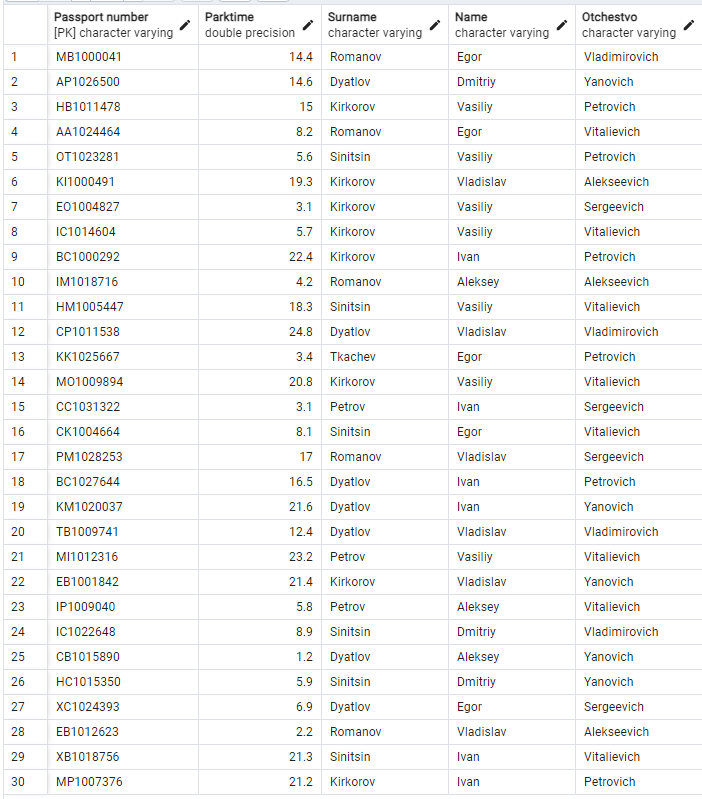


Рисунок 3.1 – Выборка из таблицы *Clients*  без сортировки

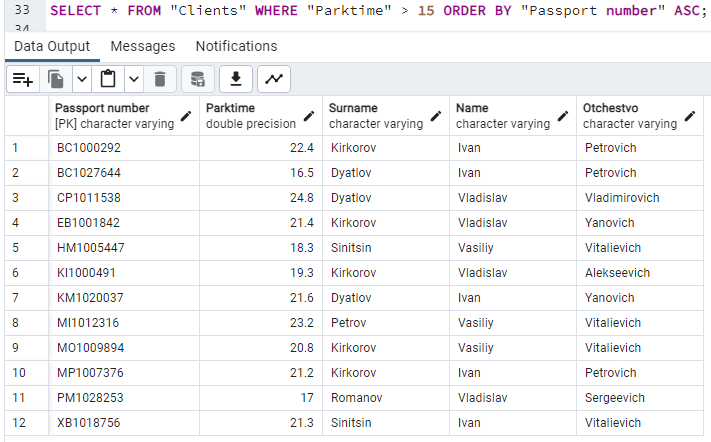


Рисунок 3.2 – Выборка из таблицы *Clients* с сортировкой по паспорту

Напишем второй запрос на выборку данных, в этот раз выведем всех клиентов и отсортируем по времени парковки.

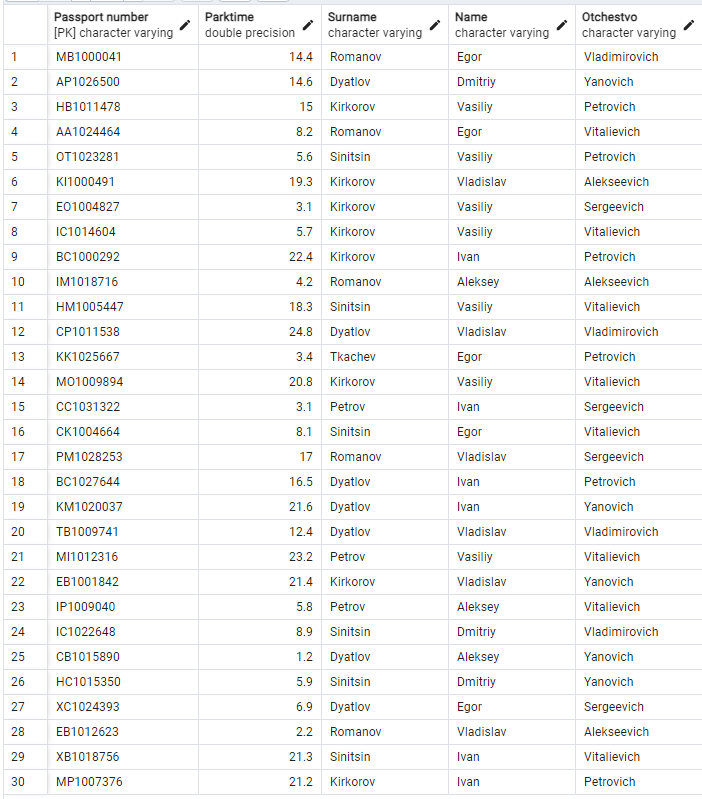


Рисунок 3.3 – Выборка из таблицы *Clients* без сортировки

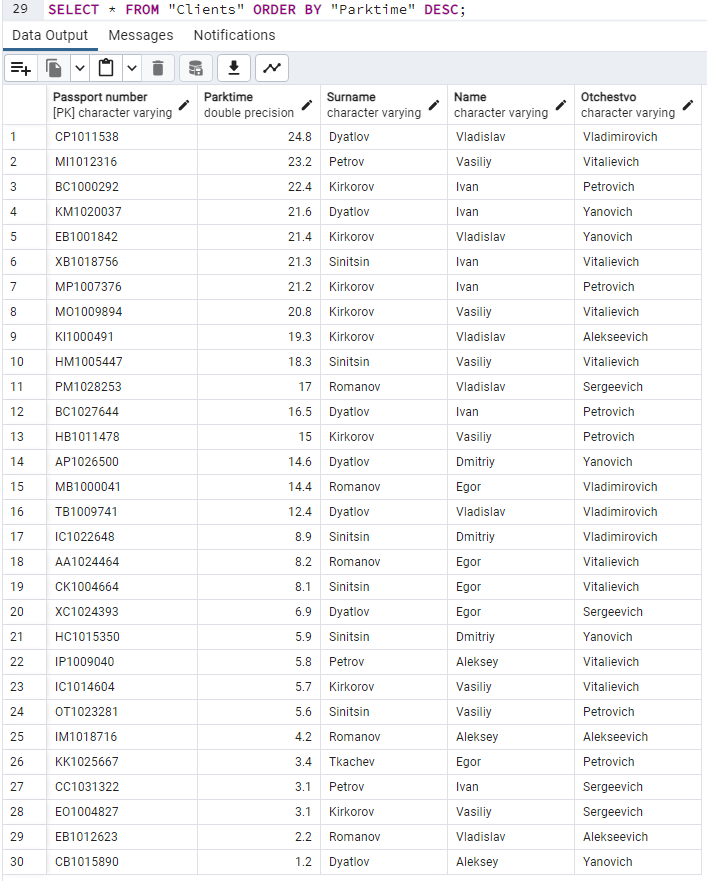


Рисунок 3.4 – Выборка из таблицы *Clients* с сортировкой по времени парковки

# CROSS JOIN

Оператор JOIN объединяет таблицы по определенному условию, CROSS – это разновидность соединения, которое на выходе дает декартово произведение строк обеих таблиц, то есть “каждый с каждым”, обычно этот оператор нужно ограничивать условиями. Напишем запрос объединения клиентов со временем парковки больше 22 и фамиоией Киркоров и мест, на которые они могут поставить свои авто.

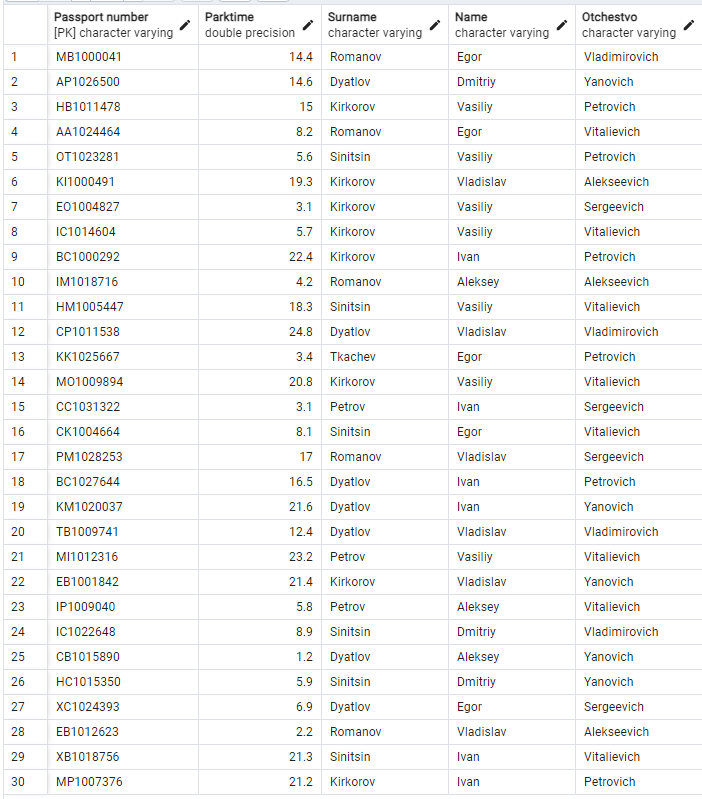


Рисунок 4.1 – Выборка из таблицы *Clients*

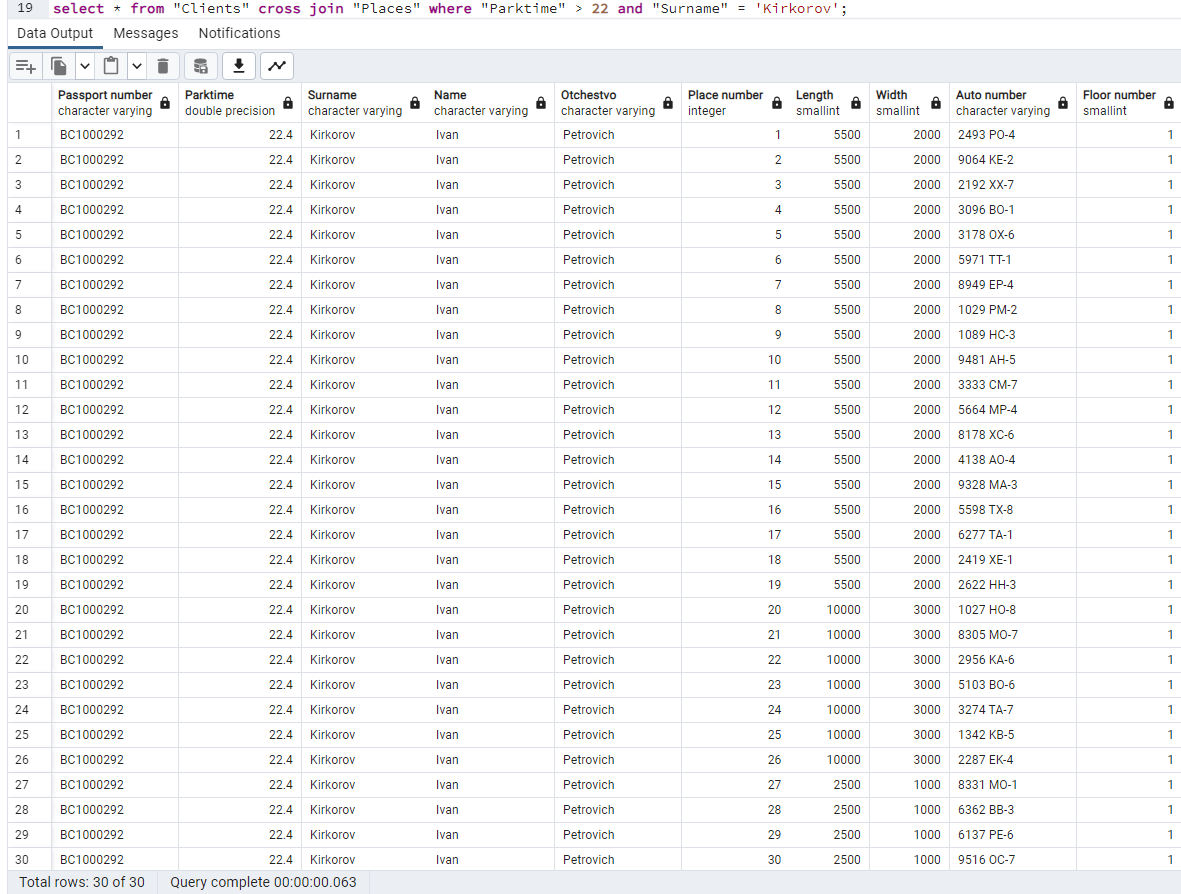
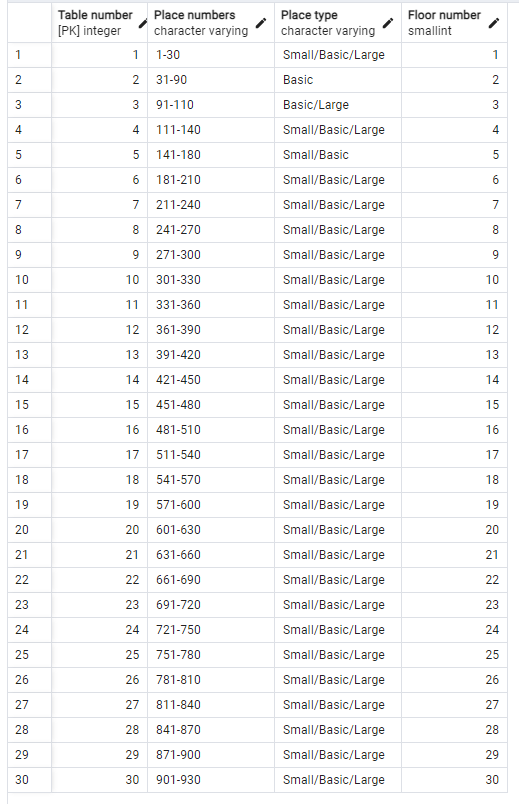


Рисунок 4.2 – Объединение таблиц *Clients* и *Places*

Также напишем запрос, который покажет, к какому табло могут обращаться клиенты со временем парковки больше 10 на втором этаже.



Рисунк 4.3 - Выборка из таблицы *Tables*

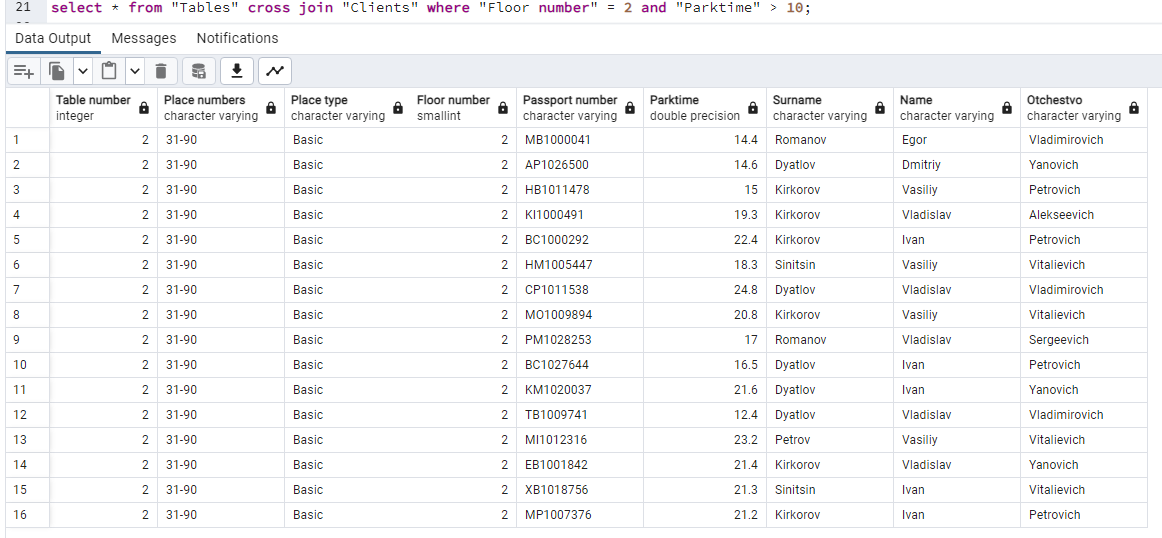


Рисунок 4.4 – Объединение таблиц *Tables* и *Clients*

# 5 INNER JOIN

INNER JOIN находит пересечение таблиц по определенному условию. Например напишем запрос, который вернет номер авто каждого пользователя.

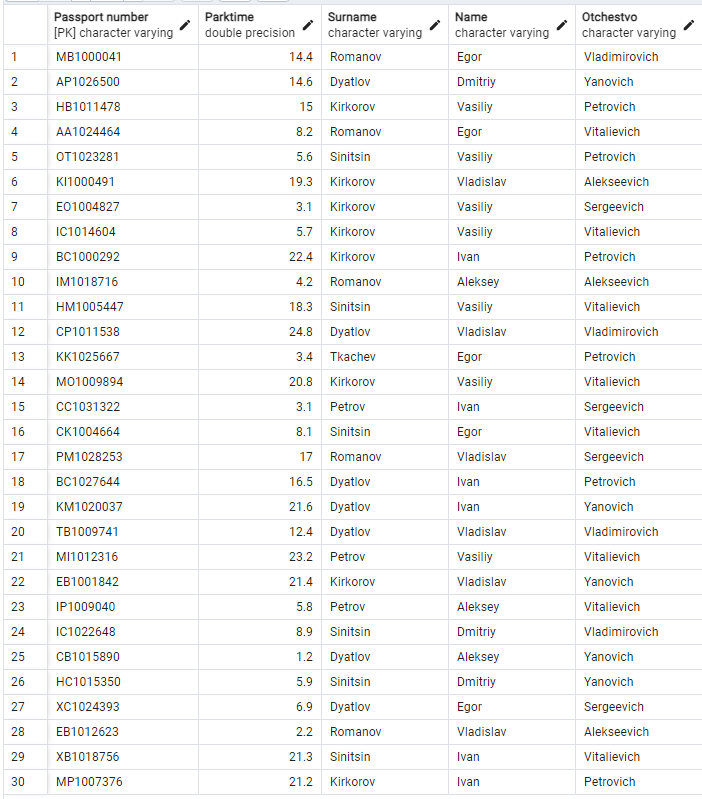


Рисунок 5.1 – Выборка из таблицы *Clients*

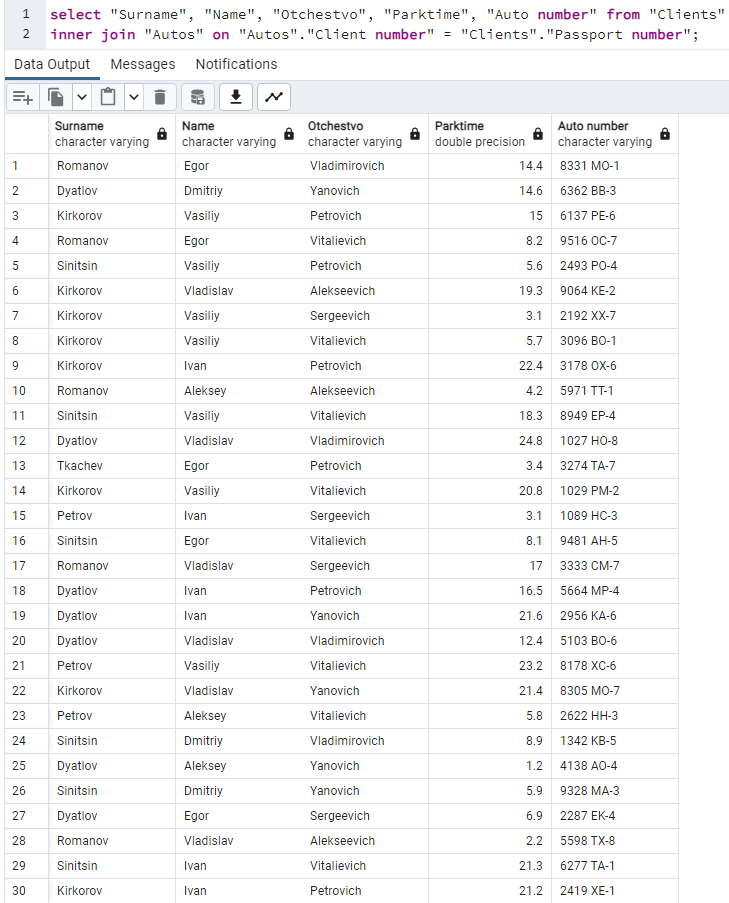


Рисунок 5.2 – Пересечение таблиц *Clients* и *Autos*

Далее напишем запрос который номер места для каждого авто

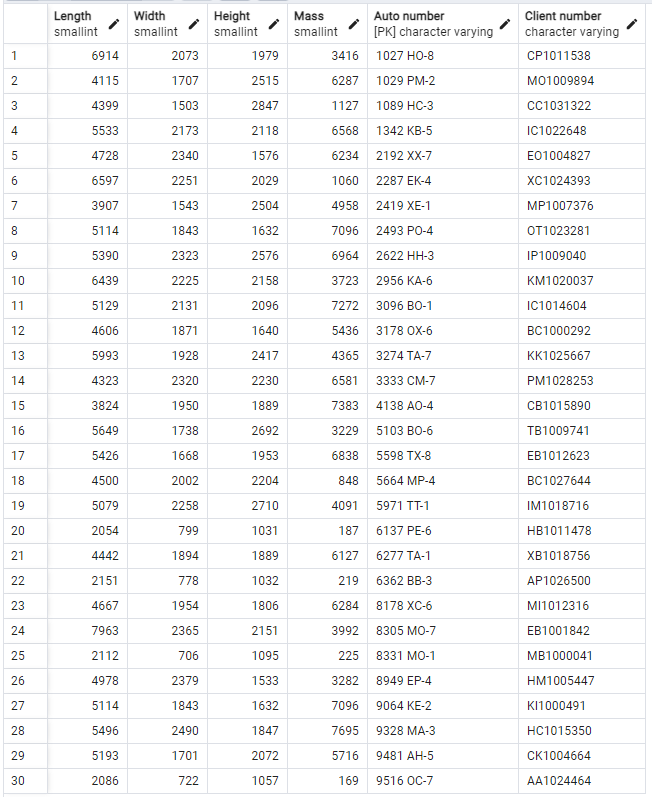


Рисунок 5.3 – Выборка таблицы *Autos*

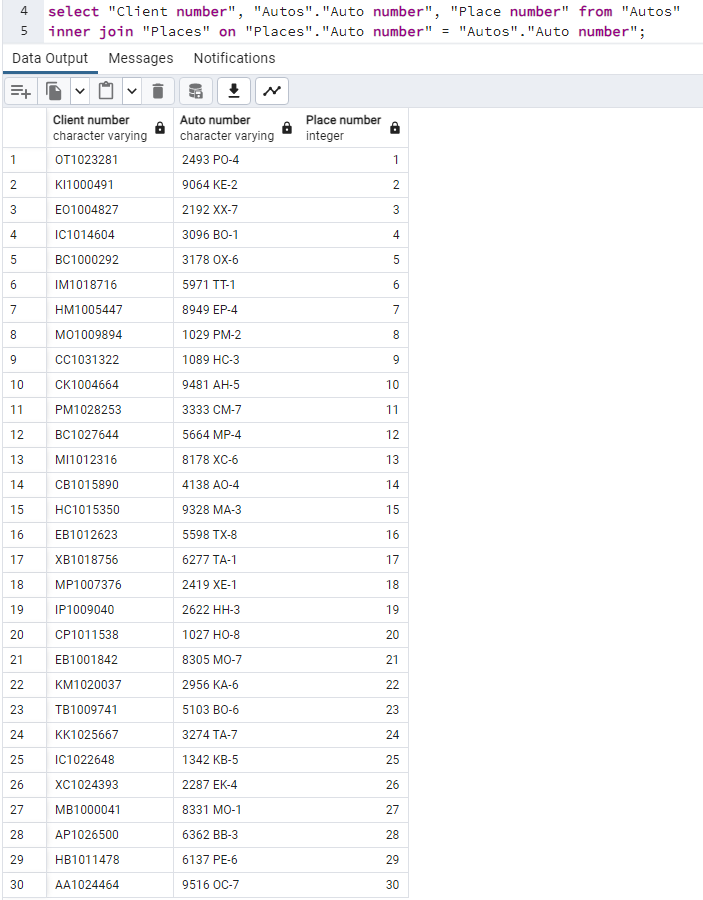


Рисунок 5.4 – Пересечение таблиц *Autos* и *Places*

# 6 LEFT OUTER JOIN

OUTER JOIN объединяет одну таблицу (LEFT/RIGHT) с другой таблицей по условию, если условие не выполняется в записи будет NULL. Напишем запрос, который покажет, в какие лифты вмещаются авто больше 7.2 тонны.

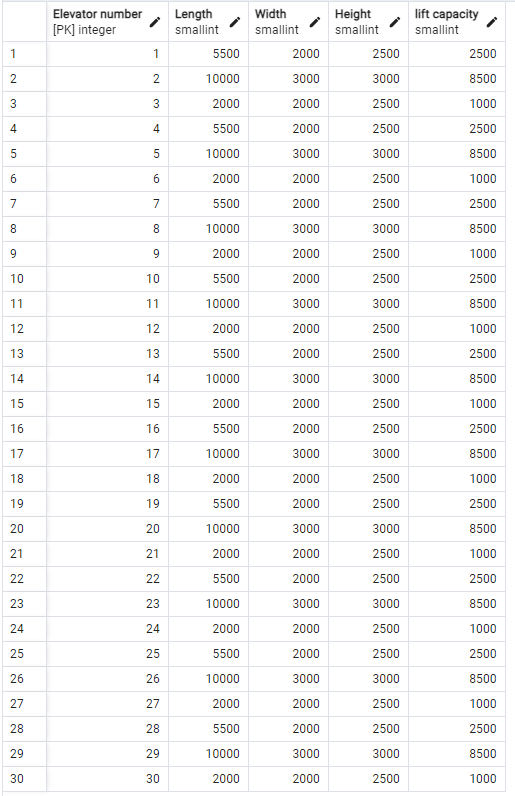


Рисунок 6.1 – Выборка из таблицы Elevators

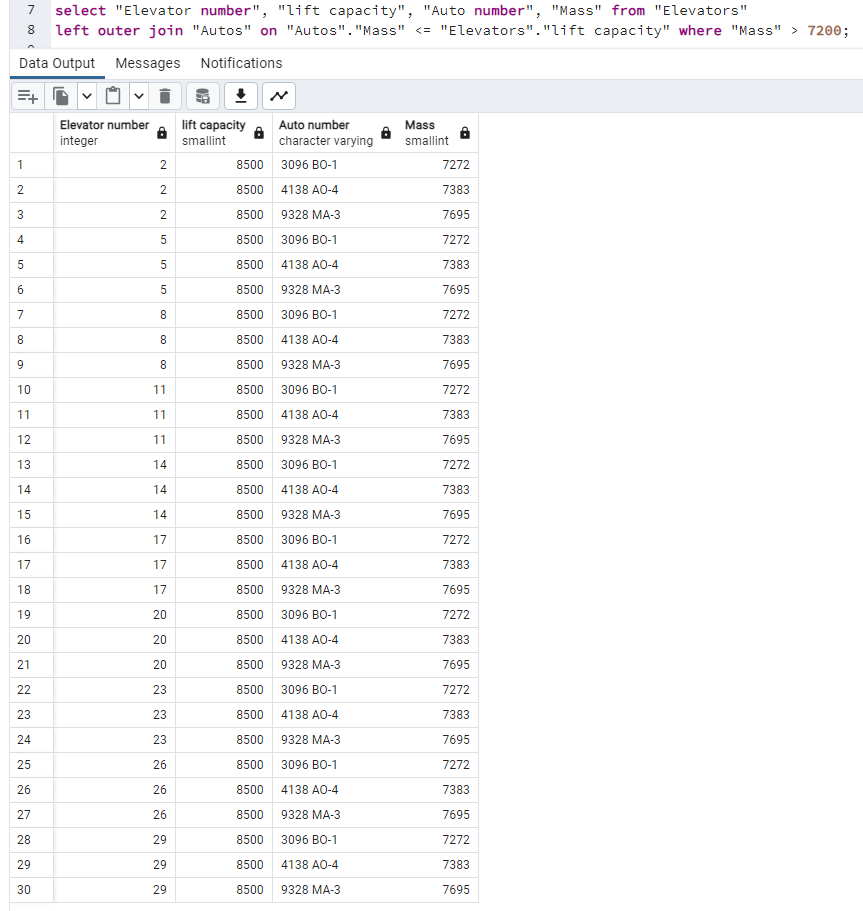


Рисунок 6.2 – Объединение таблицы *Elevators* и *Autos*Также напишем запрос, который объединит таблицы user и tariff.

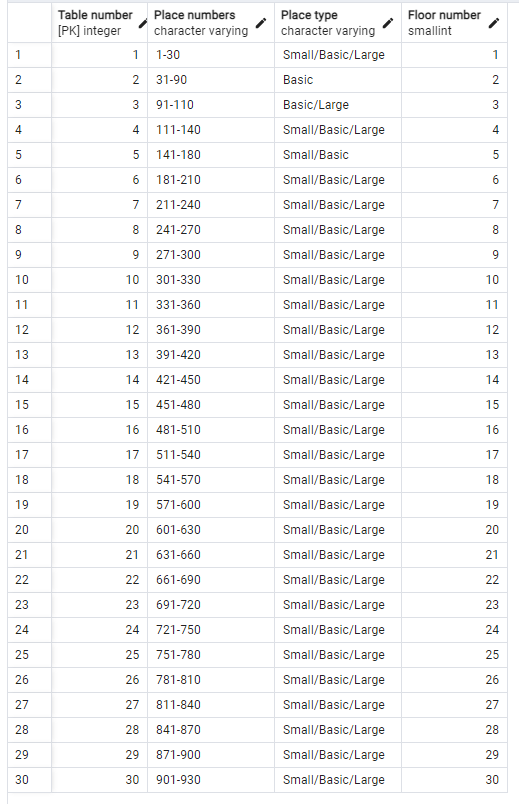


Рисунок 6.3 – Выборка из таблицы *Tables*

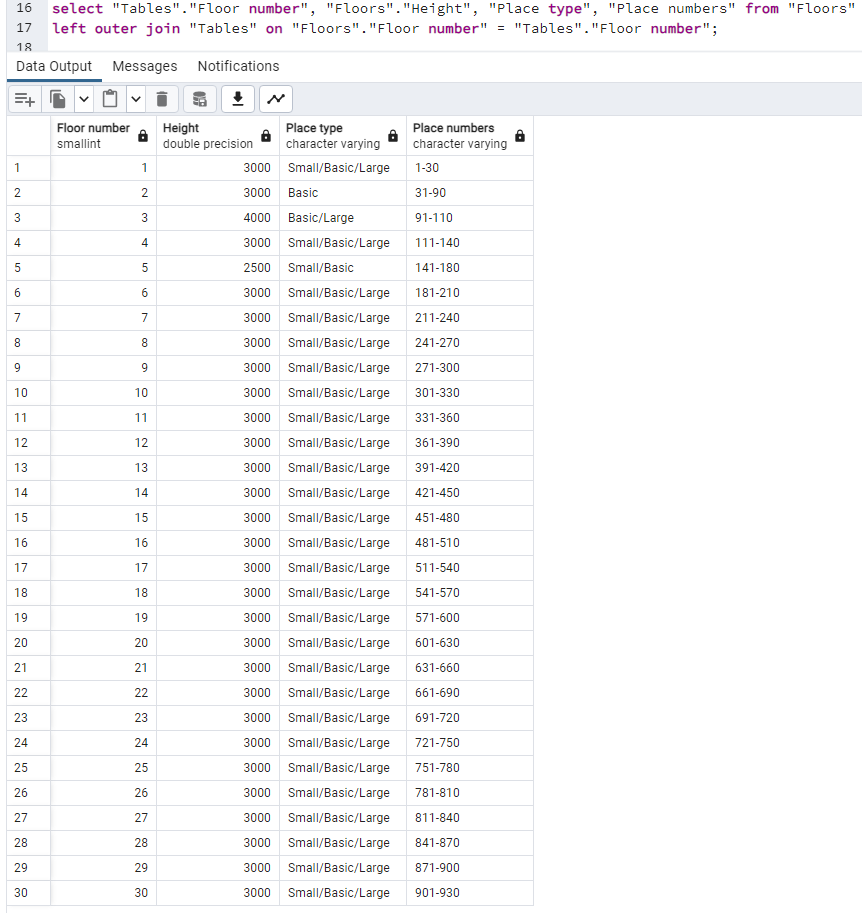


Рисунок 6.4 – Объединение таблицы *Floors* и *Tables*

# 7 RIGHT OUTER JOIN

RIGHT JOIN к строкам таблицы пытается присоединить строки второй таблицы, если это не выходит в таблице будет NULL.

Напишем запрос который присоединит *Elevators* к таблице *Autos*

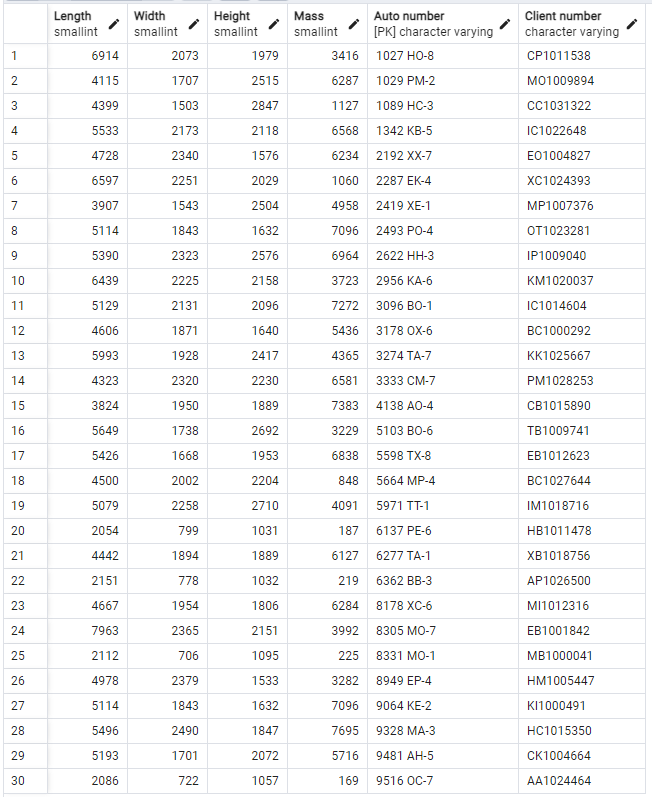


Рисунок 7.1 – Выборка из таблицы *Autos*

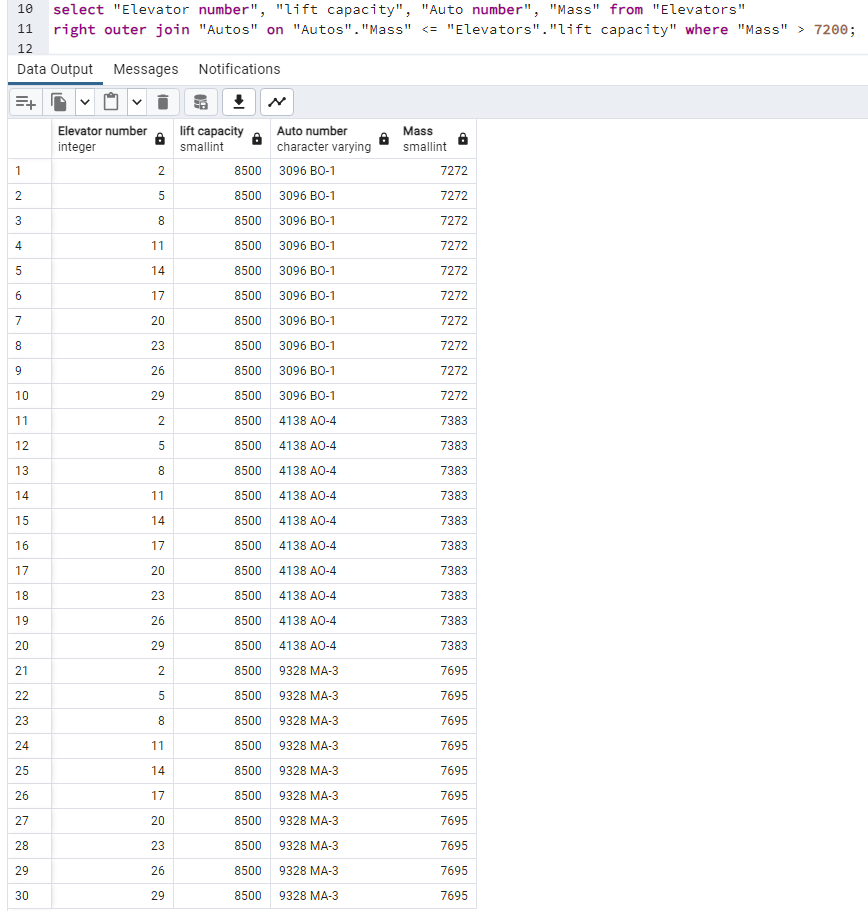
****

Рисунок 7.2 – Объединение таблицы *Elevators* и *Autos*

Также напишем запрос, который присоединит *Floors* к таблице *Tables.*

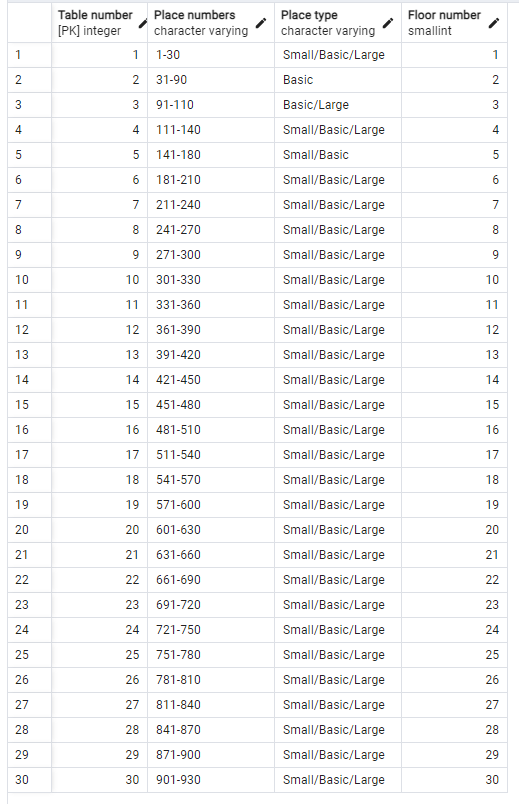


Рисунок 7.3 – Выборка из таблицы *Tables*

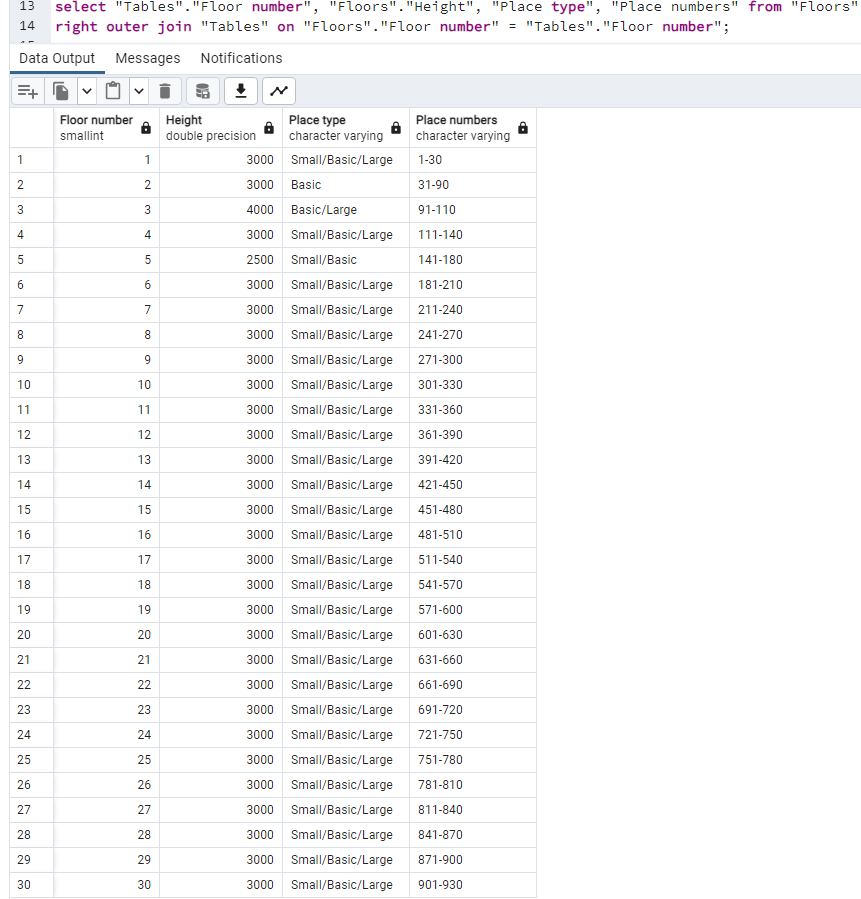


Рисунок 7.4 – Объединение таблицы *Tables* и *Floors*

# 8 FULL OUTER JOIN

FULL OUTER JOIN – это сочетание LEFT и RIGHT, по факту это CROSS JOIN с условием ON, который может дать NULL как в исходной таблице, так и в присоединяемой.

Напишем запрос, который соединит таблицы *Places* и *Autos*

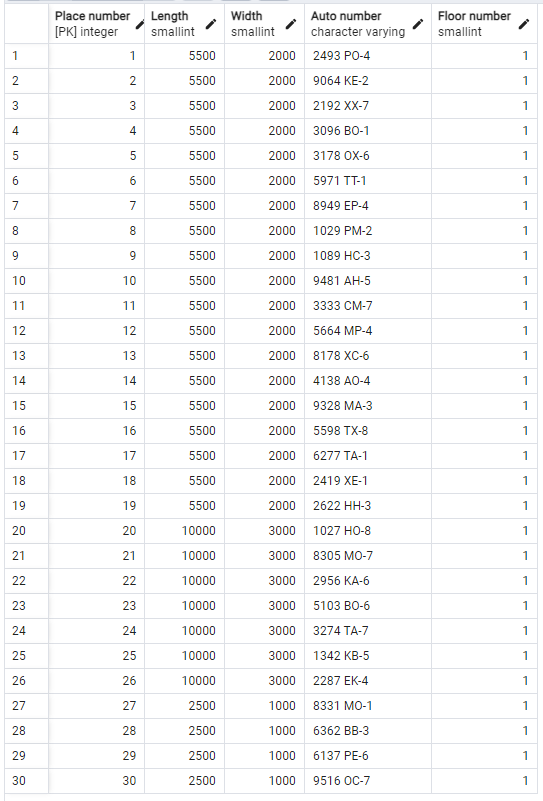
****

Рисунок 8.1 – Выборка из таблицы *Places*

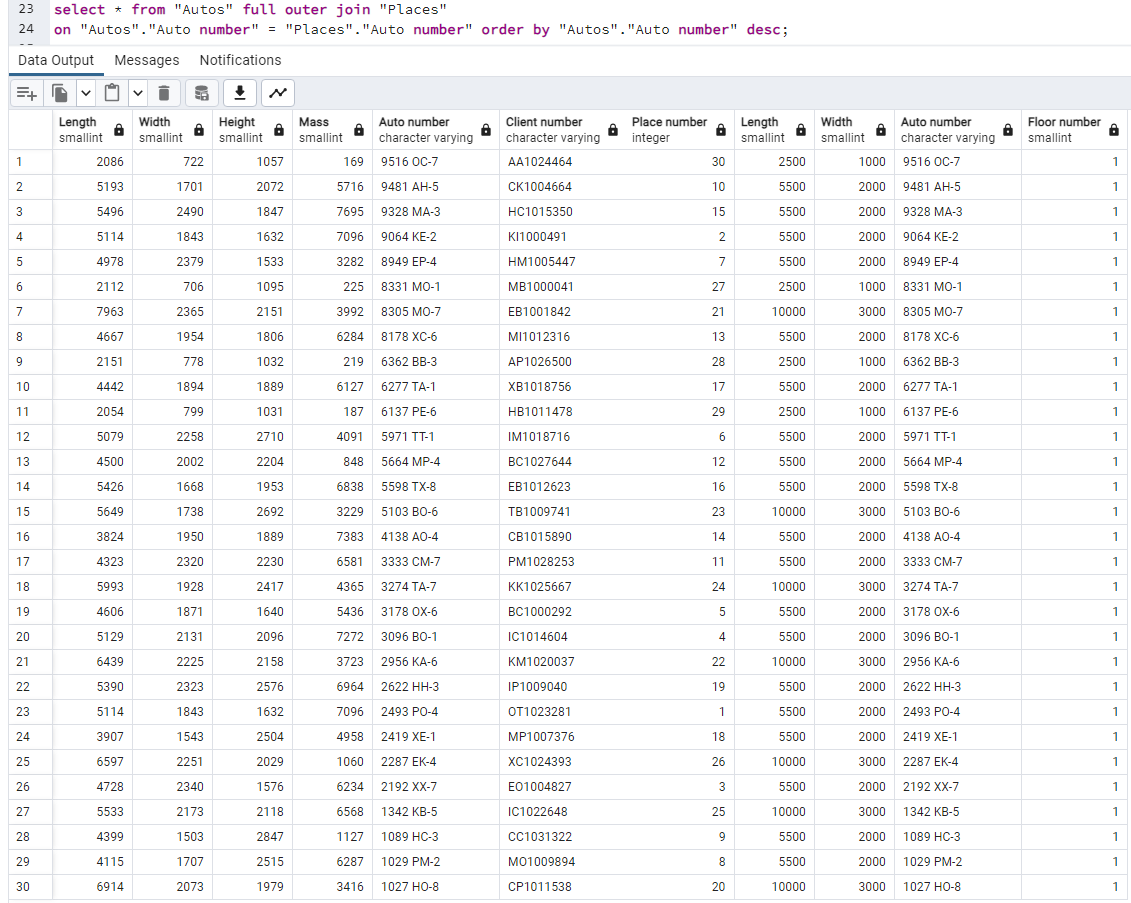
****

Рисунок 8.2 – Объединение таблицы *Places и Autos*

Также напишем запрос на объединение *Clients* и *Autos*

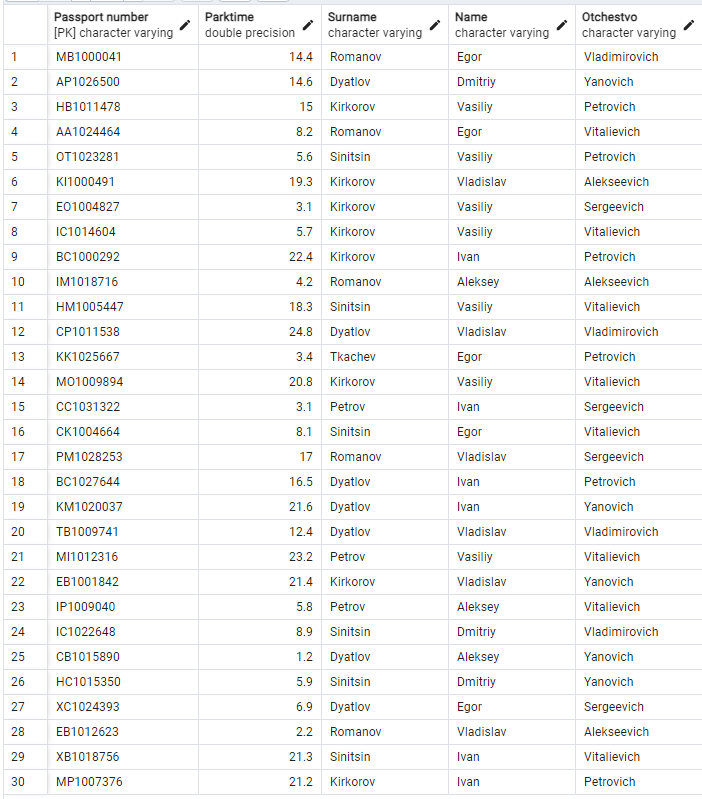


Рисунок 8.3 – Выборка из таблицы *Clients*

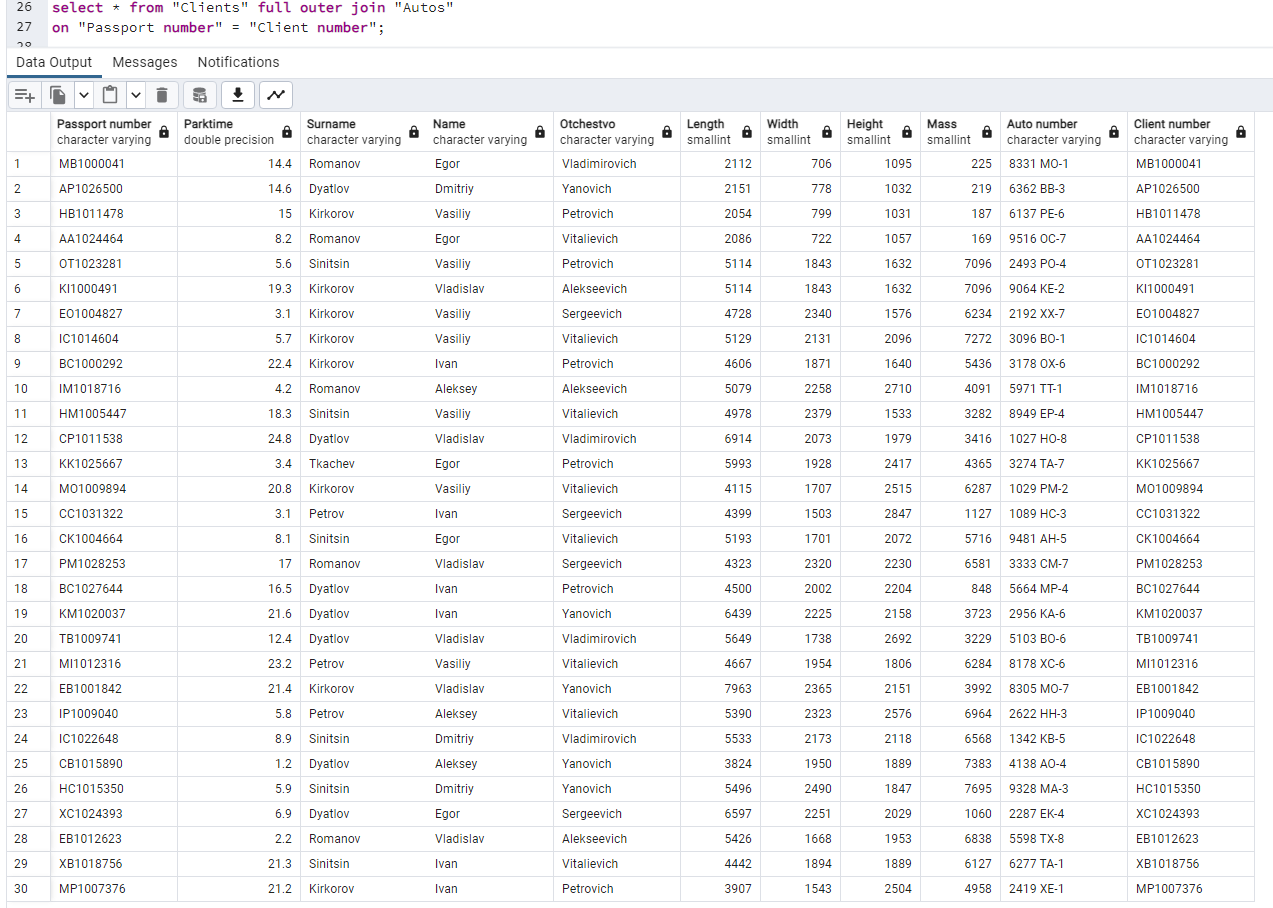


Рисунок 8.4 – Объединение таблицы *Clients* и *Autos*