МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«Севастопольский государственный университет»

кафедра Информационные системы

Сирота Марина Романовна

Институт информационных технологий и управления в технических системах

курс 2 группа ИС/б-22-о

Управление в технических системах (уровень бакалавриата)

ОТЧЕТ

по лабораторной работе №3

по дисциплине: «Теория баз данных»

по теме: «элементарные выборки посредством оператора SELECT. Агрегатные фунукции SQL»

Отметка о зачете \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_

(дата)

Руководитель практикума:

доц. Волкова Т.В.

(должность) (подпись) (инициалы, фамилия)

Севастополь

2018

1 ЦЕЛЬ РАБОТЫ

Целью лабораторной работы является изучение работы оператораSELECT языка SQL, а также изучение возможностей обработки данных с помощью агрегатных функций языка SQL.

2 ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

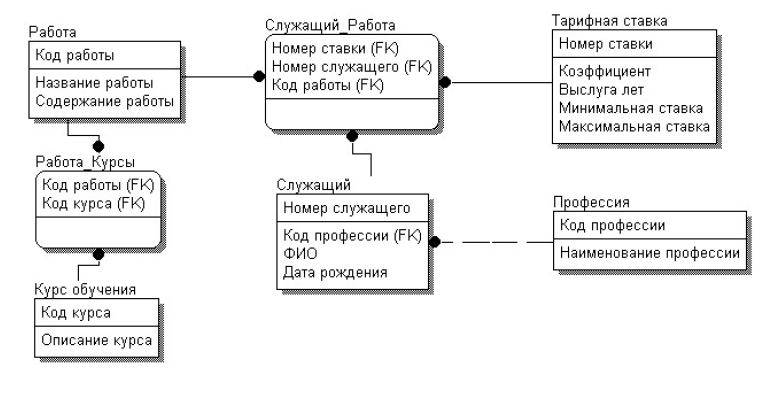


Рисунок 2.1 ─ Вариант задания

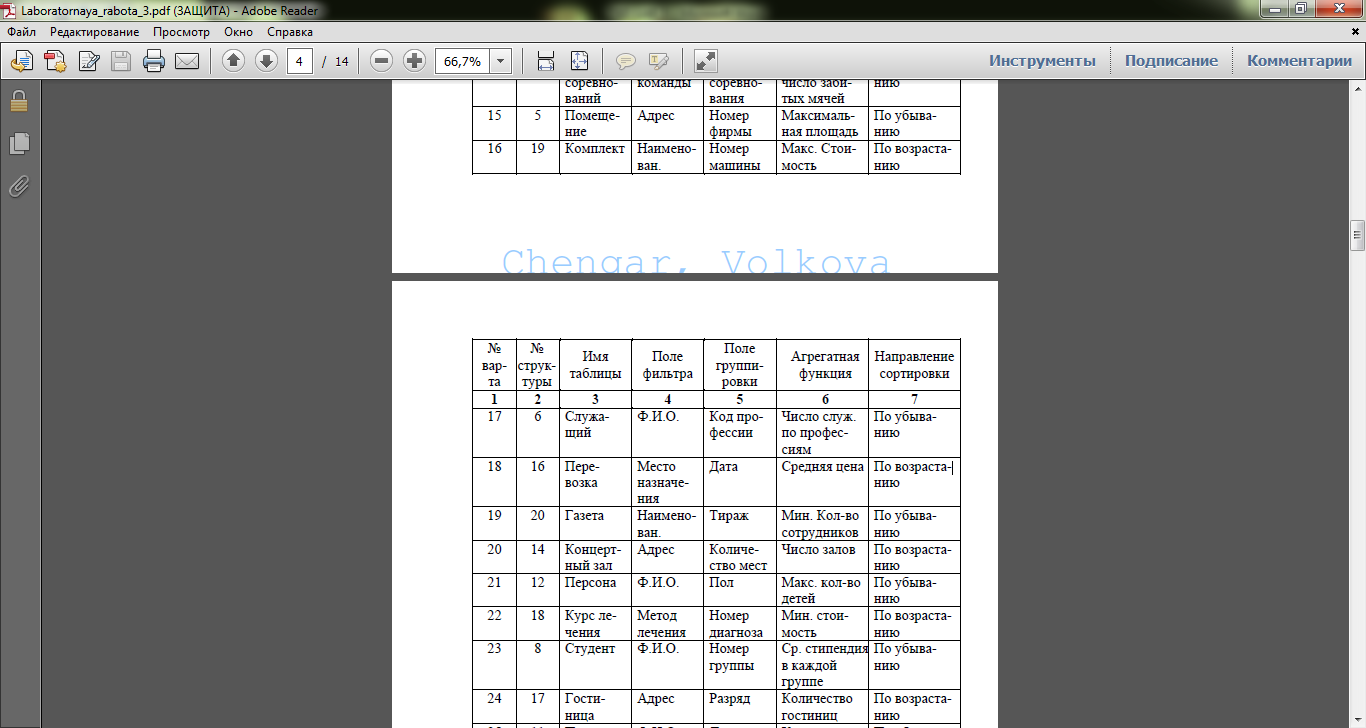


Рисунок 2.2 ─ Вариант задания

3 ВЫПОЛНЕНИЕ РАБОТЫ

1. Запрос, отображающий все строки таблицы.

SELECT \* FROM `служащий`

*(Выбрать все из «Служащий»)*

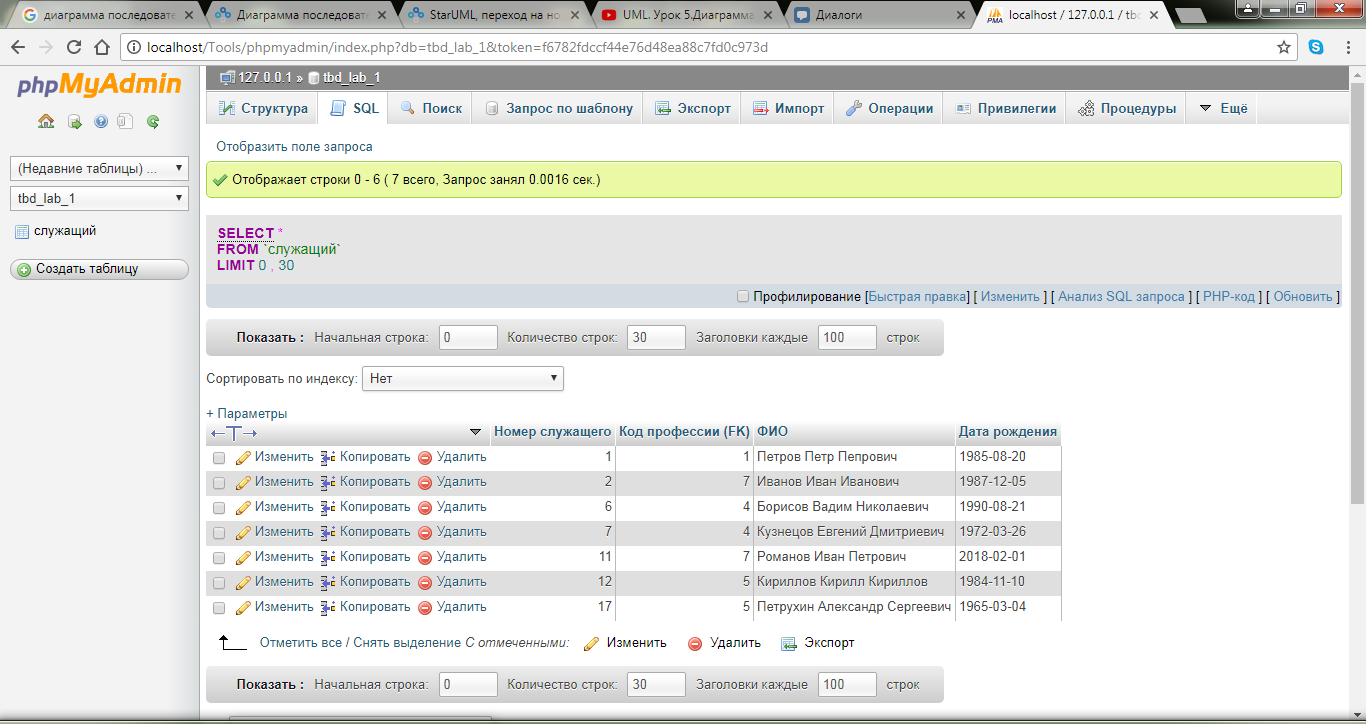


Рисунок 3.1 – Отображение всех записей

1. Запрос, задающий порядок столбцов, отличный от исходного.

SELECT \* FROM `служащий` GROUP BY `служащий`.`Дата рождения` DESC

*(Выбрать все из таблицы Служащий, отсортировав по убыванию даты рождения)*

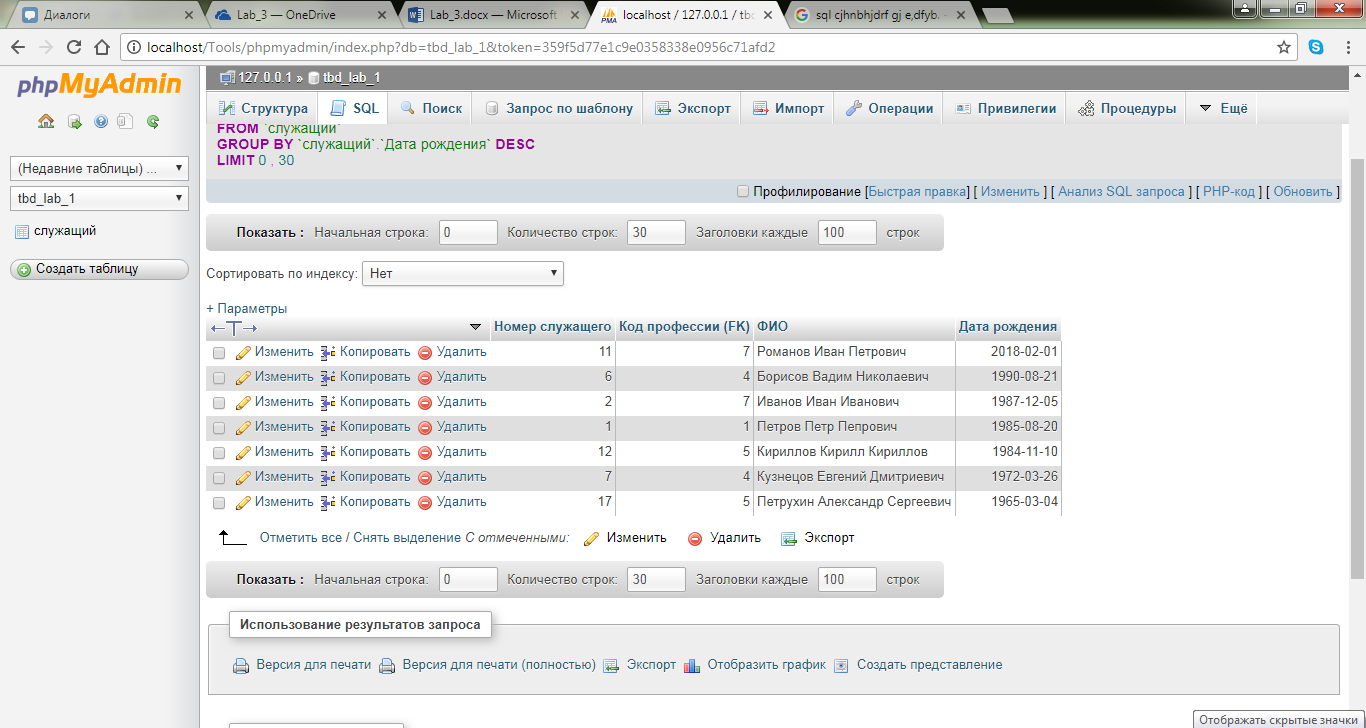


Рисунок 3.2 – Сортировка по дате рождения

1. Дейcтвие модификатора distinct.

SELECT COUNT(DISTINCT `Tip sobitiya`) FROM TRUD;

*(Выбрать к подсчету только уникальные поля «Код профессии (FK)» из таблицы Служащий)*

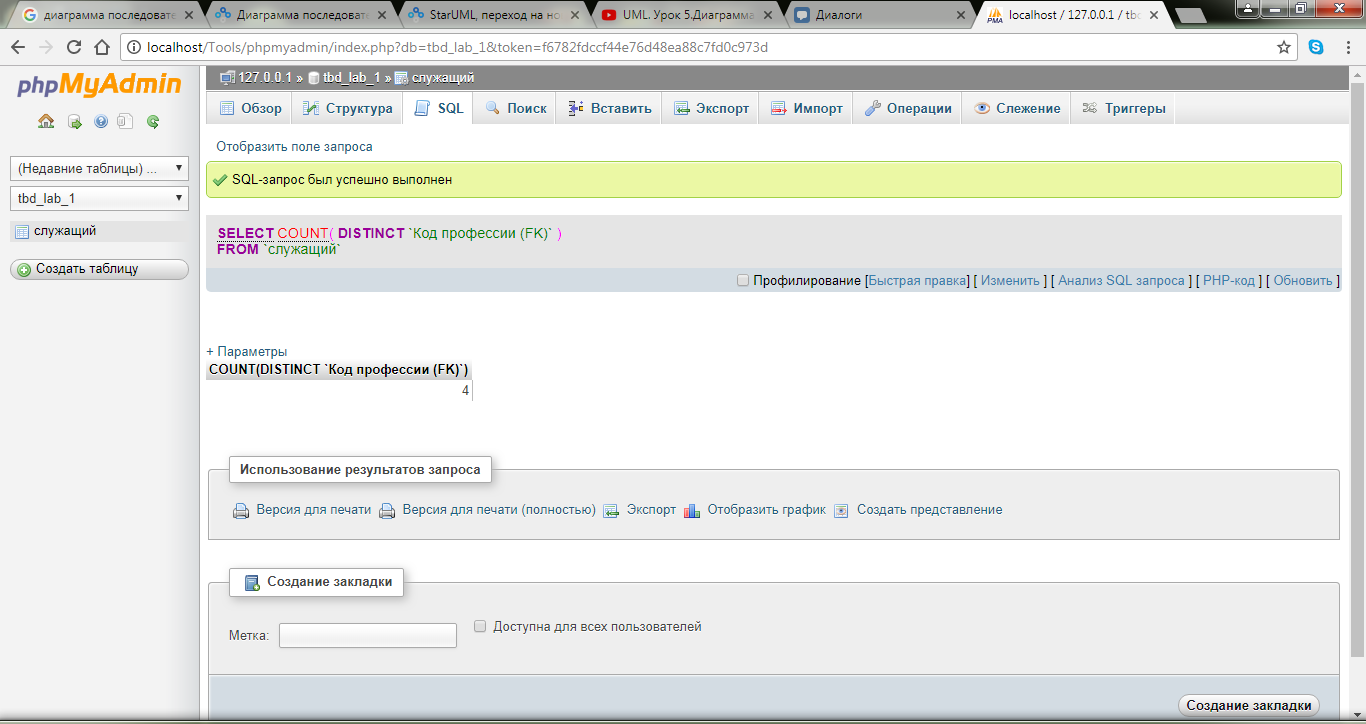


Рисунок 3.3 – Модификатор distinct

1. Ограничение вывода данных с помощью SQL-запроса, используяWHERE с простым условием

SELECT \* FROM `служащий` WHERE `Код профессии (FK)`='5';

*(Выбрать из таблицы Служащий только записи, где «Код профессии» - «5» )*

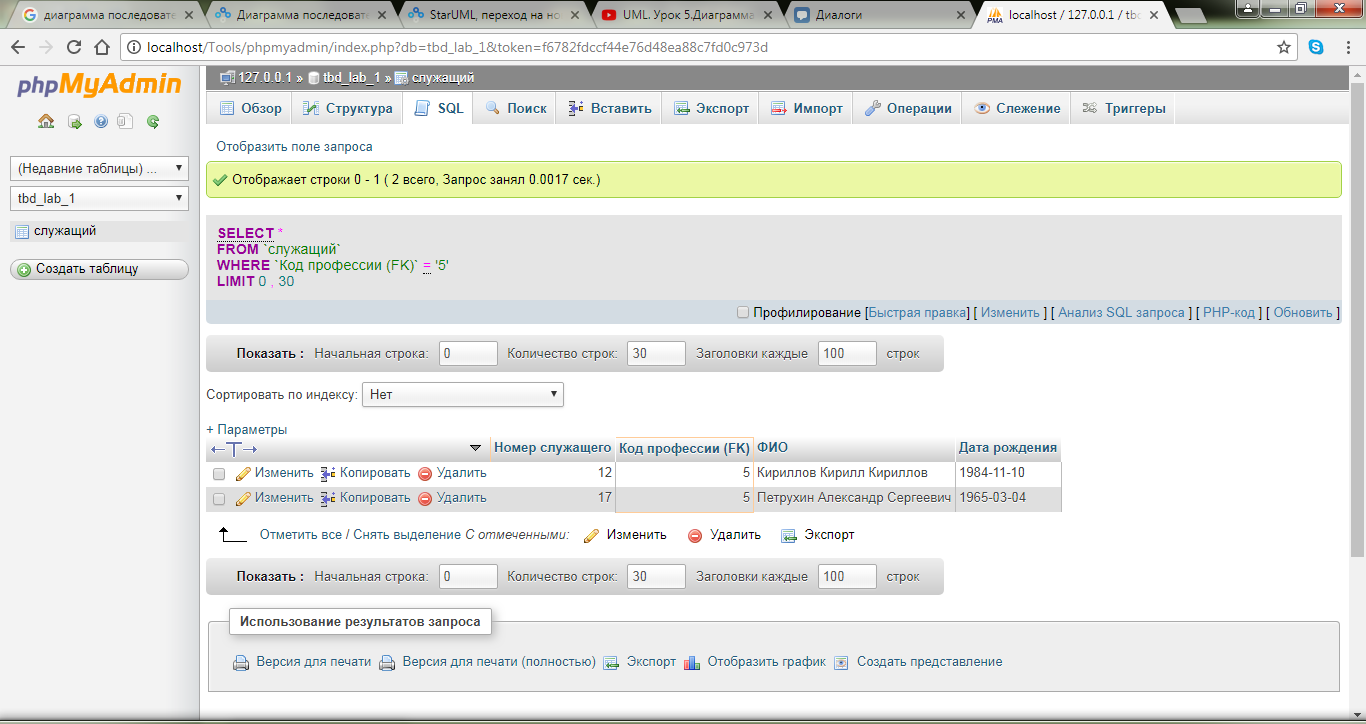


Рисунок 3.4 – WHERE с простым условием

1. Ограничение вывода данных с помощью SQL-запроса, используяWHERE и составное условие

SELECT \* FROM `*cлужащий* ` WHERE `Код професси (FK)`> 14

*(Выбрать из таблицы Служащий только записи, где «Код професси (FK)» больше 4 )*

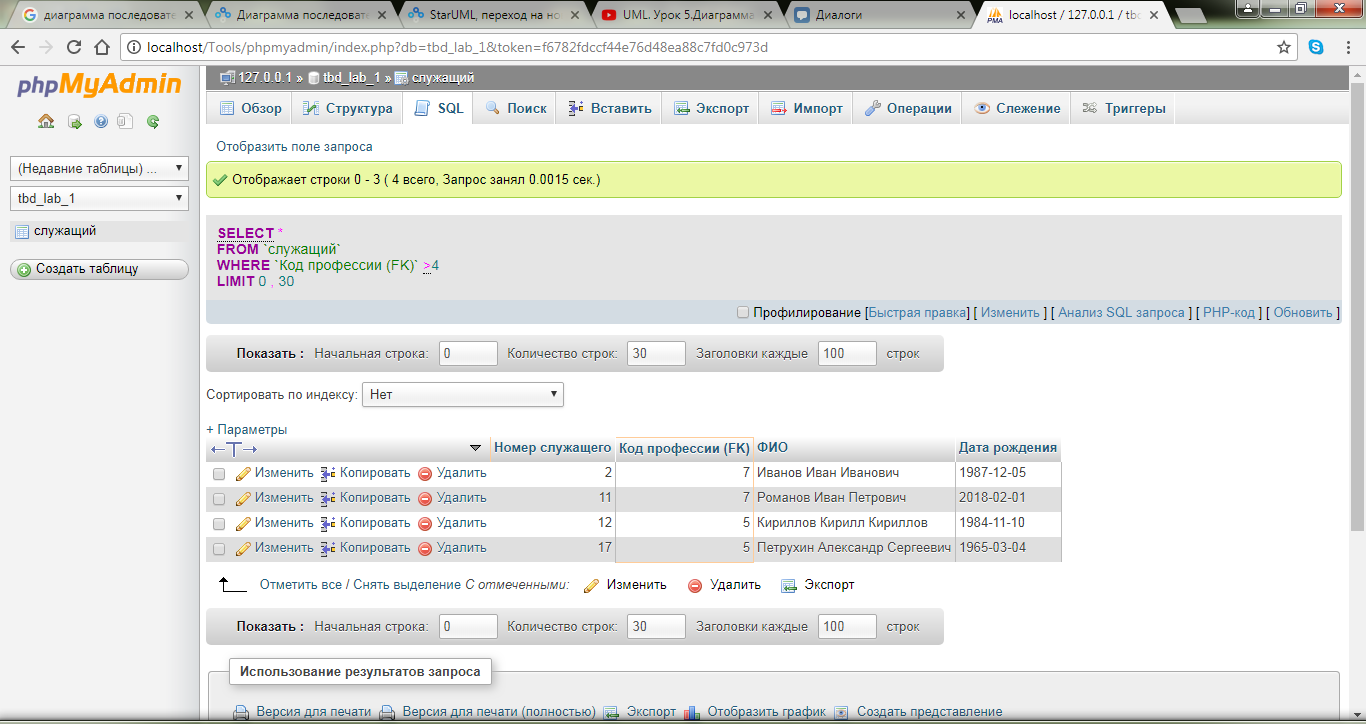


Рисунок 3.5 – WHERE и составное условие

1. Демонстрация SQL-запроса действия специальныхфункций IN, BETWEEN, LIKE и IS NULL в условии.

SELECT \* FROM `служащий` WHERE `Код профессии (FK)` BETWEEN 4 AND 7

*(Выбрать из таблицы Служащий только записи, где «Код профессии (FK)» от 4 до 7)*

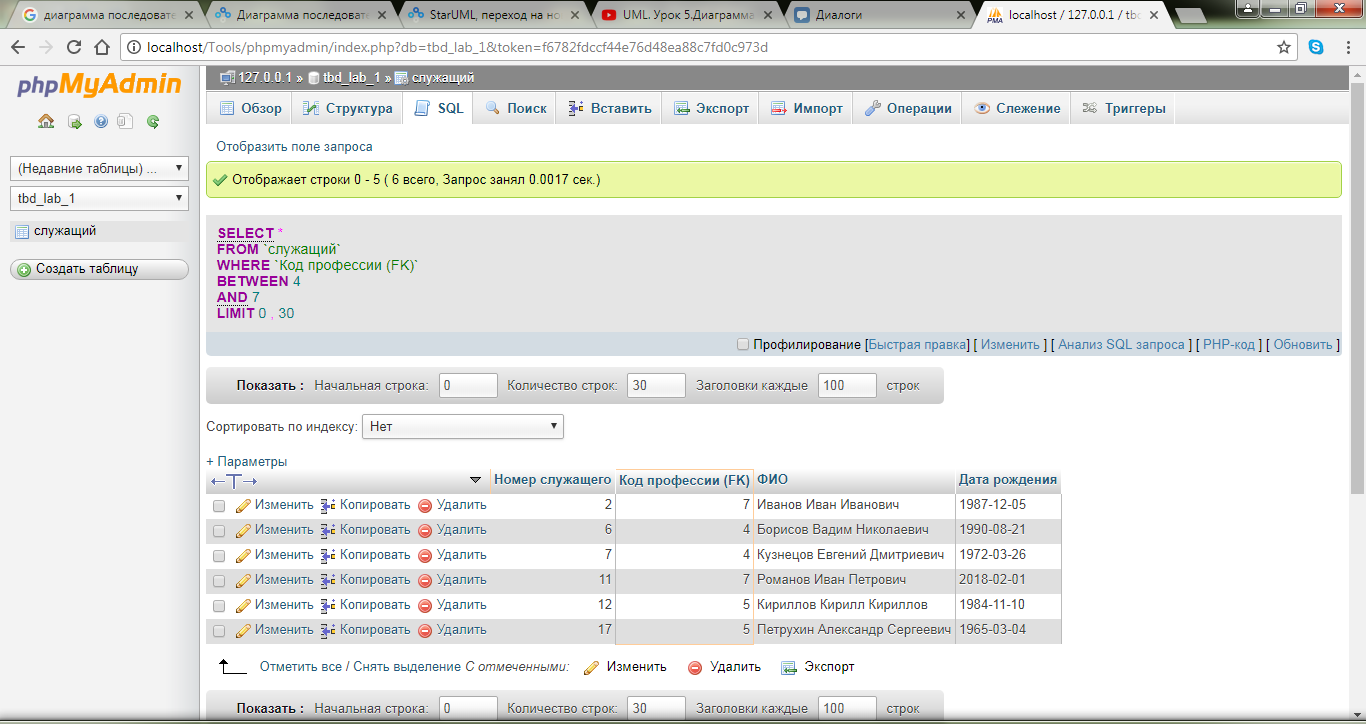
**

Рисунок 3.6 – Действия функций BETWEEN и AND

SELECT \* FROM `служащий` WHERE `Код профессии (FK)` IN ( 1, 7); *(Выбрать из таблицы Служащий только записи, где Код профессии (FK)» 1 или 7)*

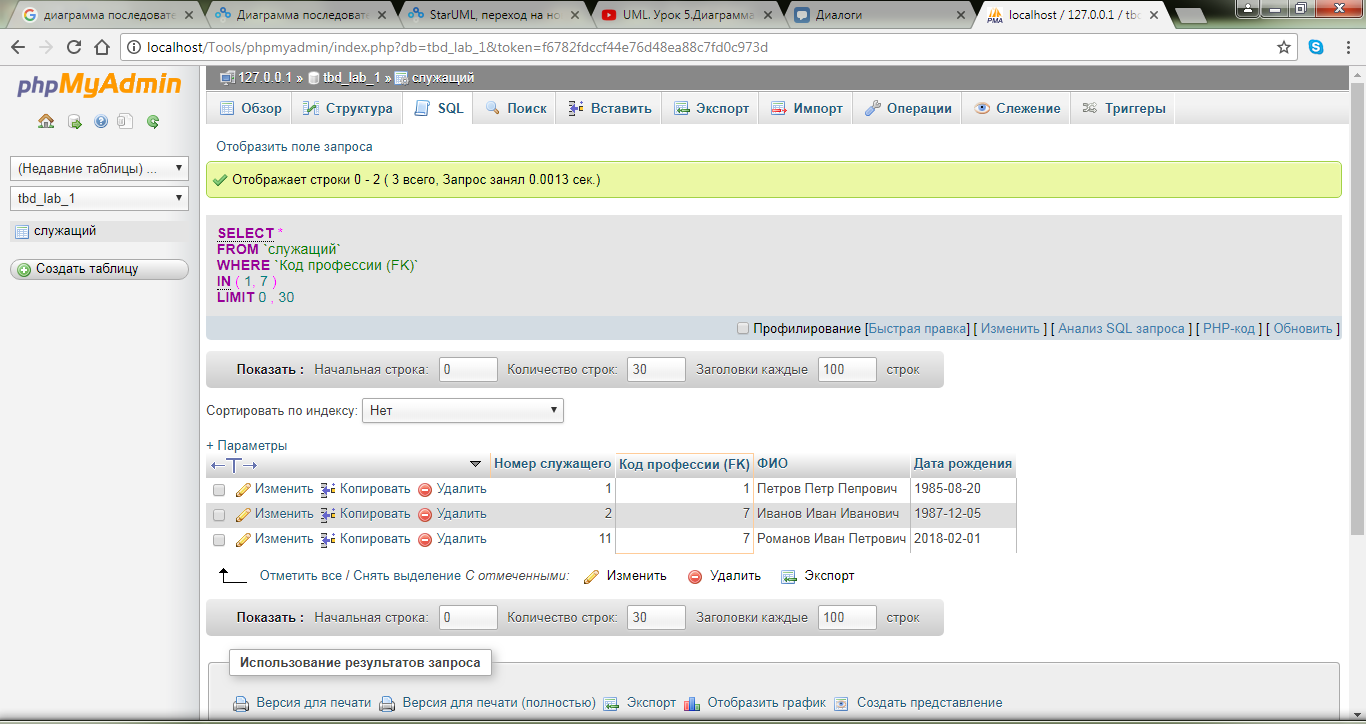


Рисунок 3.7 – Действия функции IN

SELECT \* FROM `служащий` WHERE `ФИО` LIKE 'П%';

*(Выбрать из таблицы Служащий только записи, где «ФИО» содержит «П…»)*

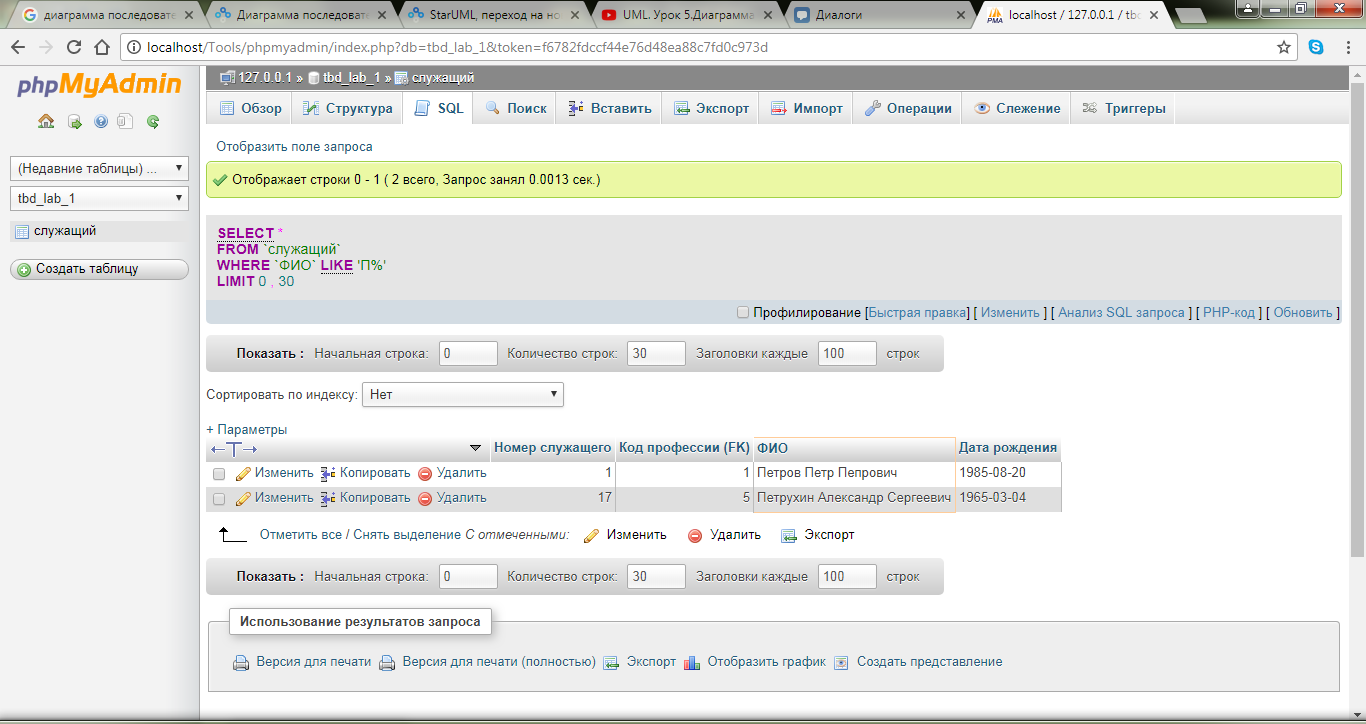
**

Рисунок 3.8 – Действия функции LIKE

SELECT \* FROM `служащий` WHERE `Номер служащего` IS NULL;

*(Выбрать из таблицы Служащий только записи, где «Номер служащего» пуст)*

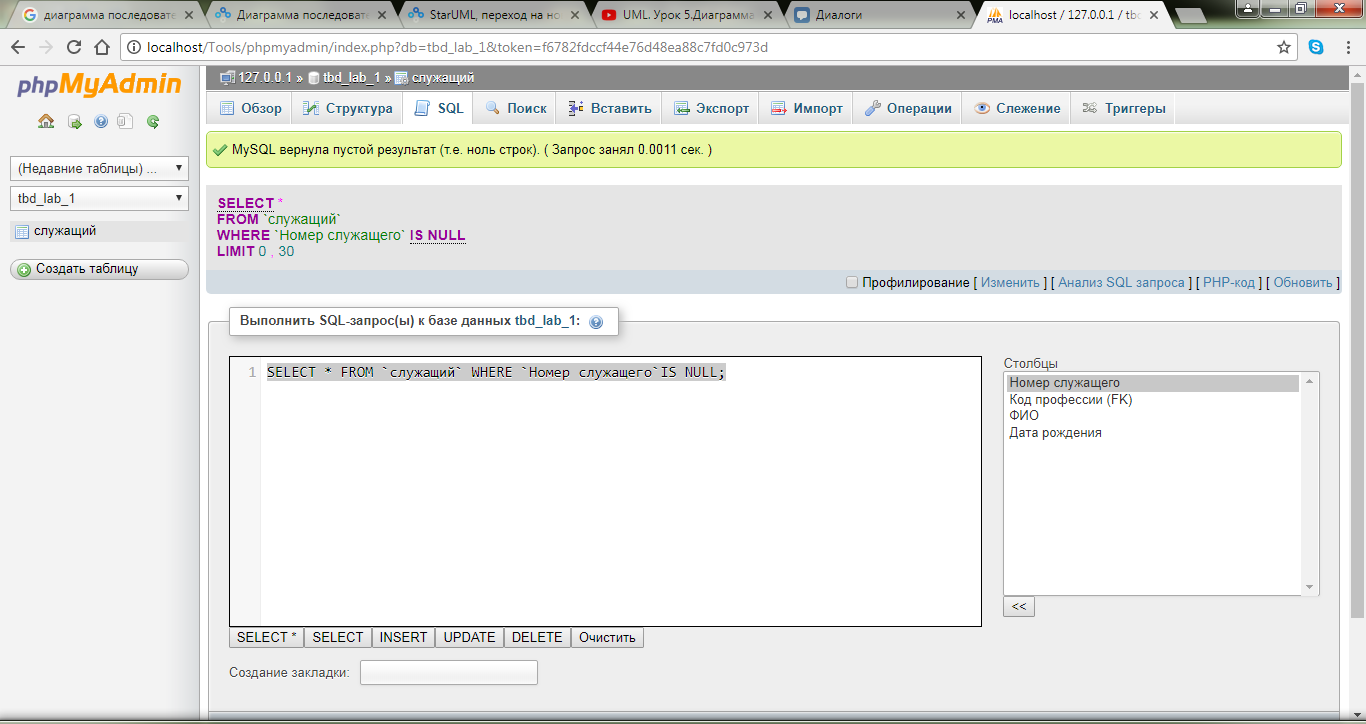


Рисунок 3.9 – Действия функции IS NULL

1. Демонстрация SQL-запроса специальных функций с условием NOT:

SELECT \* FROM `служащий` WHERE `ФИО` NOT LIKE 'П%';

*(Выбрать из таблицы Служащий только записи, где «ФИО» не содержит «П…»)*

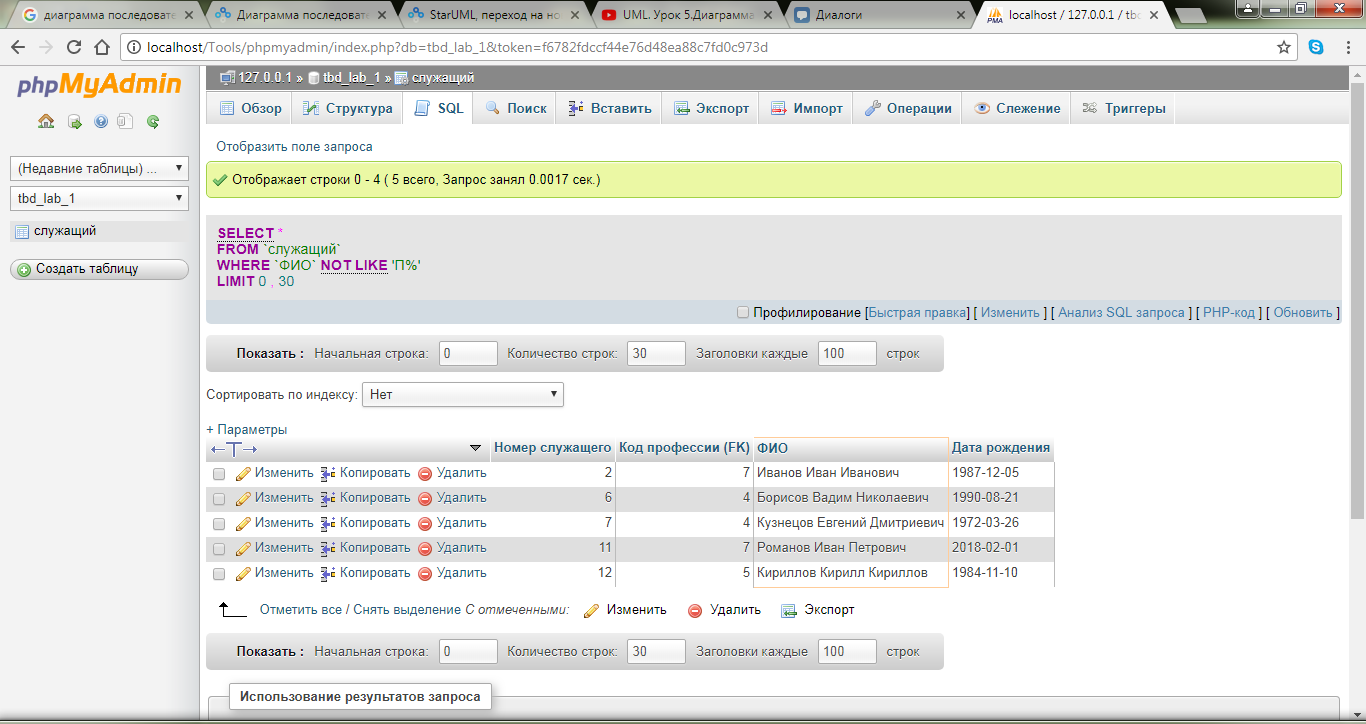
**

Рисунок 3.10 – Действия функции NOT

1. Составить SQL-запрос с одной из агрегатных функций:

SELECT MIN(`Код профессии (FK)`) FROM `служащий`;

*(Выбрать минимальный кодпрофессии из таблицы Служащий )*

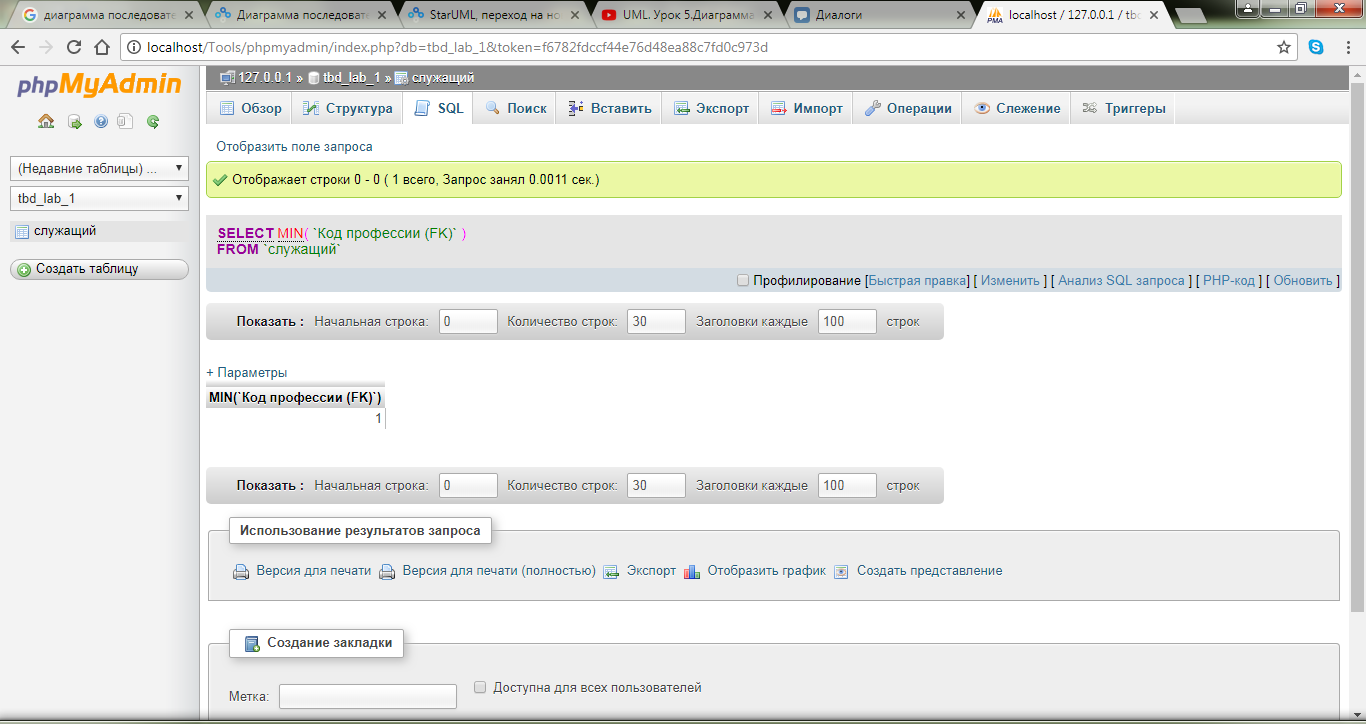


Рисунок 3.11 – Действия функции MIN

1. Демонстрация COUNT(\*)

SELECT COUNT(\*) FROM `служащий`;

*(Подсчитать все записи таблицы Служащий)*

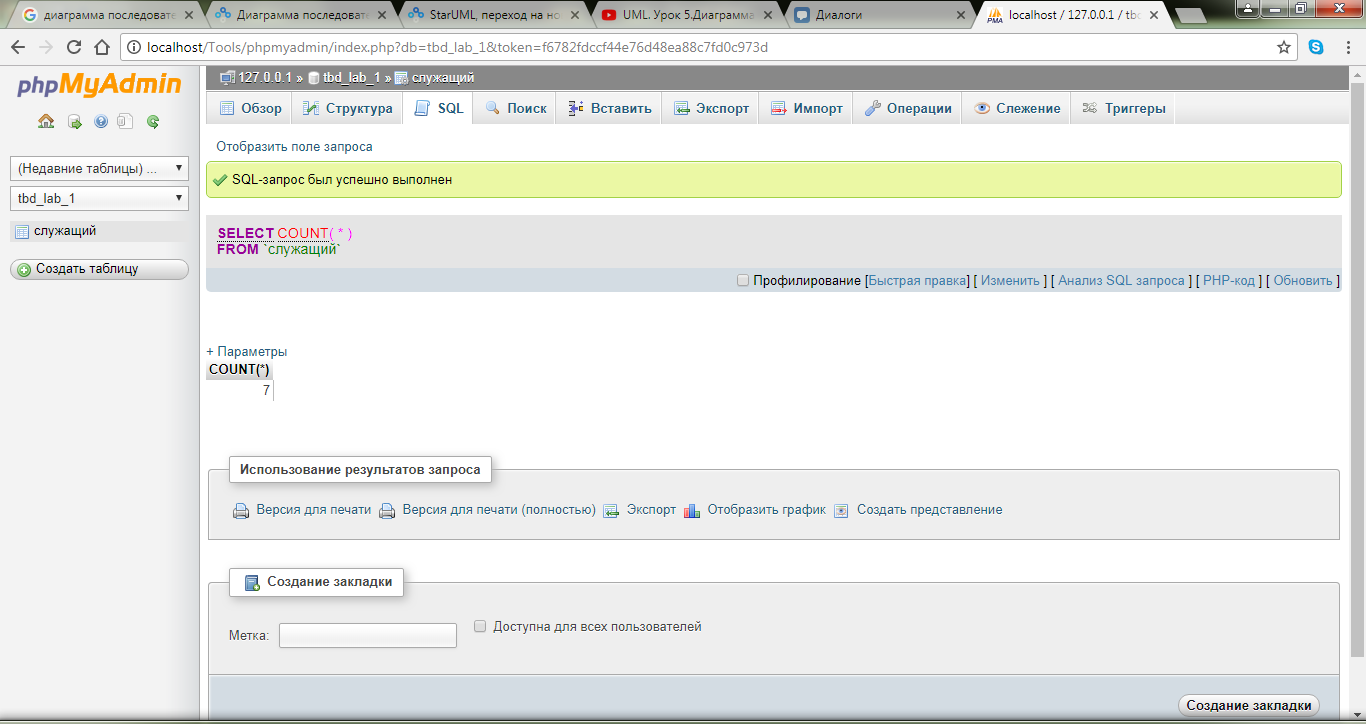
**

Рисунок 3.12 – COUNT(\*)

1. Демонстрация простых вычислений в запросе:

SELECT `Код профессии (FK)`\*2 FROM `служащий`;

*(Выбрать записи и удвоить номера служащих из таблицы Служащий)*

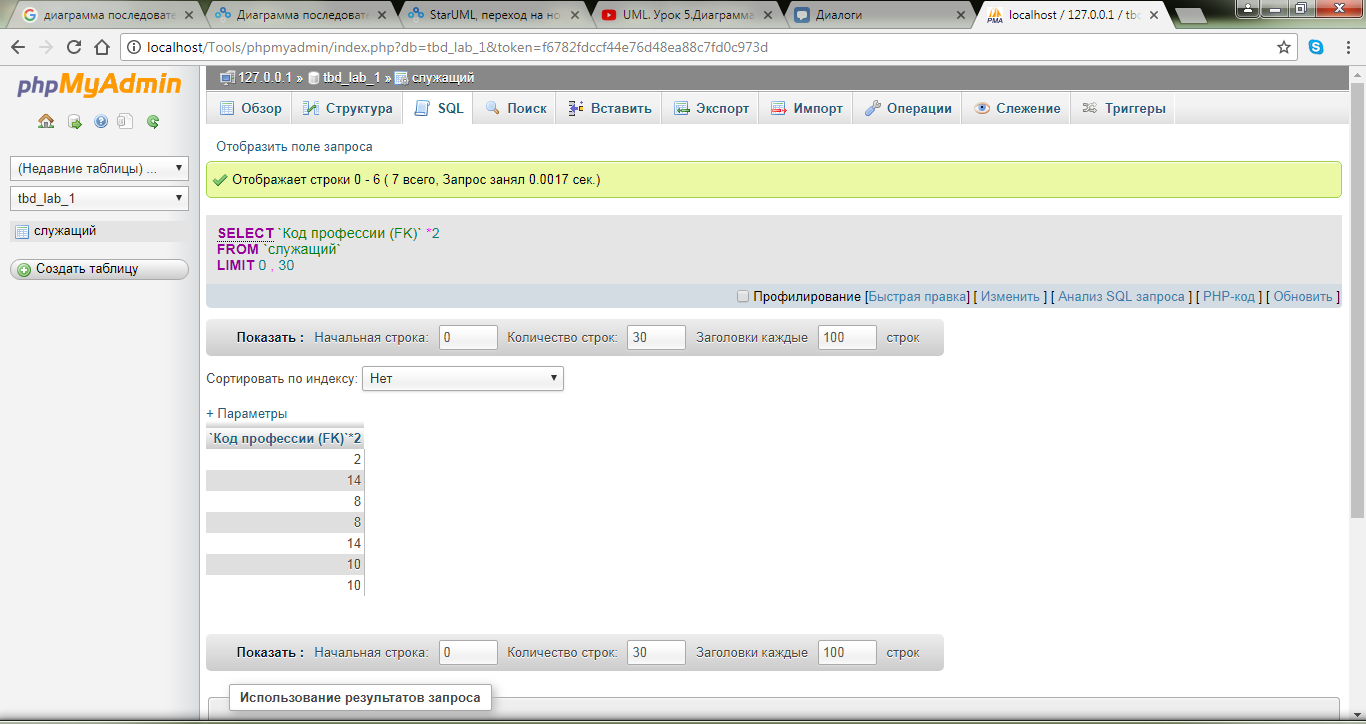


Рисунок 3.13 – Простые вычисления в запросе

1. Использование простого вычисления, как параметра агрегатной функции в запросе

SELECT MAX (`Номер служащего`+1) FROM `служащий`;

*(Выбрать максимальный* номер служащего *из таблицы Служащий и увеличить на 1)*

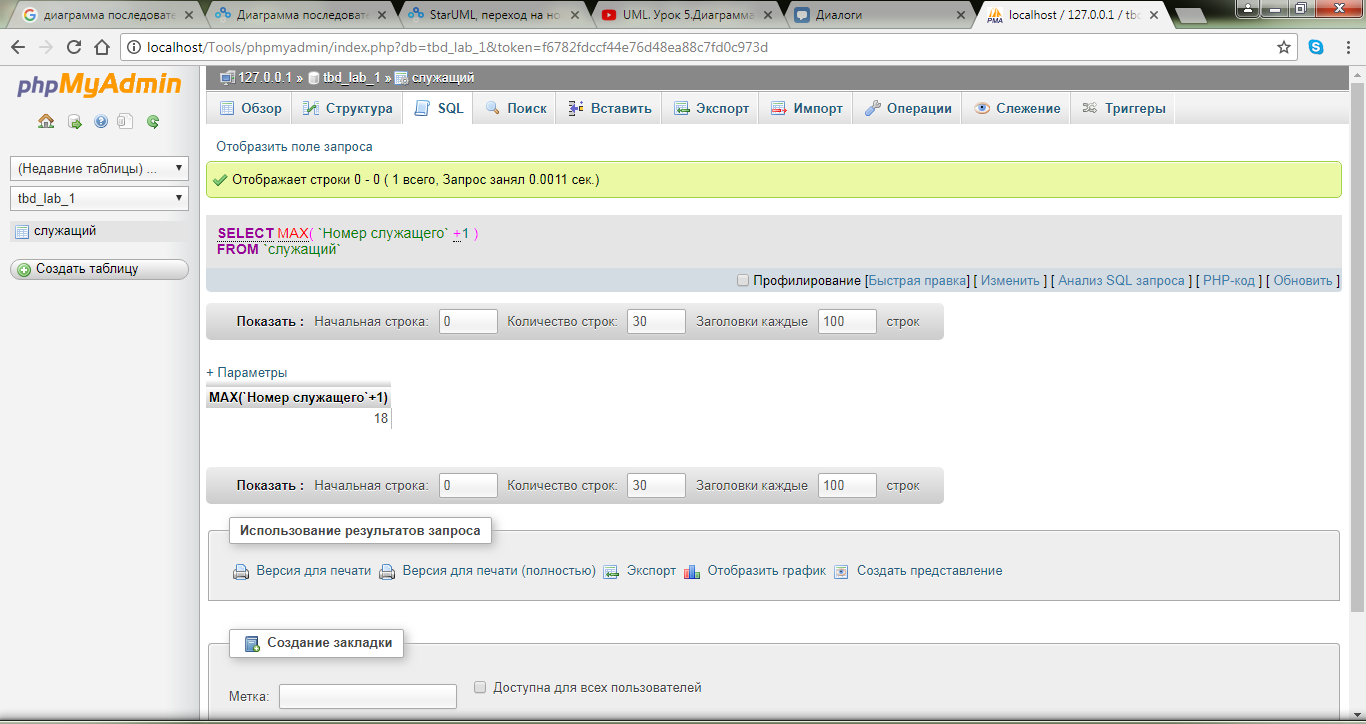
**

Рисунок 3.14 – Простые вычисления, как параметр агрегатной функции в запросе

1. Демонстрация работы функции HAVING

SELECT MAX( `Номер служащего` ) , `Код профессии (FK)`

FROM `служащий`

GROUP BY `Код профессии (FK)`

HAVING MAX( `Номер служащего` ) >3

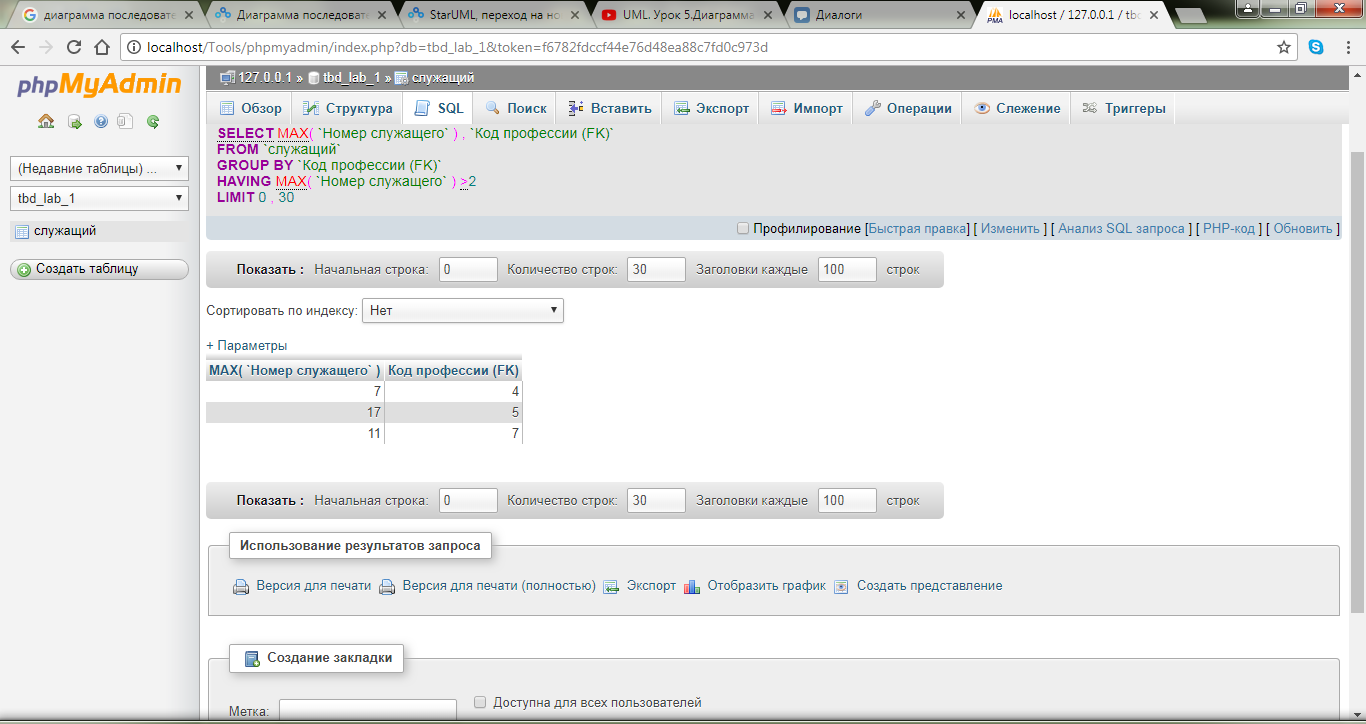


Рисунок 3.15 – Демонстрация работы функции HAVING

1. Результаты агрегатной функции по полю группировки

SELECT COUNT( `Код профессии (FK)` ) , `Код профессии (FK)`

FROM `служащий`

GROUP BY `Код профессии (FK)`

*(Подсчитать количество записей каждого кода профессии в таблице служащий и сгруппировать по коду профессии)*

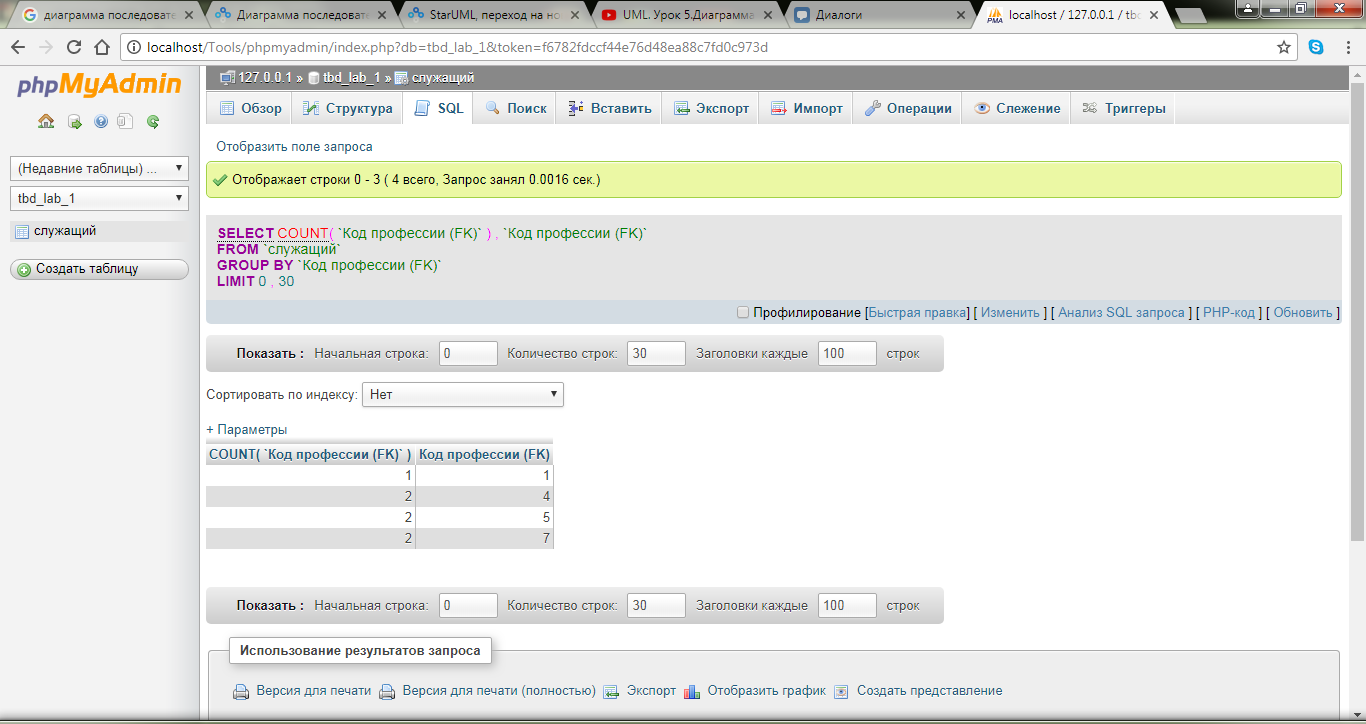
**

Рисунок 3.16 – Результаты агрегатной функции по полю группировки

ВЫВОД

В ходе лабораторной работы изучены принципы, синтаксис, назначение работы оператораSELECT, GROUP языка SQL и многих других функций и операторов с простыми и составными условиями, а также изучены возможности обработки данных с помощью агрегатных функций языка SQL.