ОТЧЕТ

по лабораторной работе № 5

«Разработка сложных запросов к базе данных. Запросы на основе нескольких таблиц. Коррелированные сложенные подзапросы»

по дисциплине

«ТЕОРИЯ БАЗ ДАННЫХ»

Выполнил студент группы ИС/б-22о

Горбенко К.Н.

Проверила:

Лебедева М.А.

* 1. ЦЕЛЬ РАБОТЫ

Выполнение лабораторной работы преследует следующие цели:

1. изучить способы получения информации из нескольких таблиц;
2. записать запросы, демонстрирующие выборки из нескольких таблиц с использованием оператора JOIN и без него;
3. изучить способы выполнения и принцип действия рекурсивных запросов;
4. научиться использовать вложенные подзапросы;
5. ознакомиться с возможностями построения вложенных коррелированных подзапросов с применением кванторов;
6. ознакомиться с возможностью формирования отчетов в клиентском приложении.
   1. ЗАДАНИЕ НА РАБОТУ

Проанализировать созданную в л/р №4 базу данных.

Сформулировать и выполнить с помощью SQL-редактора PhpMyAdmin запросы, соединяющие две таблицы с помощью JOIN и без него.

Сформулировать и выполнить с помощью SQL-редактора PhpMyAdmin запросы, соединяющие более чем две таблицы с помощью JOIN и без него.

Продемонстрировать с помощью SQL-редактора PhpMyAdmin следующие возможности SQL:

использование псевдонимов на примере рекурсивного запроса;

запрос с подзапросом;

использование агрегатных функций в подзапросе;

подзапросы, возвращающие единственное и множественные значения;

подзапросы, использующие вычисление;

использование подзапросов в HAVING;

Ознакомится с принципом и продемонстрировать работу коррелированных подзапросов:

привести пример соединения таблицы со своей копией;

привести пример коррелированного запроса, использующего две разные таблицы;

привести пример запроса с оператором EXIST;

привести пример запроса с оператором ALL;

привести пример запроса с оператором ANY.

Разработать и выполнить с помощью SQL-редактора PhpMyAdmin обязательные запросы, заданные вариантом (таблица 1).

Таблица 1 – Запросы к базе данных

|  |  |
| --- | --- |
| № варианта | Запросы к базе данных |
| 15 | 1. Вывести все телефоны фирмы «Оптима Крым»; 2. Вывести названия фирмы, которая не имеет контрагентов; 3. Вывести название фирмы, занимающей помещение с максимальной площадью. |

Заданная вариантом № 15 предметная область:

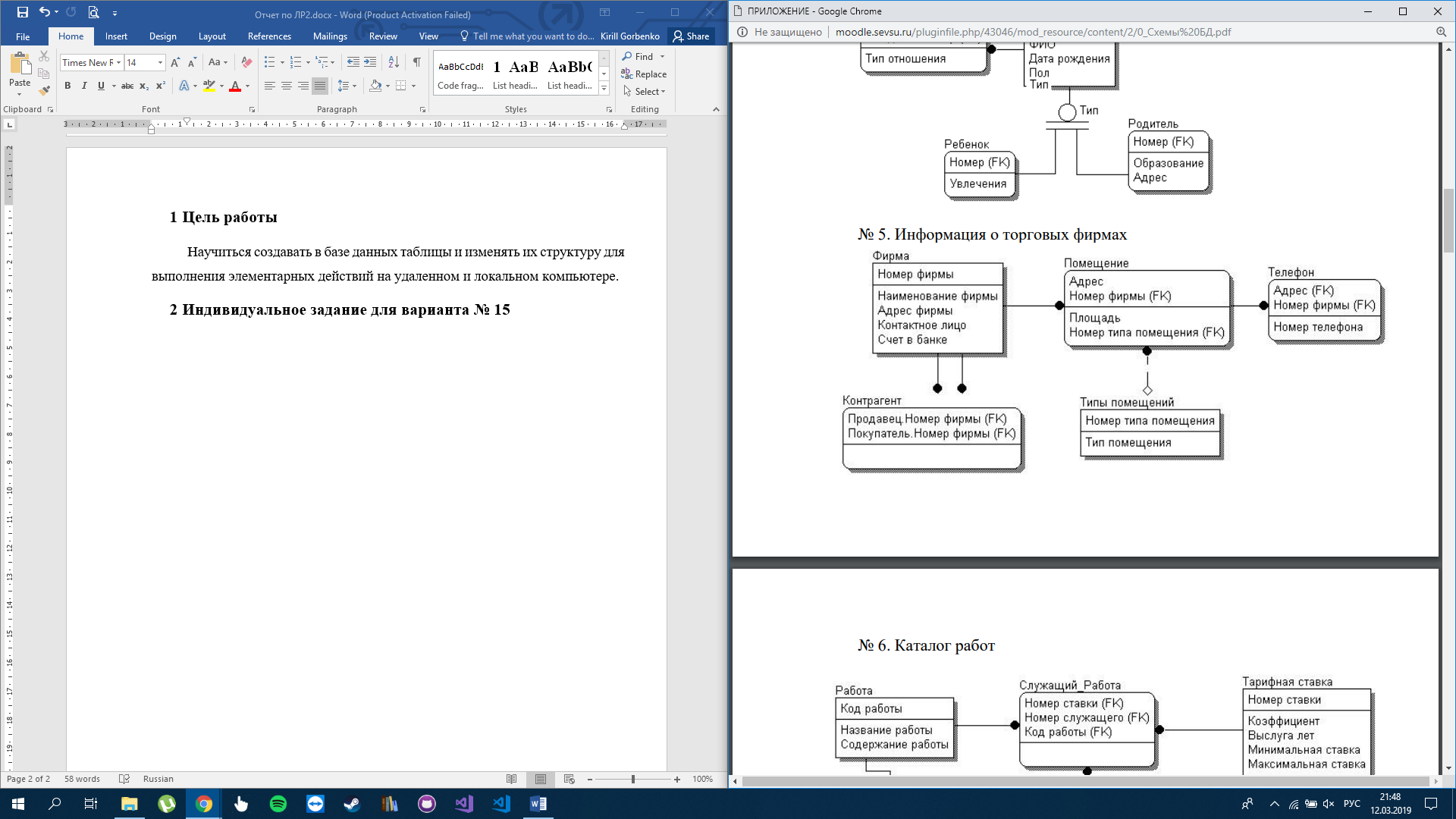


Рисунок 1 – Схема № 5 «Информация о торговых фирмах»

* 1. ХОД РАБОТЫ
     1. Запросы для объединения двух таблиц
        1. Объединение с помощью условия

Вывести имена и номера компаний, которые имеют помещения с площадью менее 100.

SELECT Name, company.Number, Area FROM company, room

WHERE company.Number = room.Number AND Area < 100

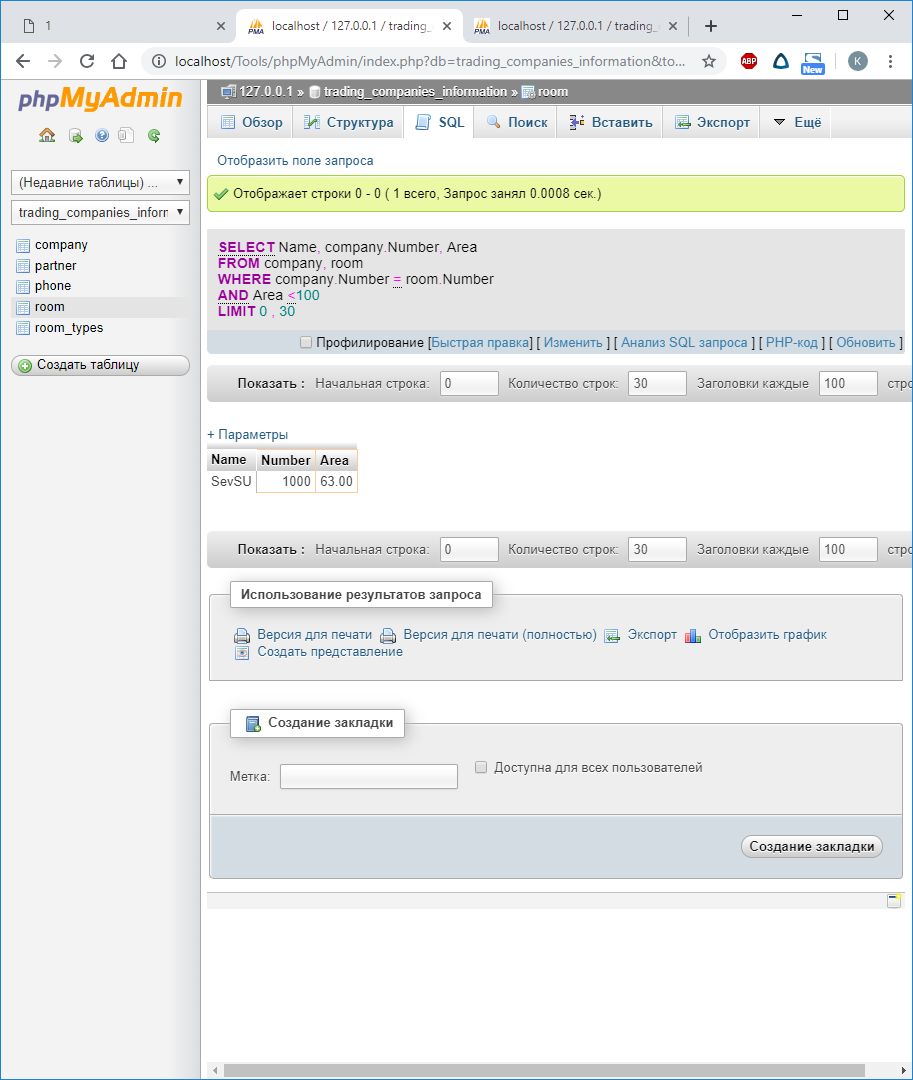


Рисунок 2 – Результат выполнения запроса объединения данных двух таблиц с помощью условия

* + - 1. Объединение с помощью команды JOIN

Выполним тот же запрос с помощью оператора JOIN:

SELECT Name, room.Number, Area FROM

company JOIN room on company.Number = room.Number

WHERE Area < 100

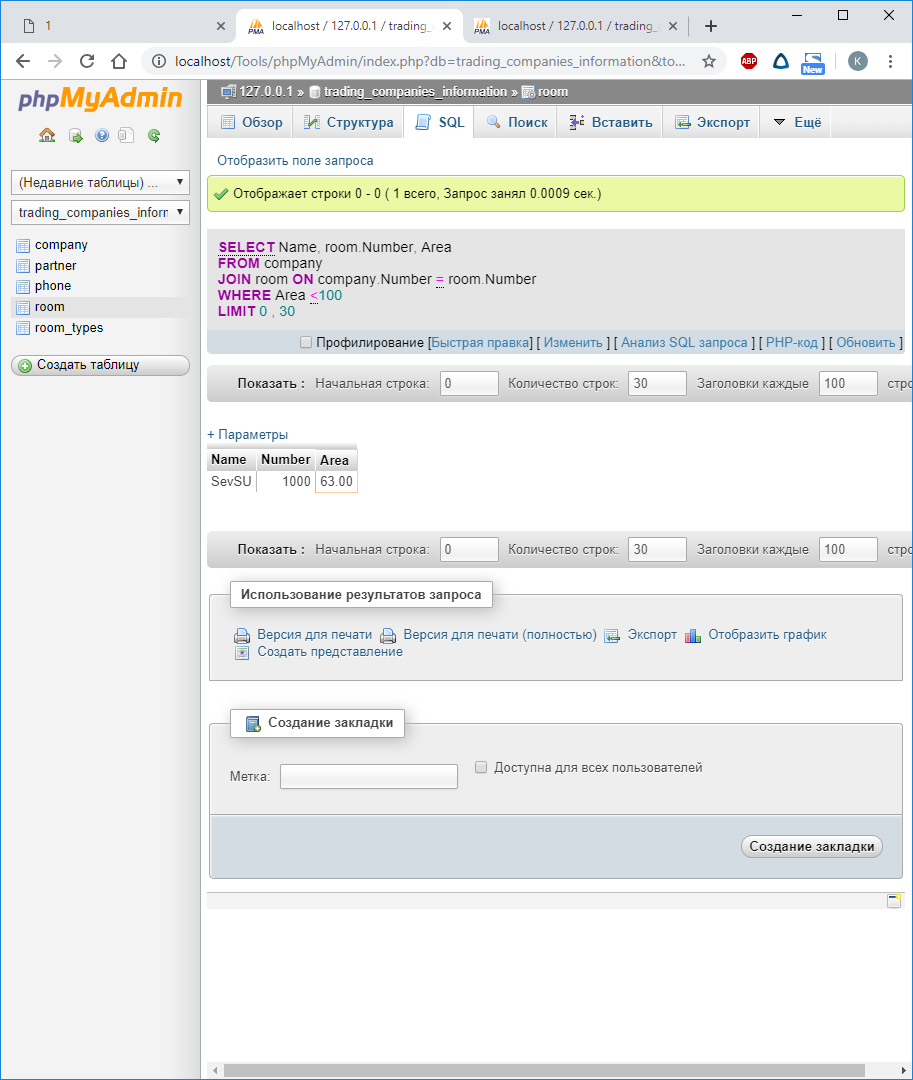


Рисунок 3 – Результат выполнения запроса объединения данных двух таблиц с помощью оператора JOIN

* + 1. Запросы для объединения более двух таблиц
       1. Объединение с помощью условия

Выведем имена фирм, их номера и адреса и мобильные телефоны тех фирм, у которых есть помещения и номера телефонов.

SELECT Name, company.Number, room.Address, Phone\_number FROM company, room, phone

WHERE company.Number = room.Number AND company.Number = phone.Number AND room.Address = phone.Address

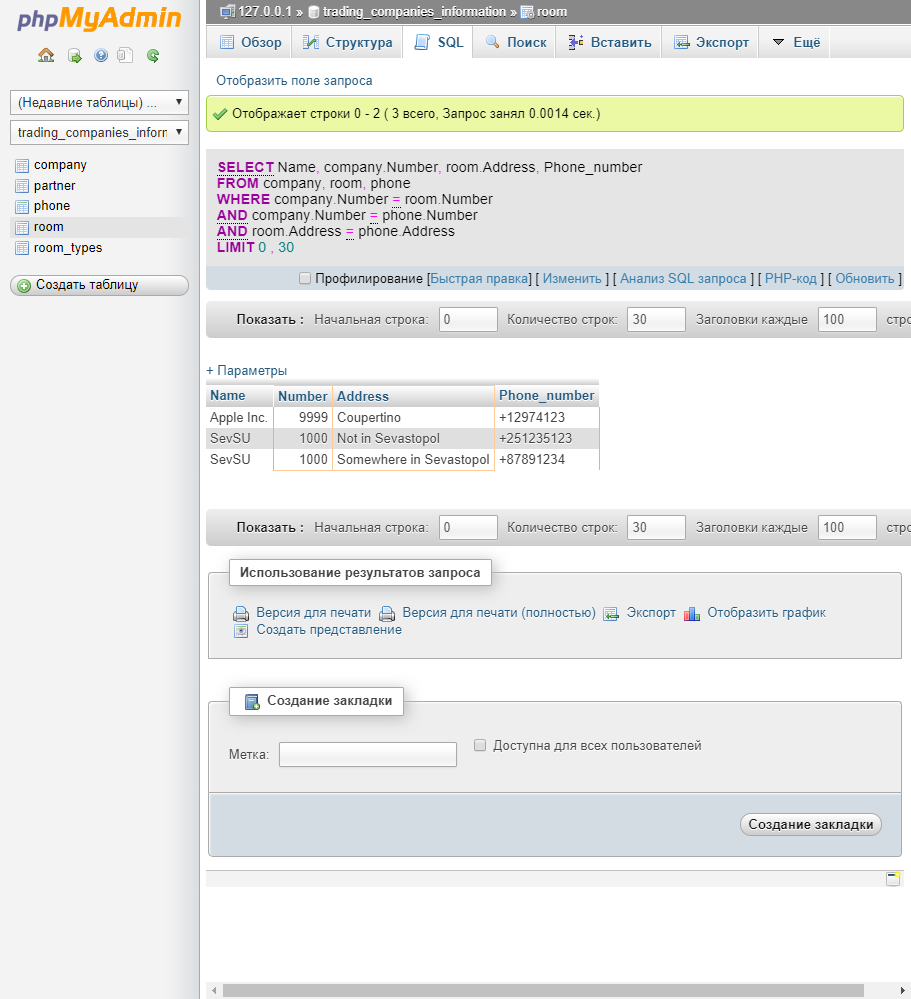


Рисунок 4 – Результат выполнения запроса объединения трех таблиц с помощью условия

* + - 1. Объединение с помощью оператора JOIN

SELECT Name, company.Number, room.Address, Phone\_number FROM (company JOIN room ON company.Number = room.Number) JOIN phone ON room.Number = phone.Number AND room.Address = phone.Address

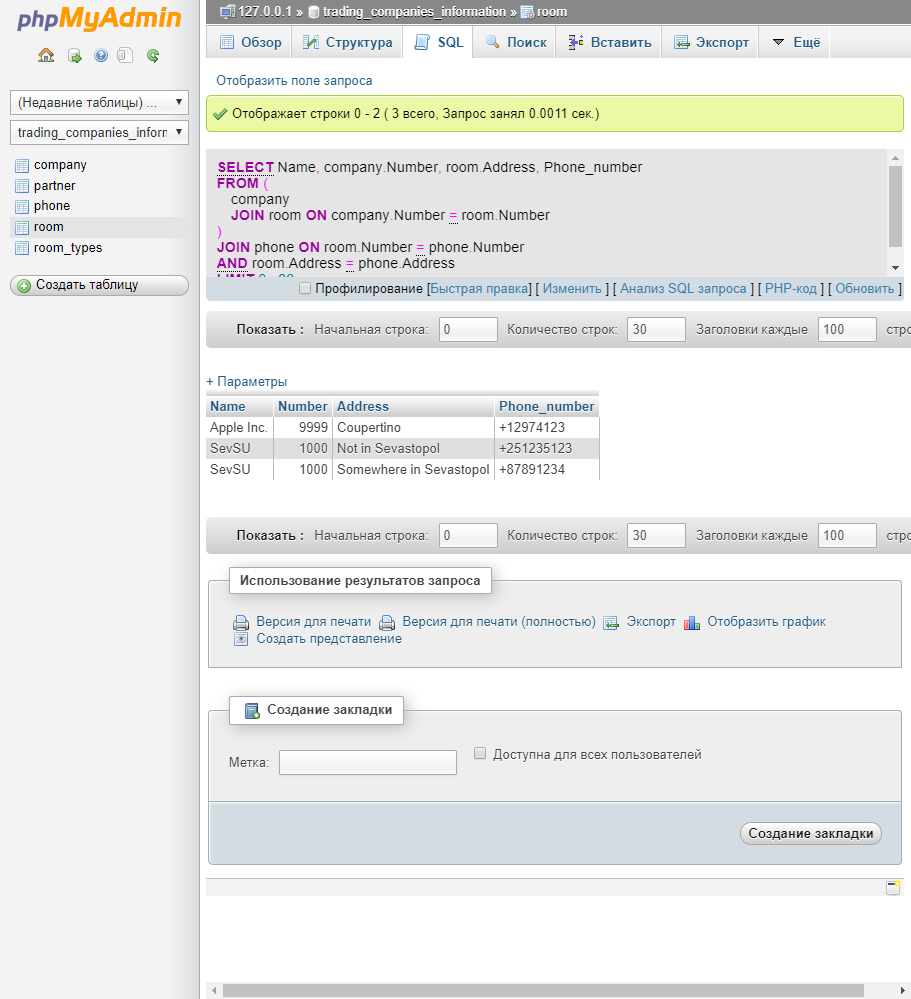


Рисунок 5 – Результат выполнения запроса объединения данных трех таблиц с помощью оператора JOIN

* + 1. Использование рекурсивных запросов

Вывести пары номеров фирм, у которых есть несколько помещений одного типа:

SELECT DISTINCT one.Number, two.Number

FROM room one, room two

WHERE one.Room\_type\_number = two.Room\_type\_number

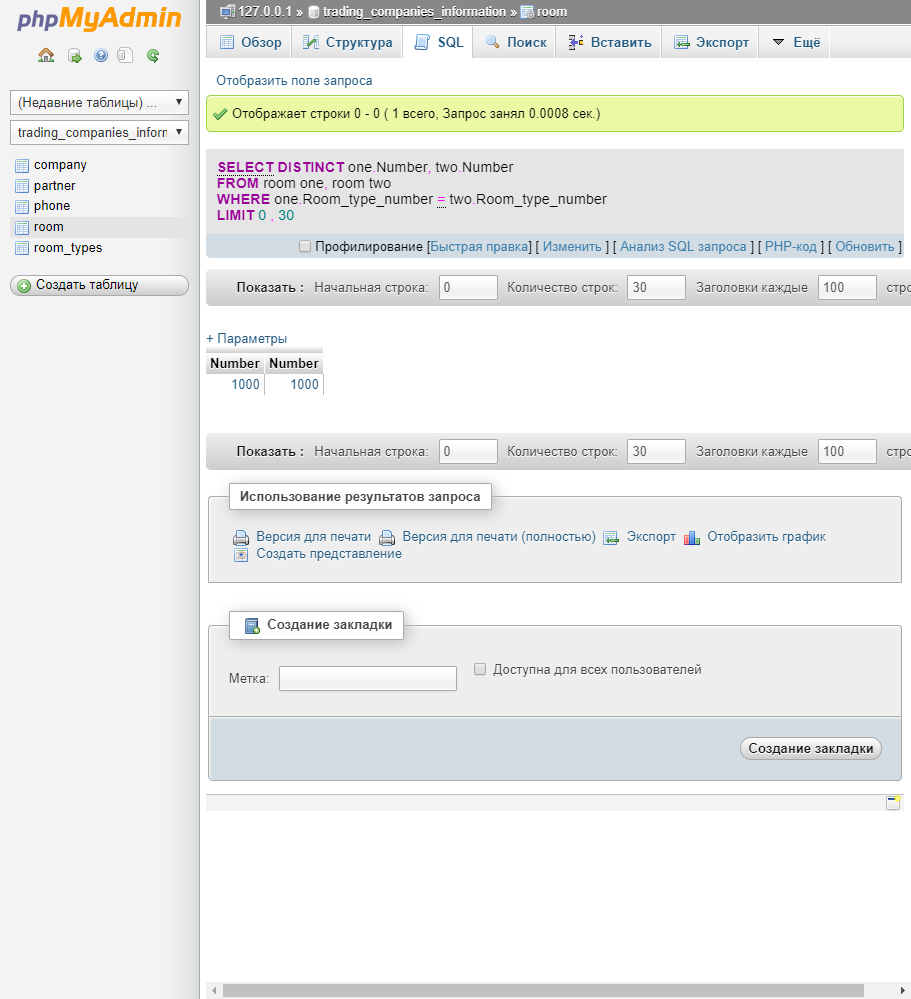


Рисунок 6 – Результат выполнения рекурсивного запроса

* + 1. Запрос с подзапросом, возвращающим единственное значение, и использование агрегатных функций

Вывести все данные о помещениях, площадь которых больше средней площади помещений.

SELECT \* FROM room

WHERE Area > (SELECT AVG(Area) FROM room)

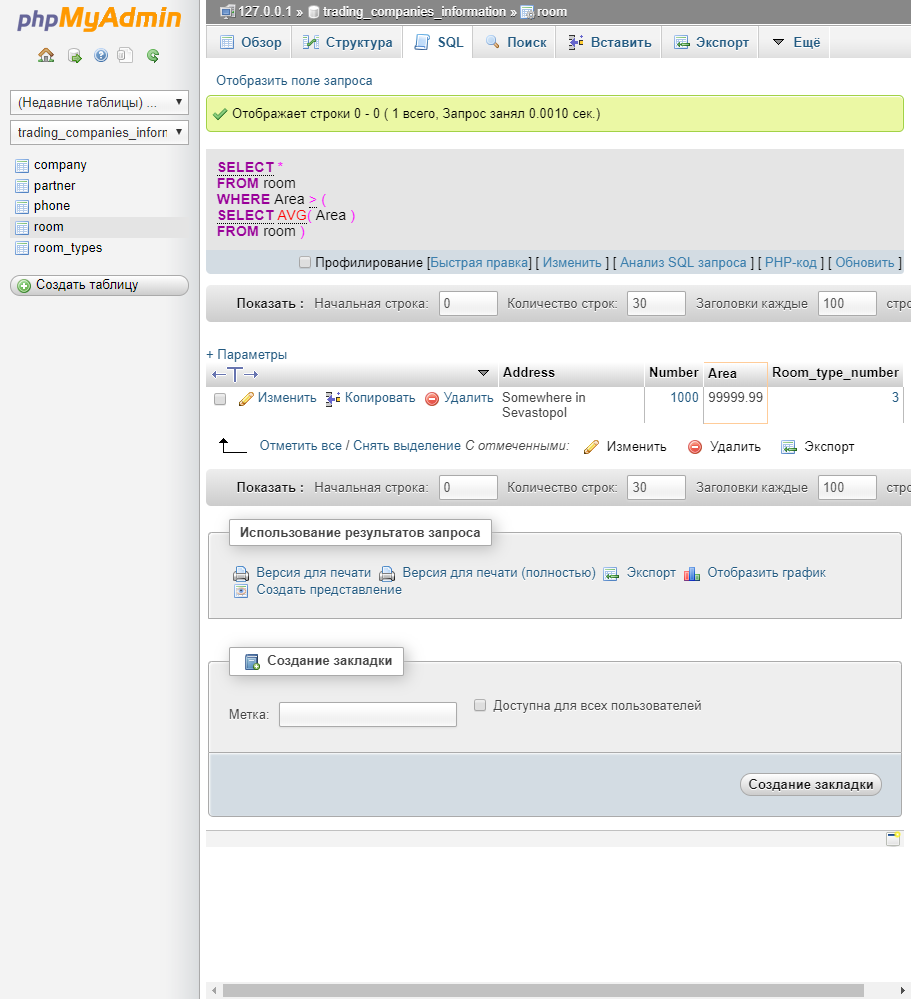


Рисунок 7 – Результат выполнения запроса с подзапросом

* + 1. Запрос с подзапросом, возвращающим множественные значения

SELECT \* FROM room

WHERE Number IN (SELECT Number FROM room WHERE Room\_type\_number = 3)

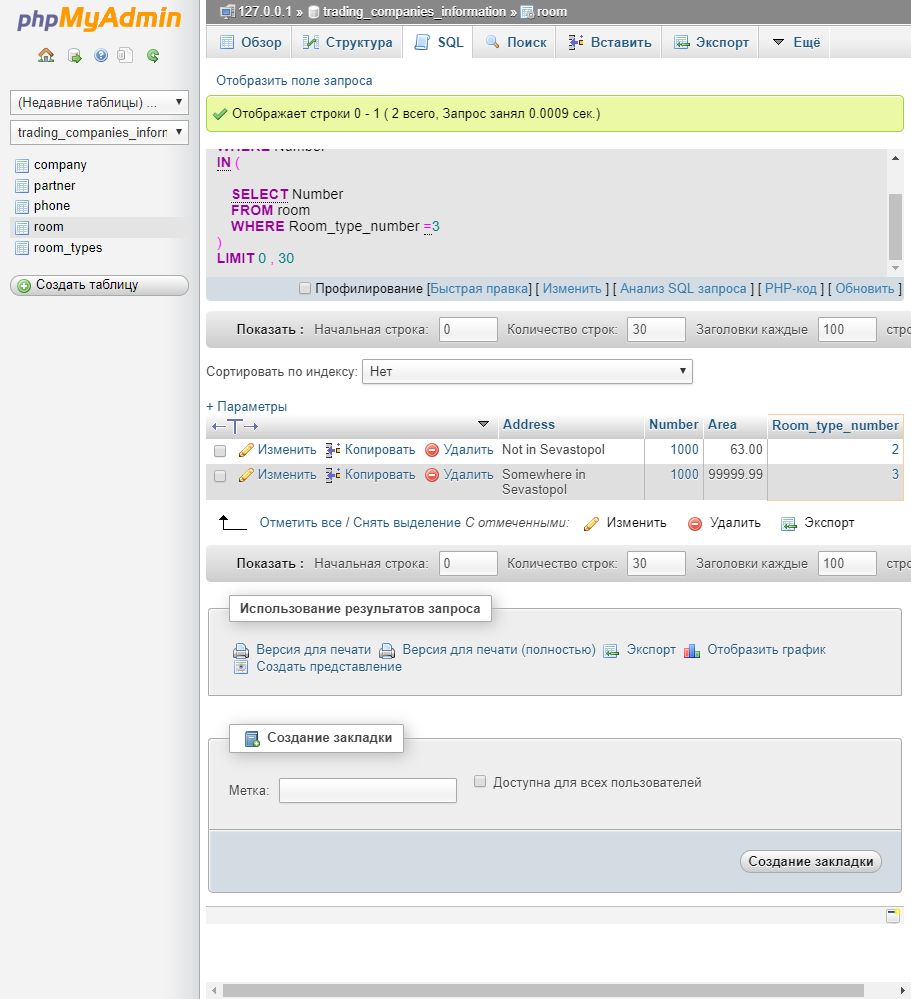


Рисунок 8 – Результат выполнения запроса с подзапросом, возвращающим множественные результаты

* + 1. Подзапросы, использующие вычисления

Вывести все данные о помещениях, площадь которых больше двух площадей минимальной площади помещения в таблице:

SELECT \* FROM room

WHERE Area > (SELECT MIN(Area) \* 2 FROM room)

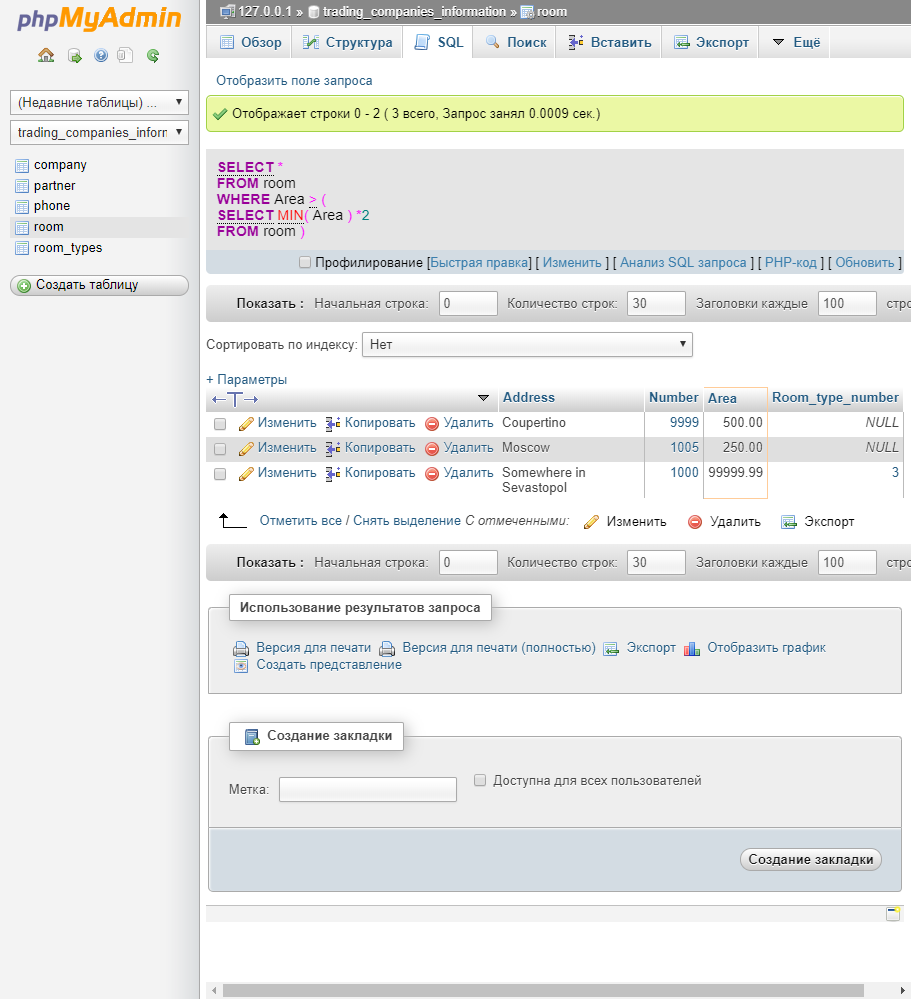


Рисунок 9 – Результат выполнения запроса с подзапросом, использующим вычисления

* + 1. Подзапросы с HAVING

Вывести среднее значение площади для каждого типа комнат, среднее значение площади которых больше минимальной площади всех комнат.

SELECT AVG(Area), Room\_type\_number FROM room

GROUP BY Room\_type\_number

HAVING AVG(Area) > (SELECT MIN(Area) FROM room)

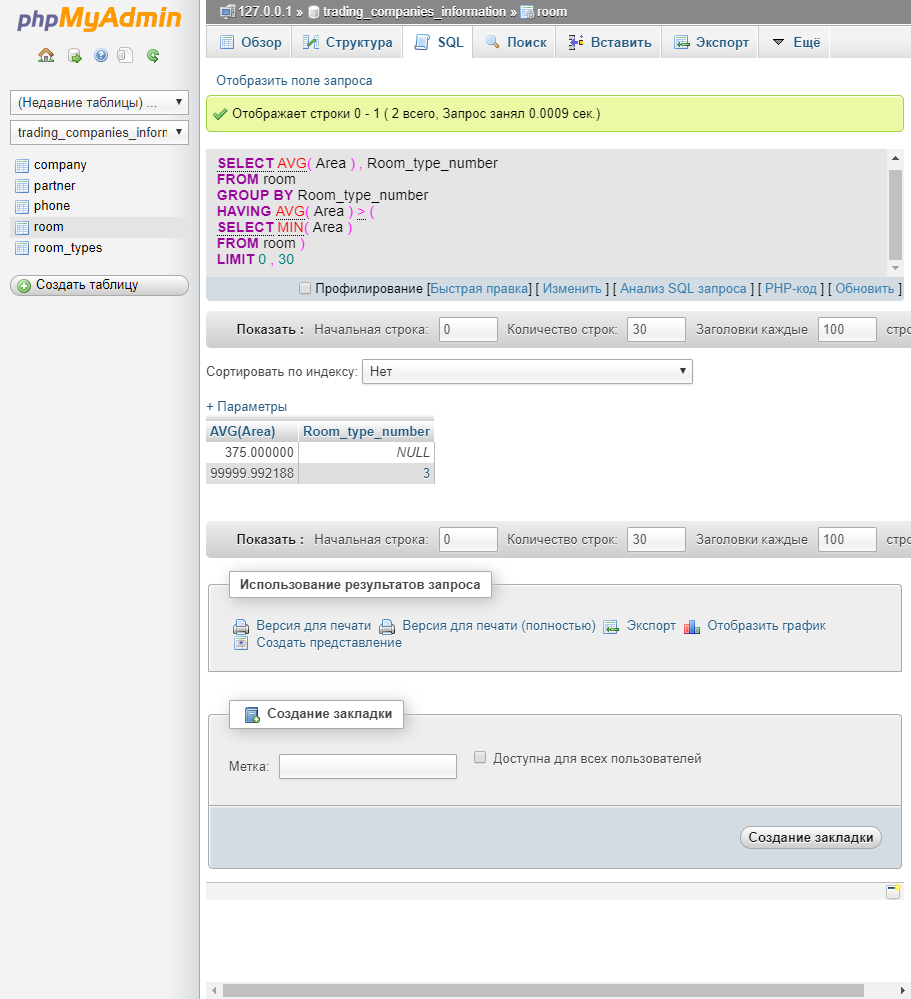


Рисунок 10 – Результат выполнения запроса с HAVING

* + 1. Соотнесение таблицы со своей копией

SELECT \* FROM room one

WHERE Area > (SELECT AVG(Area) FROM room two WHERE one.Number = two.Number)

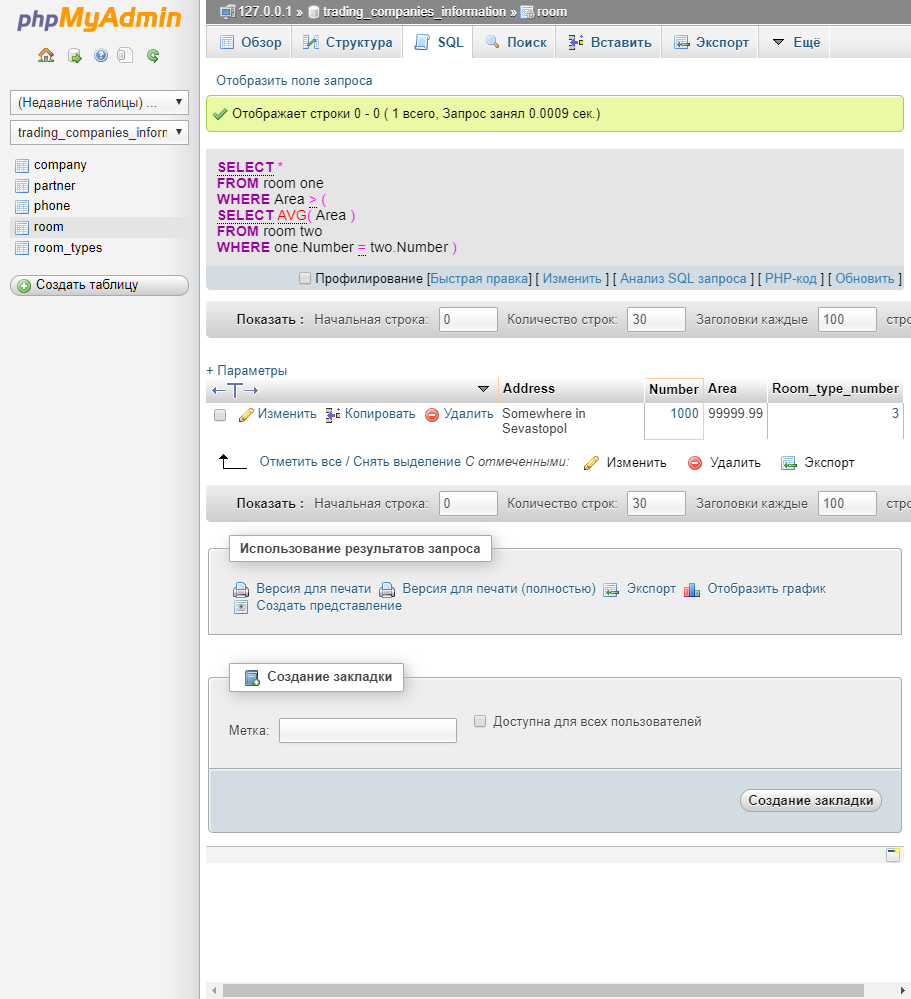


Рисунок 11 – Результат выполнения запроса с коррелированным вложенным подзапросом

* + 1. Коррелированные запросы для двух таблиц

Вывести все данные о компаниях, у которых есть хотя бы одно помещение.

SELECT \* FROM company main

WHERE Number IN (SELECT Number FROM room secondary WHERE main.Number = secondary.Number)

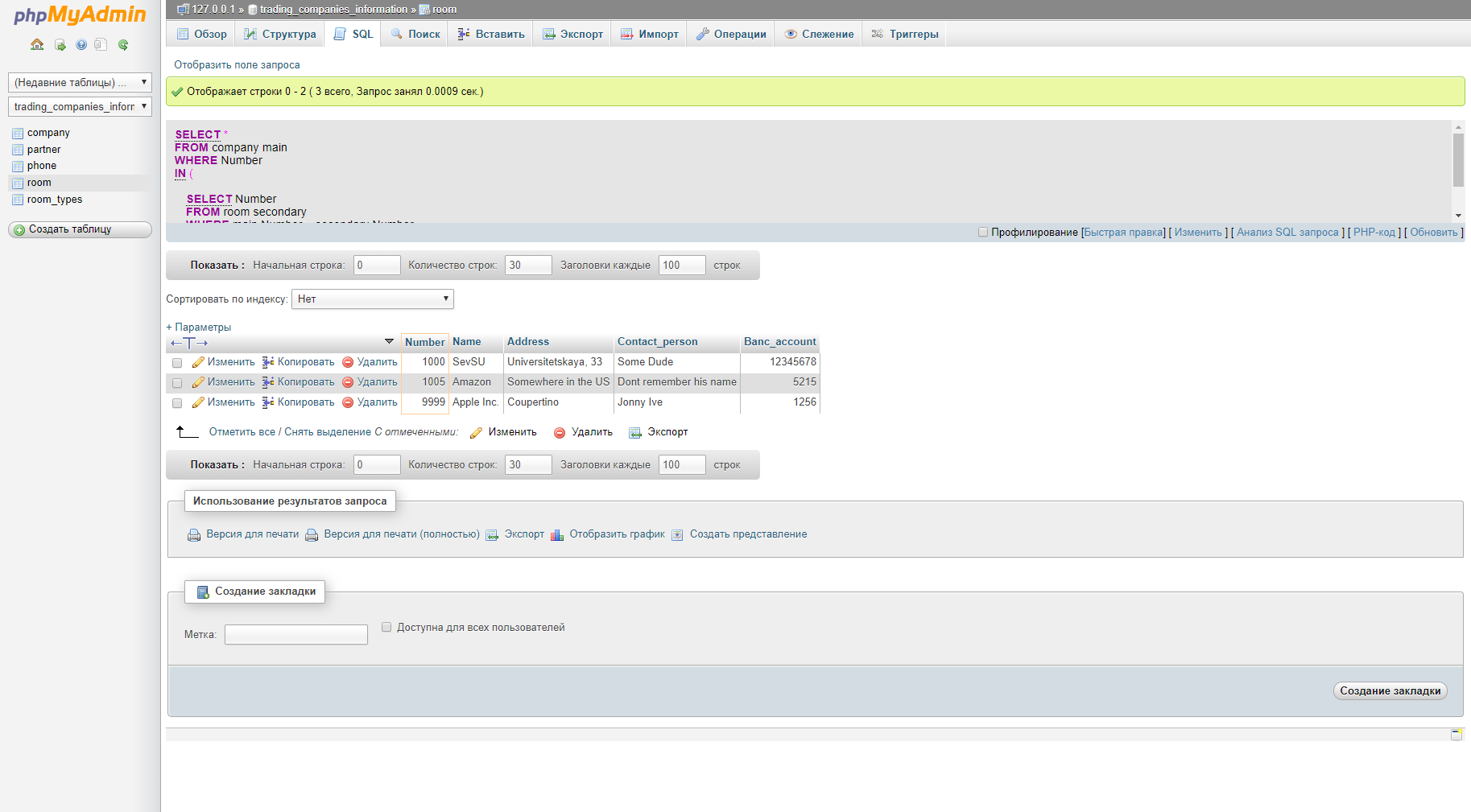


Рисунок 12 – Результат выполнения запроса с коррелированным вложенным подзапросом

* + 1. Запрос с параметром EXISTS

Выполним тот же запрос используя параметр EXISTS:

SELECT \* FROM company main

WHERE EXISTS (SELECT \* FROM room secondary WHERE main.Number = secondary.Number)

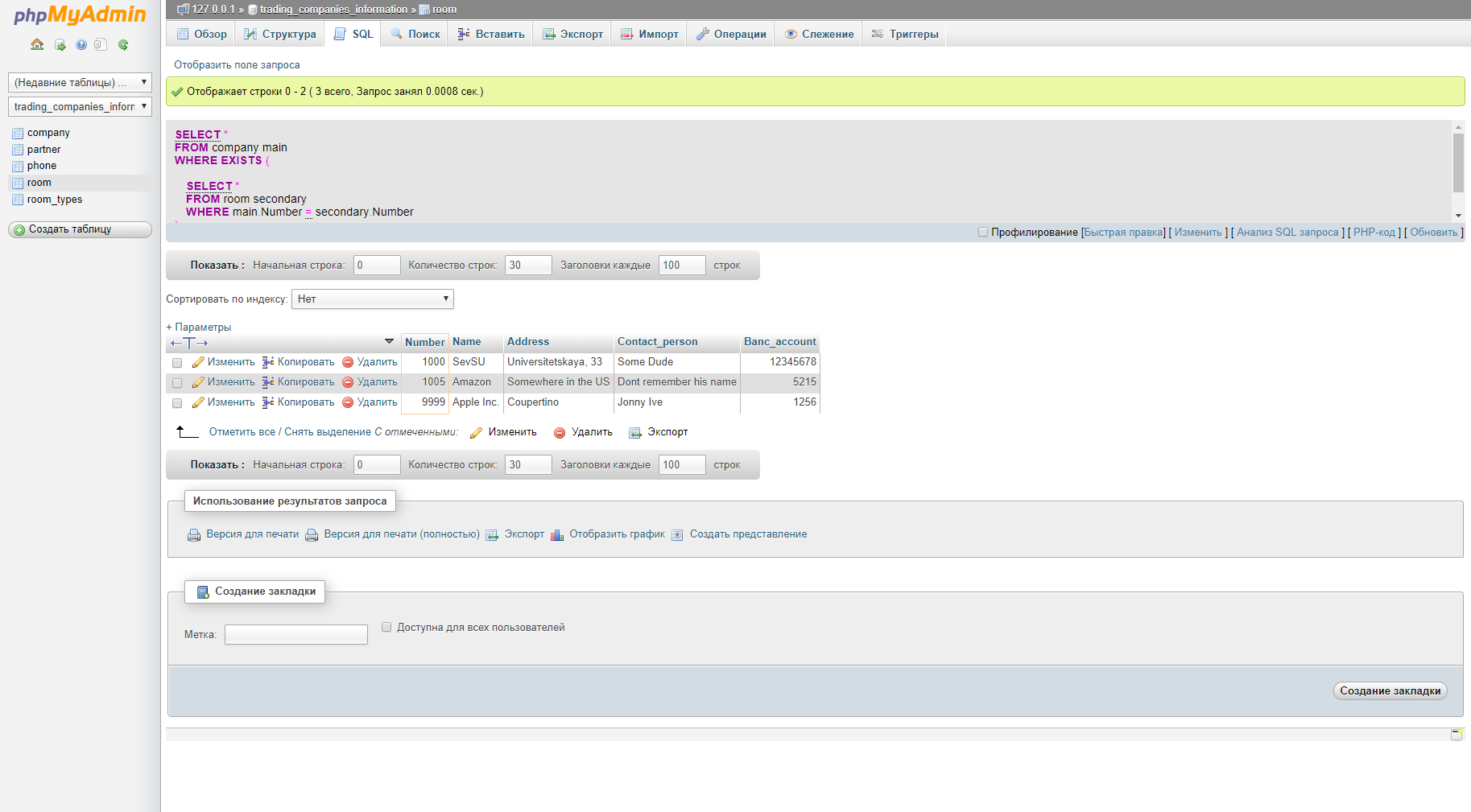


Рисунок 13 – Результат выполнения запроса с параметром EXISTS

* + 1. Выполнение индивидуального задания

Вывести все телефоны фирмы «Оптима Крым».

SELECT Phone\_number FROM company, phone, room

WHERE Name = 'Оптима Крым' AND company.Number = room.Number AND company.Number = phone.Number AND room.Address = phone.Address

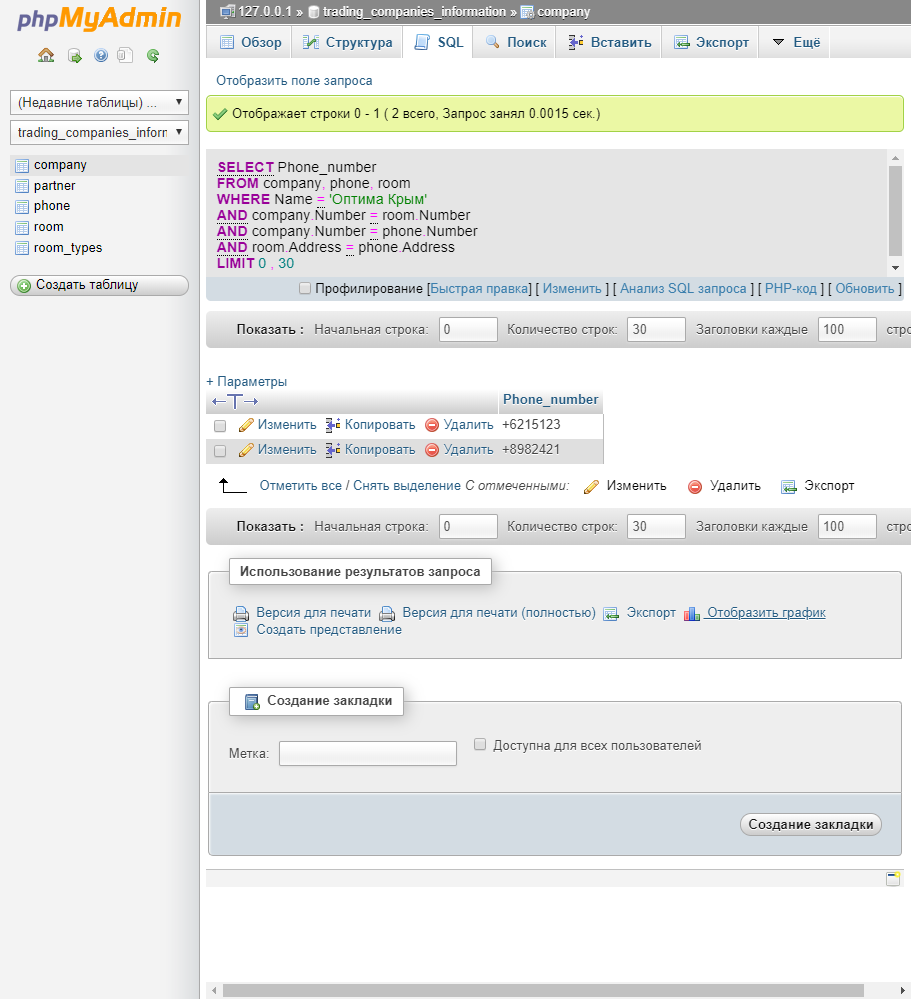


Рисунок 14 – Вывод всех телефонов фирмы «Оптима Крым»

Вывести названия фирмы, которая не имеет контрагентов.

SELECT Name FROM company

WHERE Name NOT IN (SELECT Buyer\_company\_number FROM partner)

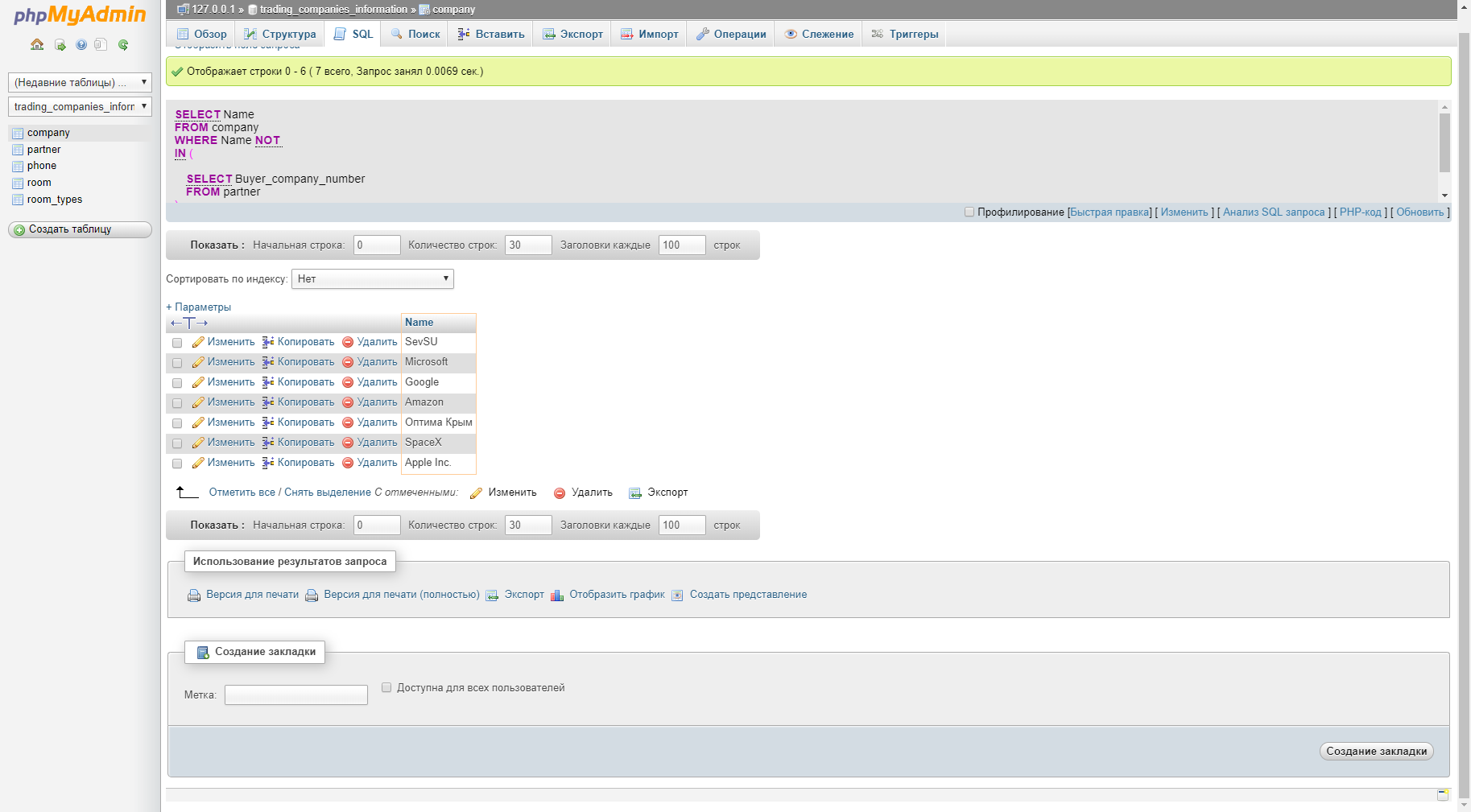


Рисунок 15 – Вывод названий фирм, не имеющих контрагентов

Вывести название фирмы, занимающей помещение с максимальной площадью.

SELECT Name FROM company JOIN room ON company.Number = room.Number

WHERE Area = (SELECT MAX(Area) FROM room)

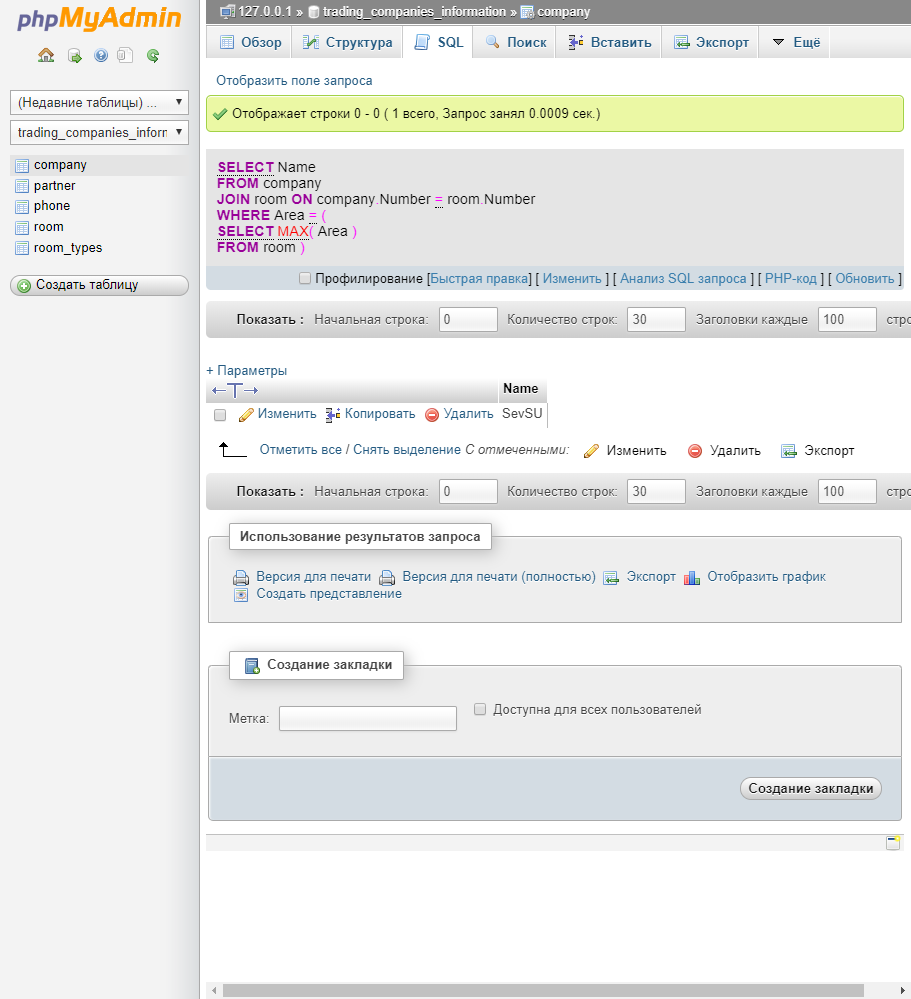


Рисунок 16 – Вывод компании, занимающей помещение с максимальной площадью

* 1. ВЫВОД

В ходе лабораторной работы были изучены команды объединения данных из нескольких таблиц, а также вложенные запросы. На практике оператор JOIN работает быстрее, чем объединение по условию, за счет использования индексов, но в данной работе это не так, так как индексы не настроены.