**Домашнее задание**

Вводные: в системе, которая работает на базе данных PostgreSQL, есть 3 таблицы:

1. Таблица разработчиков - **developers**

|  |  |
| --- | --- |
| **Id (Primary Key)** | **name** |
| 1 | Ромашка |
| 2 | Лютик |
| 3 | Одуванчик |
| … | … |

1. Таблица проектов - **projects**

|  |  |
| --- | --- |
| **Id (Primary Key)** | **name** |
| 1 | Офигеть какой проект |
| 2 | Секретный проект |
| 3 | Проект года |
| … | … |

1. Таблица истории взятия проектов в работу разработчиками – **project\_history**, где **project\_id** ключ для таблицы проектов, а **developer\_id** ключ для таблицы разработчиков

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id (Primary Key)** | **project\_id** | **developer\_id** | **start\_date** | **end\_date** |
| 1 | 2 | 3 | 01.02.2019 | 12.12.2020 |
| 2 | 3 | 2 | 13.12.2019 | 27.02.2020 |
| 3 | 1 | 1 | 02.01.2020 | 03.02.2020 |
| 4 | 3 | 1 | 27.02.2020 | 31.12.2999 |
| 5 | 1 | 2 | 03.02.2020 | 31.12.2999 |
| … | … | … | … | … |

Задания:

1. Необходимо вывести количество проектов, которое находится в работе у каждого разработчика на 07.04.2020
2. После сбоя в базе данных в таблице разработчиков появились полные дубли по ключу (id) и имени разработчика (name). Необходимо вывести задублированные строки. Как можно удалить дубли?
3. Вывести все проекты, которые никогда не брались в работу разработчиками
4. Необходимо вывести всех разработчиков и напротив каждого из них посчитать количество его полных тёзок, без использования конструкции group by.
5. Задание со \*

Для определения самых крупных проектов необходимо вывести список проектов, в которых участвовали **все** разработчики.

**Решение**

Внесём данные в таблицы, добавим несколько дополнительных значений, для массовости.

Таблица разработчиков

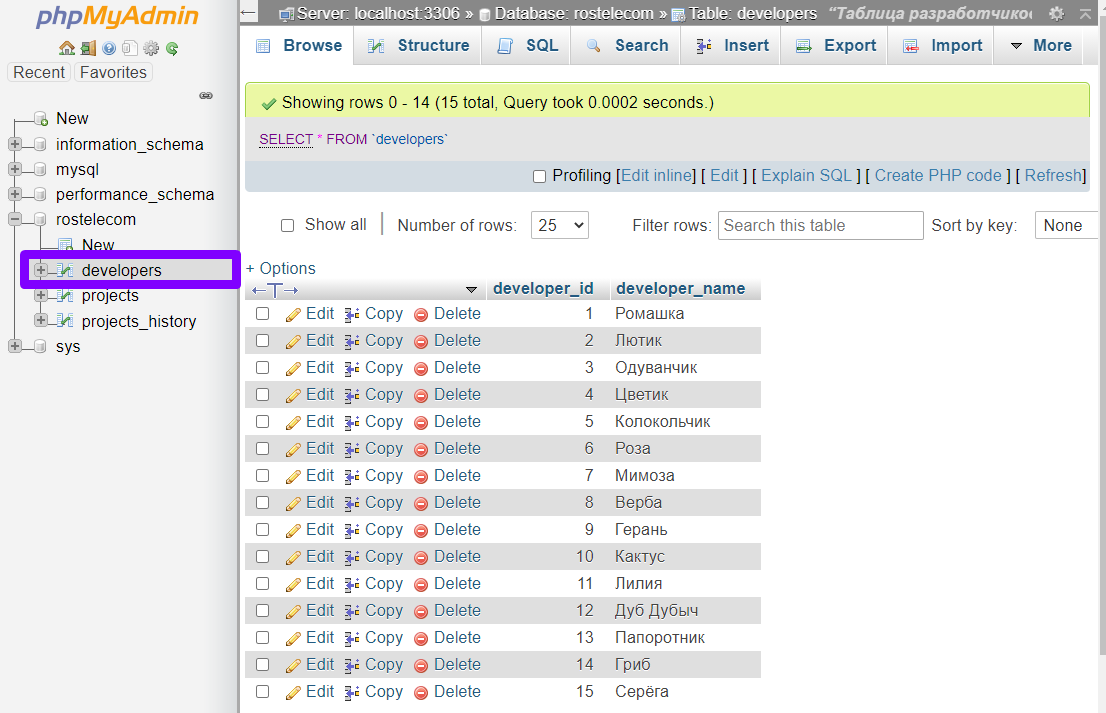
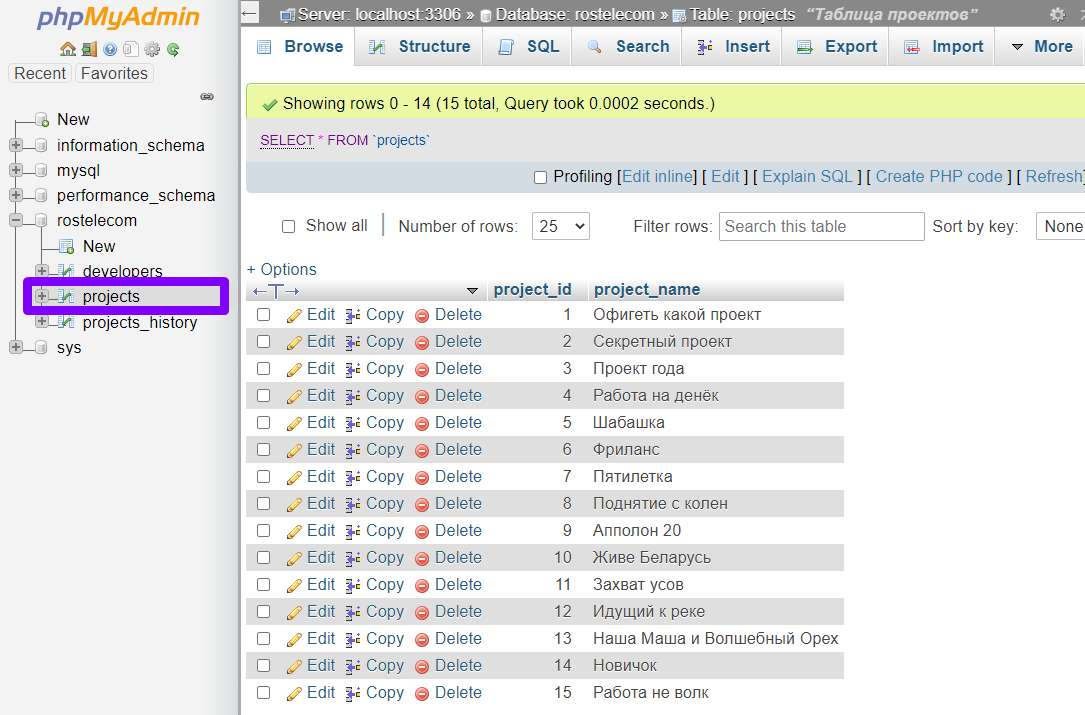
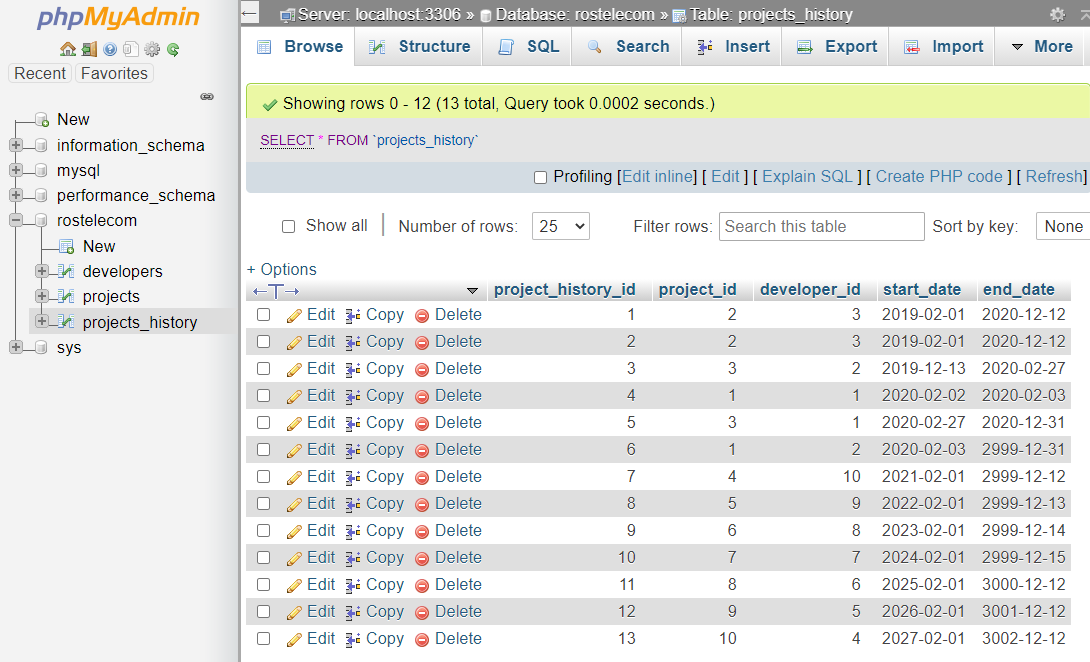
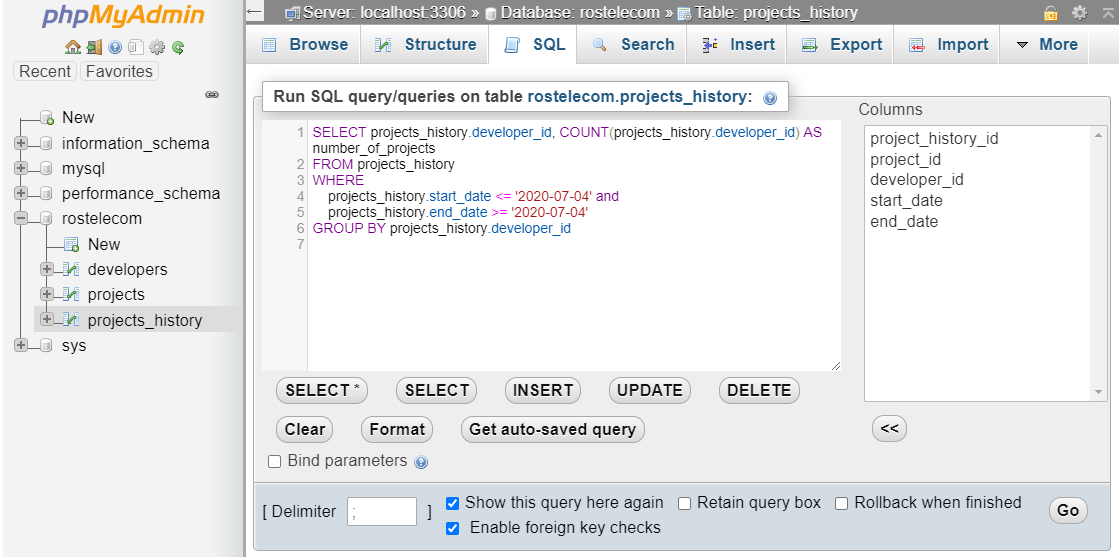


Таблица проектов  


История проектов  


**Задание 1**

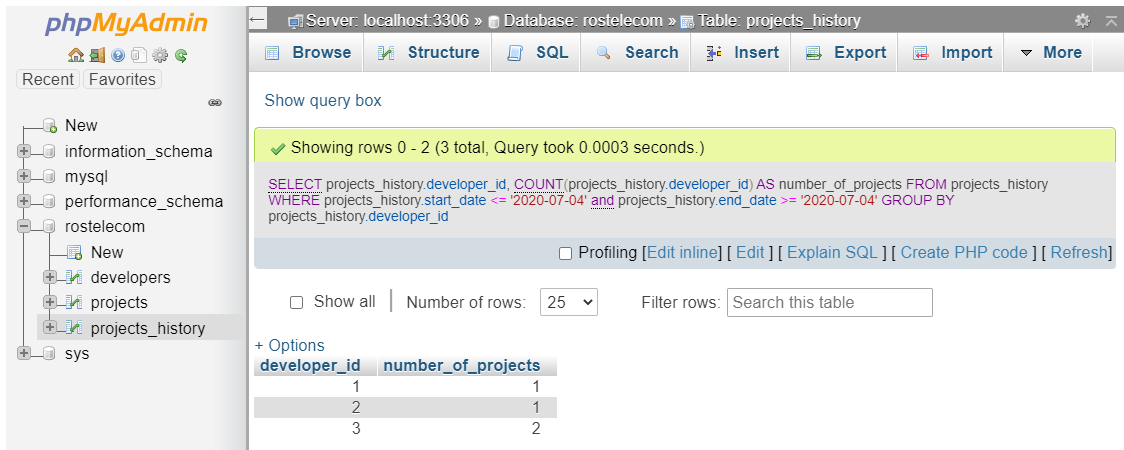
Произведём запрос. COUNT посчитаем количество упомянутых разработчиков, что будет равняться количеству проектов у них. Добавим фильтрацию по дате. Начало проекта должно быть раньше опорной даты (start <= date), конец проекта должен быть позже даты (end >= date).



Текст запроса:

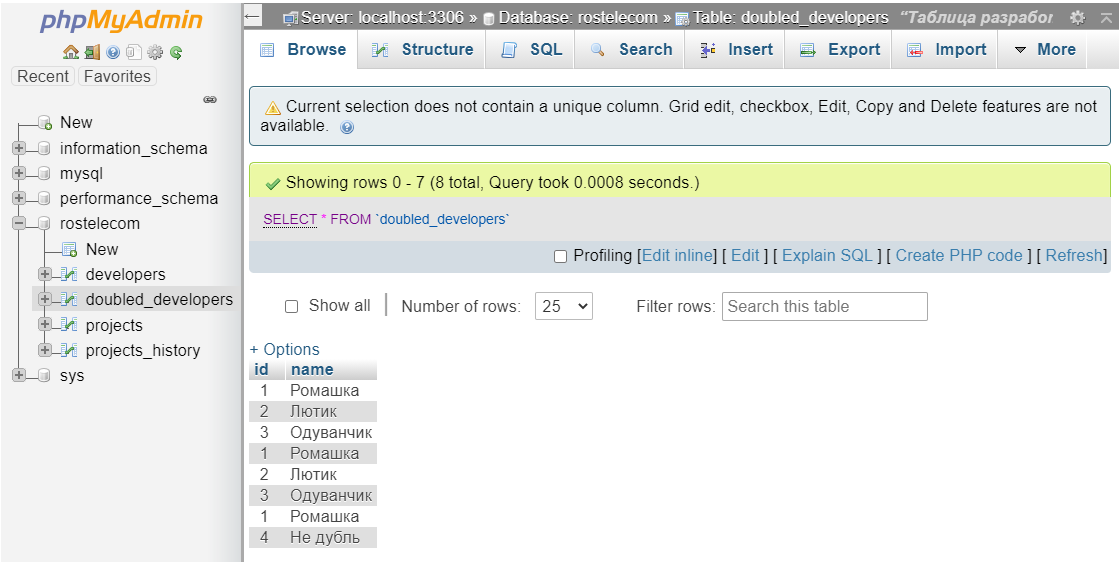
SELECT projects\_history.developer\_id, COUNT(projects\_history.developer\_id) AS number\_of\_projects  
FROM projects\_history  
WHERE   
 projects\_history.start\_date <= '2020-07-04' and   
 projects\_history.end\_date >= '2020-07-04'  
GROUP BY projects\_history.developer\_id

Полученный ответ:

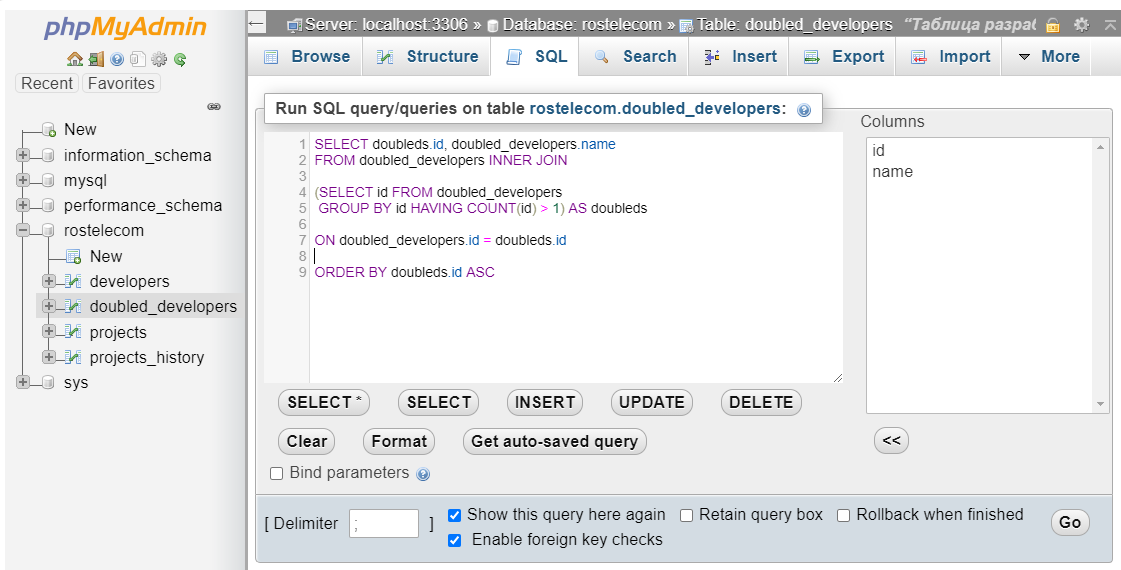


**Задание 2**

При создании таблицы я указал столбец id как ключевой, поскольку так было указано в задании. Создадим неправильную таблицу, где id будет обычной строкой:



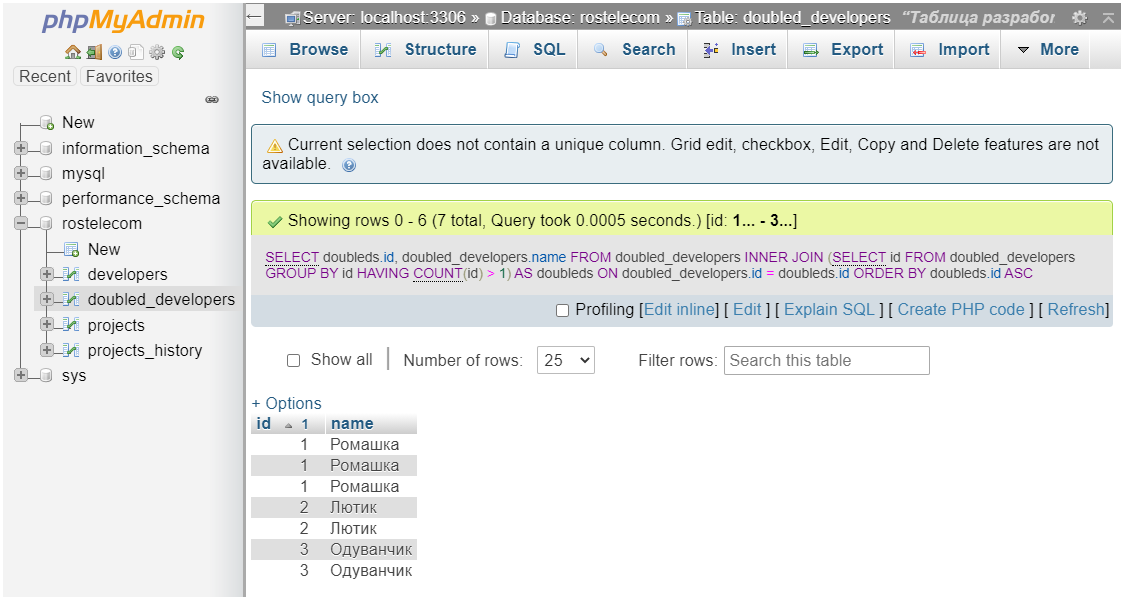
Выведем задублированные строки:



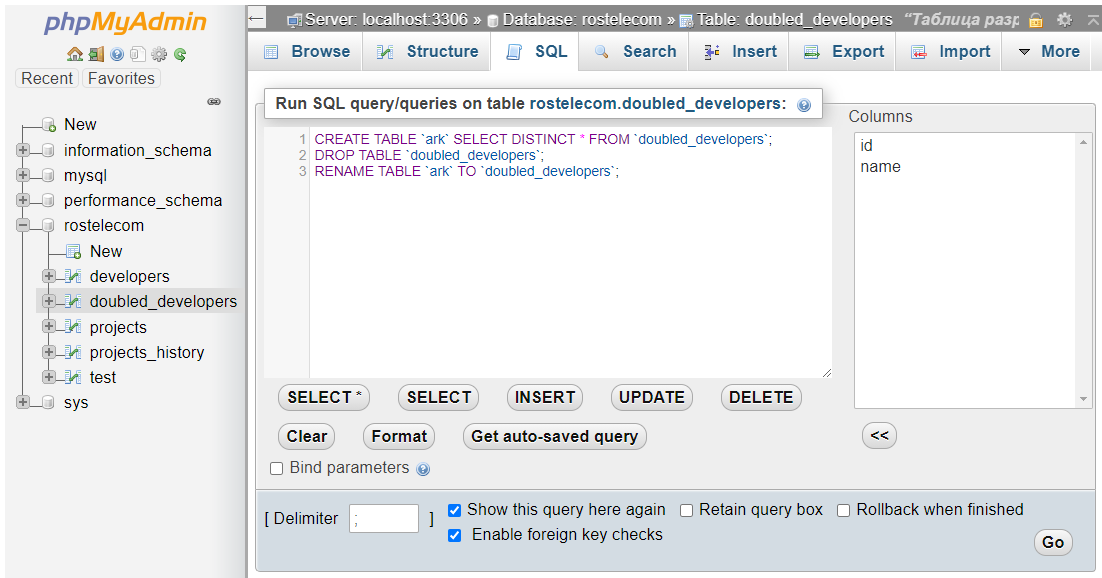
Текст запроса:

SELECT doubleds.id, doubled\_developers.name   
FROM doubled\_developers INNER JOIN  
  
(SELECT id FROM doubled\_developers  
 GROUP BY id HAVING COUNT(id) > 1) AS doubleds  
ON doubled\_developers.id = doubleds.id  
  
ORDER BY doubleds.id ASC

Результат запроса:



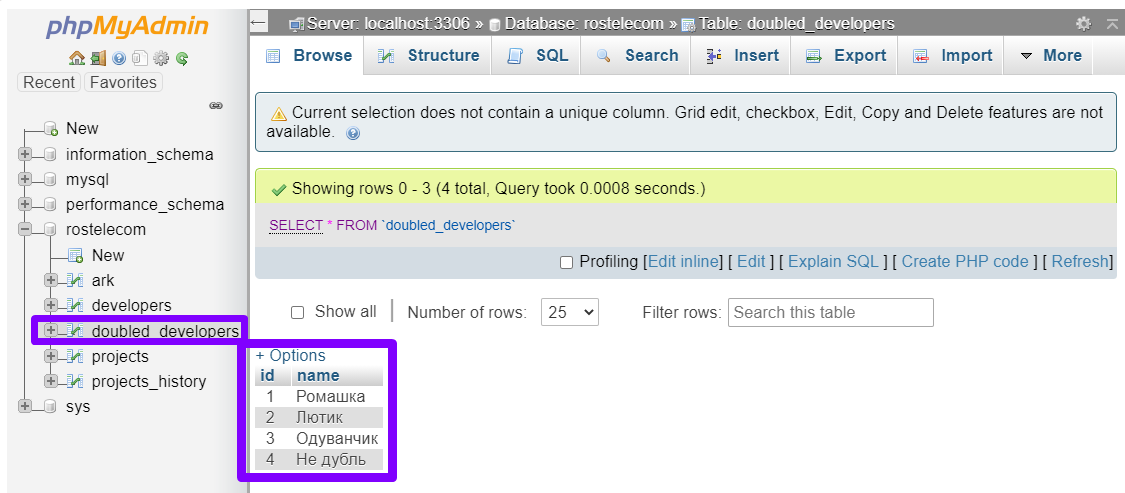
Удалить дубли можно просто перезаписав таблицу тремя запросами:



Текст запроса:

CREATE TABLE `ark` SELECT DISTINCT \* FROM `doubled\_developers`;  
DROP TABLE `doubled\_developers`;  
RENAME TABLE `ark` TO `doubled\_developers`;

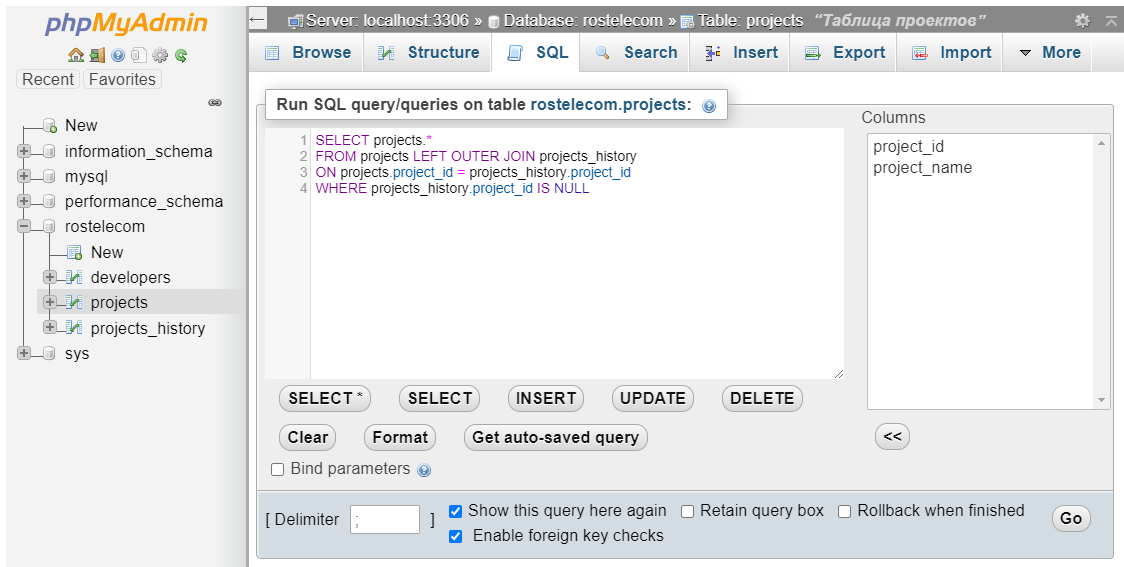
Результат манипуляций:



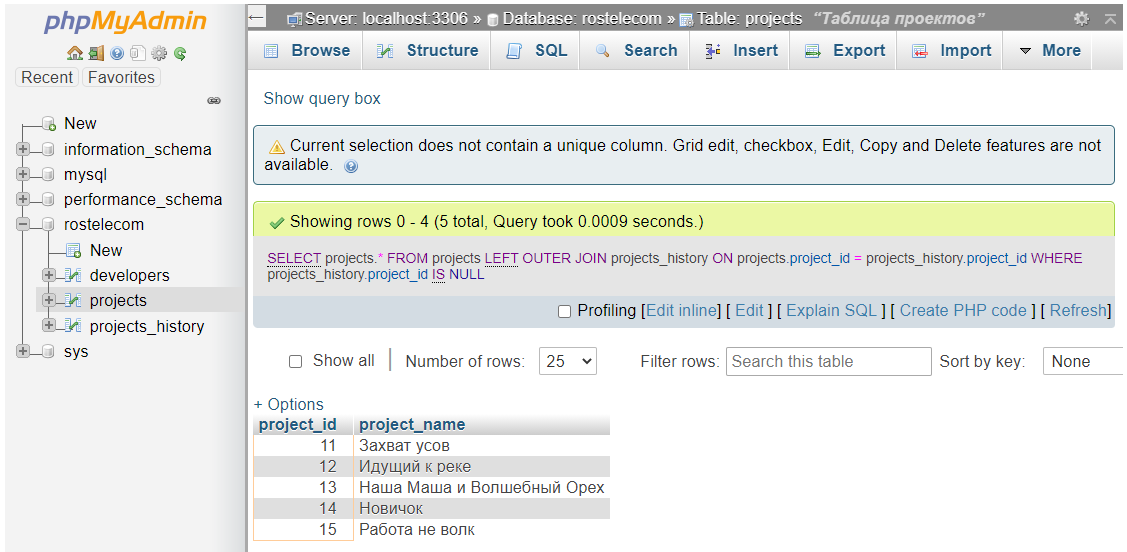
**Задание 3**

Вывести все проекты, которые никогда не брались в работу разработчиками можно следующим запросом:

SELECT projects.\*   
FROM projects LEFT OUTER JOIN projects\_history  
ON projects.project\_id = projects\_history.project\_id  
WHERE projects\_history.project\_id IS NULL

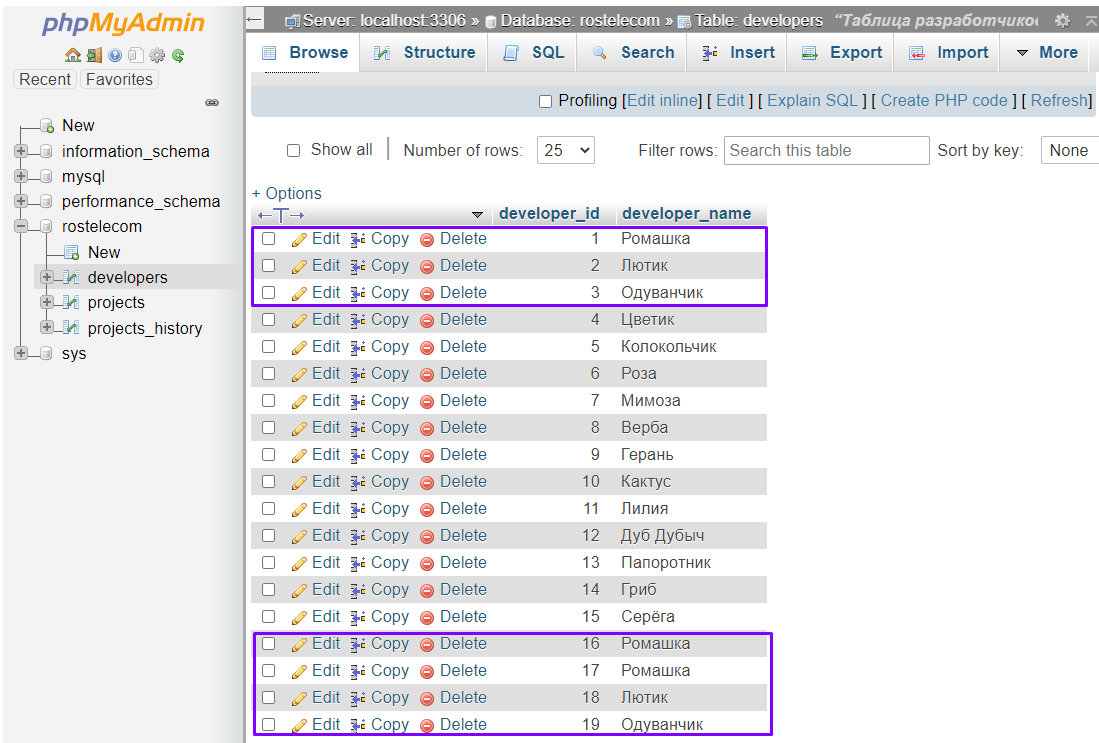


Результат:



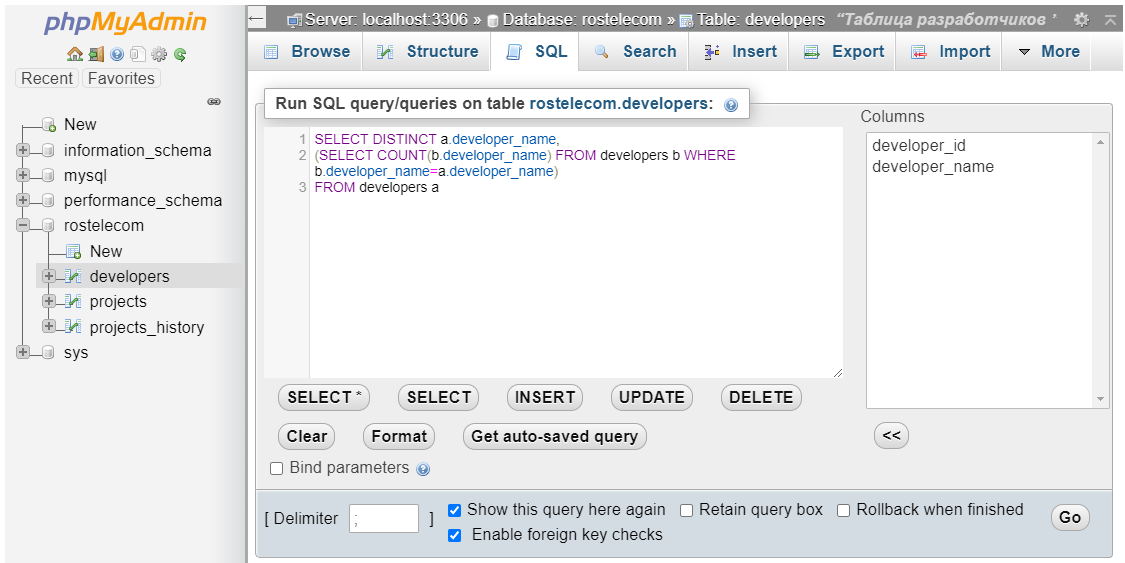
**Задание 4**

Добавим тёзок в таблицу:



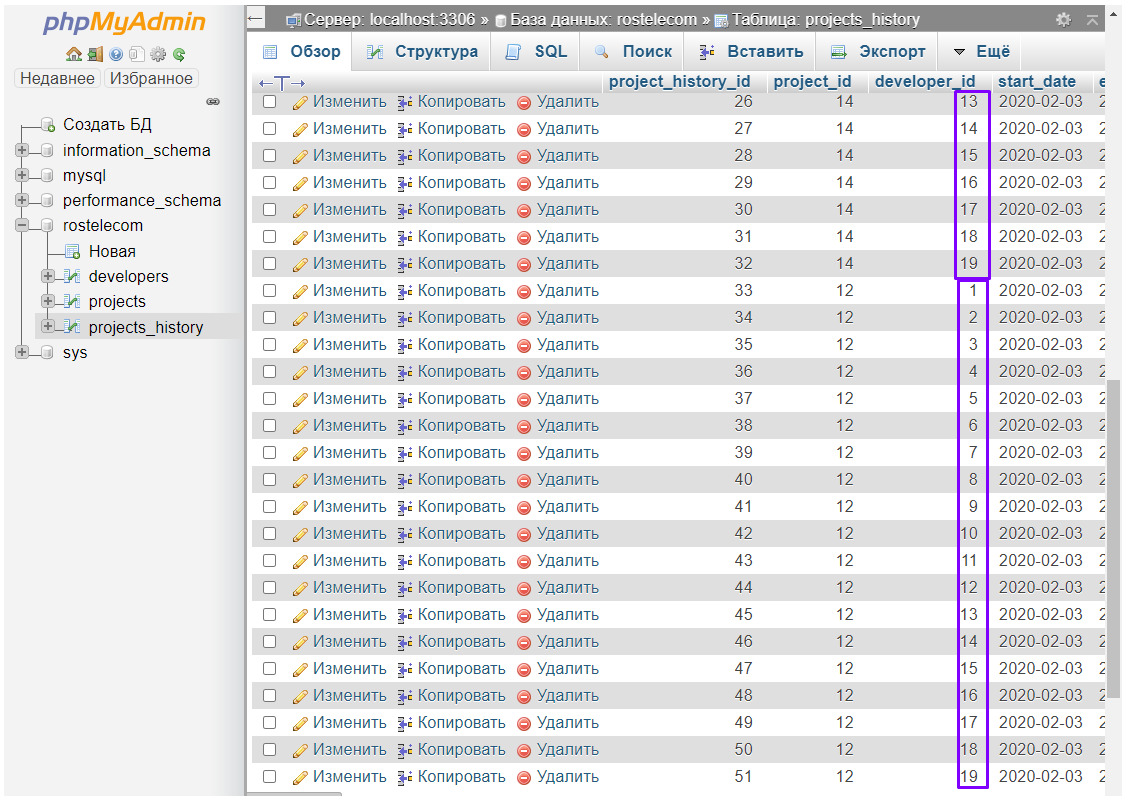
Выведем всех разработчиков и количество его полных тёзок, без использования конструкции group by: (но почему!?)

SELECT DISTINCT a.developer\_name,  
(SELECT COUNT(b.developer\_name) FROM developers b WHERE b.developer\_name=a.developer\_name)  
FROM developers a



**Задание 5**

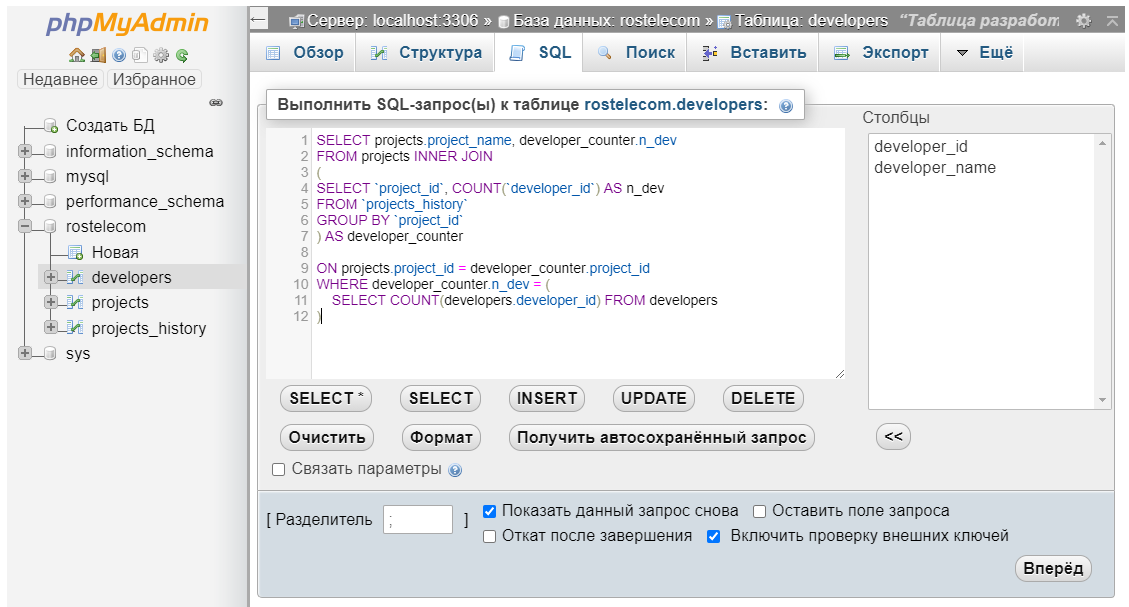
Определим самые крупные проекты (вывести список проектов), в которых участвовали **все** разработчики. Немного изменим таблицу projects\_history. Проекты 14 «Новичок» и 12 «Идущий к реке» должны удовлетворять условиям.



Выполним запрос:

SELECT projects.project\_name, developer\_counter.n\_dev  
FROM projects INNER JOIN (  
SELECT `project\_id`, COUNT(`developer\_id`) AS n\_dev  
FROM `projects\_history`  
GROUP BY `project\_id`  
) AS developer\_counter  
  
ON projects.project\_id = developer\_counter.project\_id  
WHERE developer\_counter.n\_dev = (  
 SELECT COUNT(developers.developer\_id) FROM developers

)



В результате получим:

