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 №4

по дисциплине: «Теория баз данных»

по теме: «Ссылочная целостность данных. Создание схемы базы данных»

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

(дата)

Руководитель практикума:

доц. Волкова Т.В.

(должность) (подпись) (инициалы, фамилия)

Севастополь

2018

1 ЦЕЛЬ РАБОТЫ

Ознакомиться с принципами построения схемы базы данных и обеспечения целостности данных в базе.

2 ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Проанализировать схему БД своего варианта задания (вариант тот же, что и в лабораторной работе №1), выделить и классифицировать все существующие связи, определить необходимые ограничения целостности.
2. Создать еще не созданные таблицы, изменить существующие таким образом, чтобы они могли участвовать в связях (с помощью оператора ALTER ТАВLE).
3. В процессе создания таблиц установить связи между таблицами.
4. Составить запросы на ввод данных в главную и подчиненную таблицу.
5. Проверить работу ограничений на значения первичного ключа обеих таблиц внешнего ключа подчиненной таблицы.
6. Составить запросы на обновление и удаление данных для проверки работы ограничений целостности связей между таблицами. Проверить работу ограничений целостности в случаях установки каскадирования и запрета удаления и обновления данных.

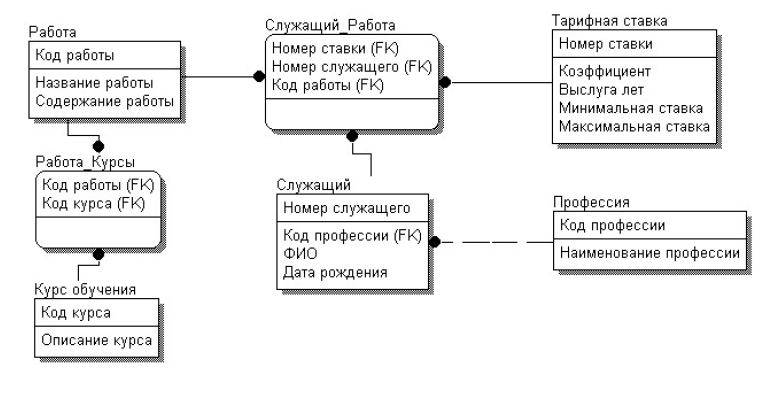


Рисунок 2.1 ─ Вариант задания

3 ВЫПОЛНЕНИЕ РАБОТЫ

1. Сброс и добавление нового первичного в таблицу «Служащий»:

ALTER TABLE `служащий` DROP PRIMARY KEY, ADD PRIMARY KEY (`Nomer slyzh`)

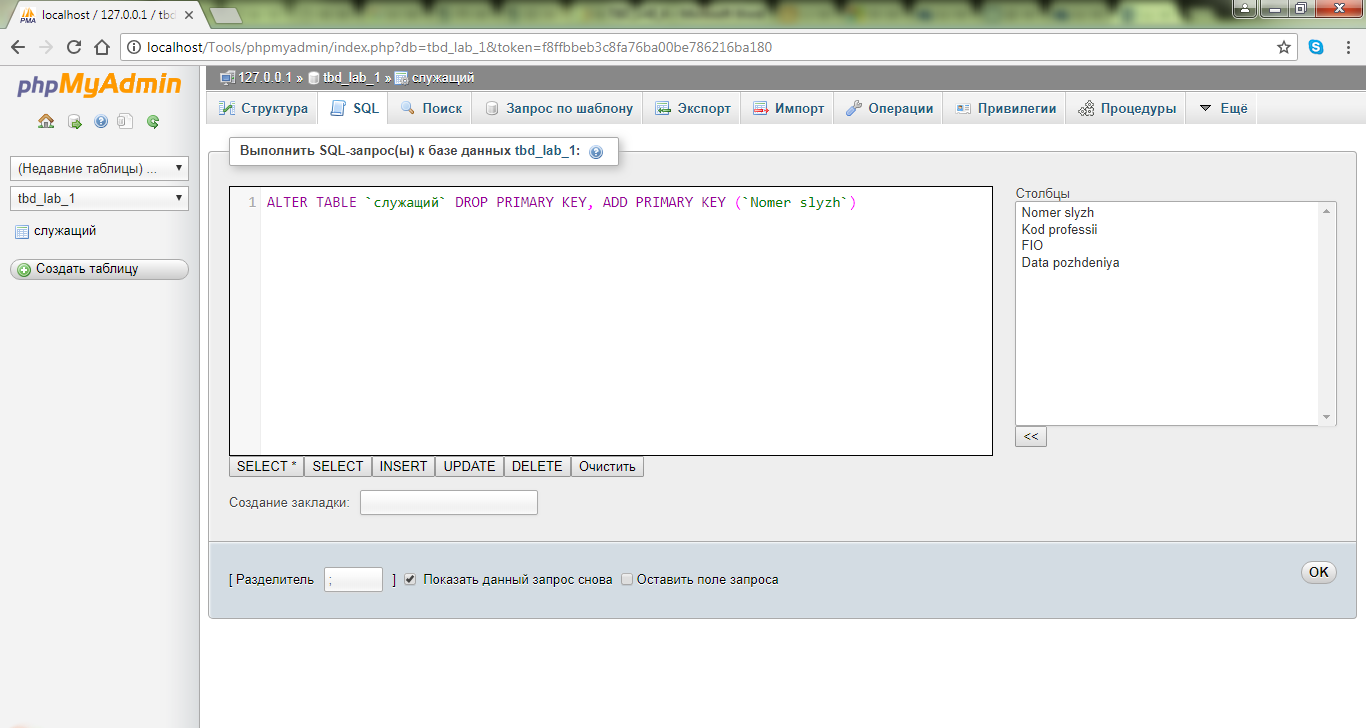


Рисунок 3.1 − Сброс и добавление нового первичного ключа в таблицу «Служащий»

1. Создание таблицы Профессия:

CREATE TABLE `Professiya` (

`Kod professii` int(6) NOT NULL PRIMARY KEY,

`Name professii` varchar(35)

);

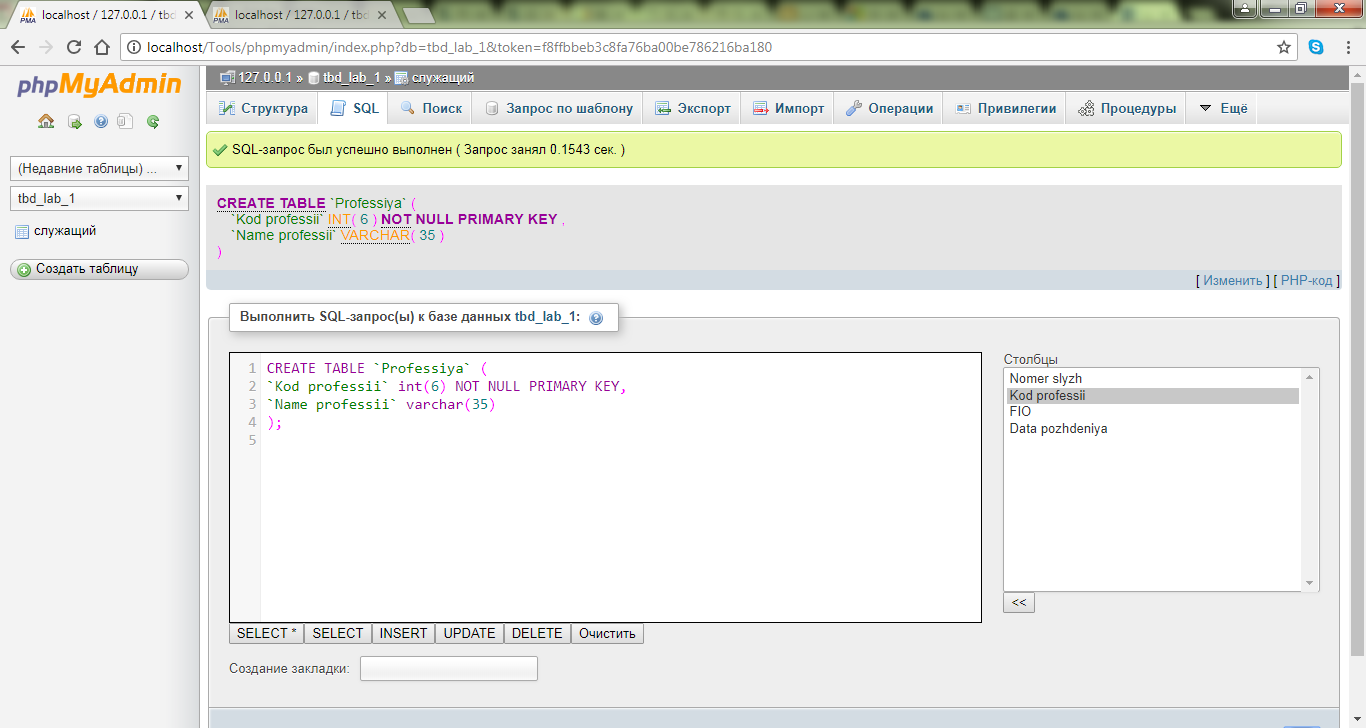


Рисунок 3.2 − Создание новой таблицы

1. Установка связи таблицы “`служащий` с таблицей “ Professiya`:

ALTER TABLE `служащий` ADD FOREIGN KEY (`Kod professii`) REFERENCES `tbd\_lab\_1`.`professiya`(`Kod professii`) ON DELETE CASCADE ON UPDATE CASCADE;

1. Добавление записи в родительскую таблицу:

INSERT INTO `tbd\_lab\_1`.`professiya` (`Kod professii`,`Name professii`) VALUES ('12', 'Строитель');

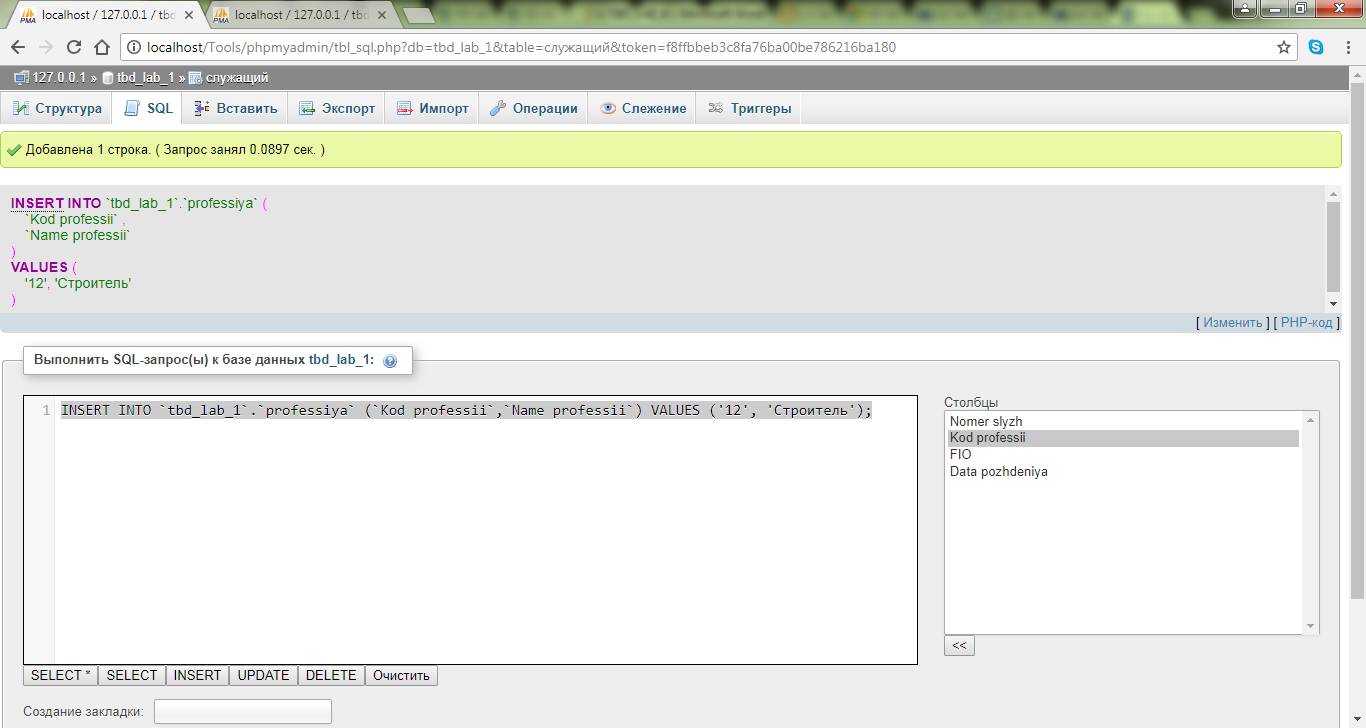


Рисунок 3.3 − Добавление записи в родительскую таблицу

1. Попытка добавить в порожденную таблицу кортеж с несуществующим кодом профессии в родительской таблице (ON DELETE CASCADE ON UPDATE CASCADE):

INSERT INTO `tbd\_lab\_1`.`служащий` (`Nomer slyzh`,`Kod professii`,`FIO`, `Data pozhdeniya`) VALUES ('1', '15','Иванов', '23.08.1999');

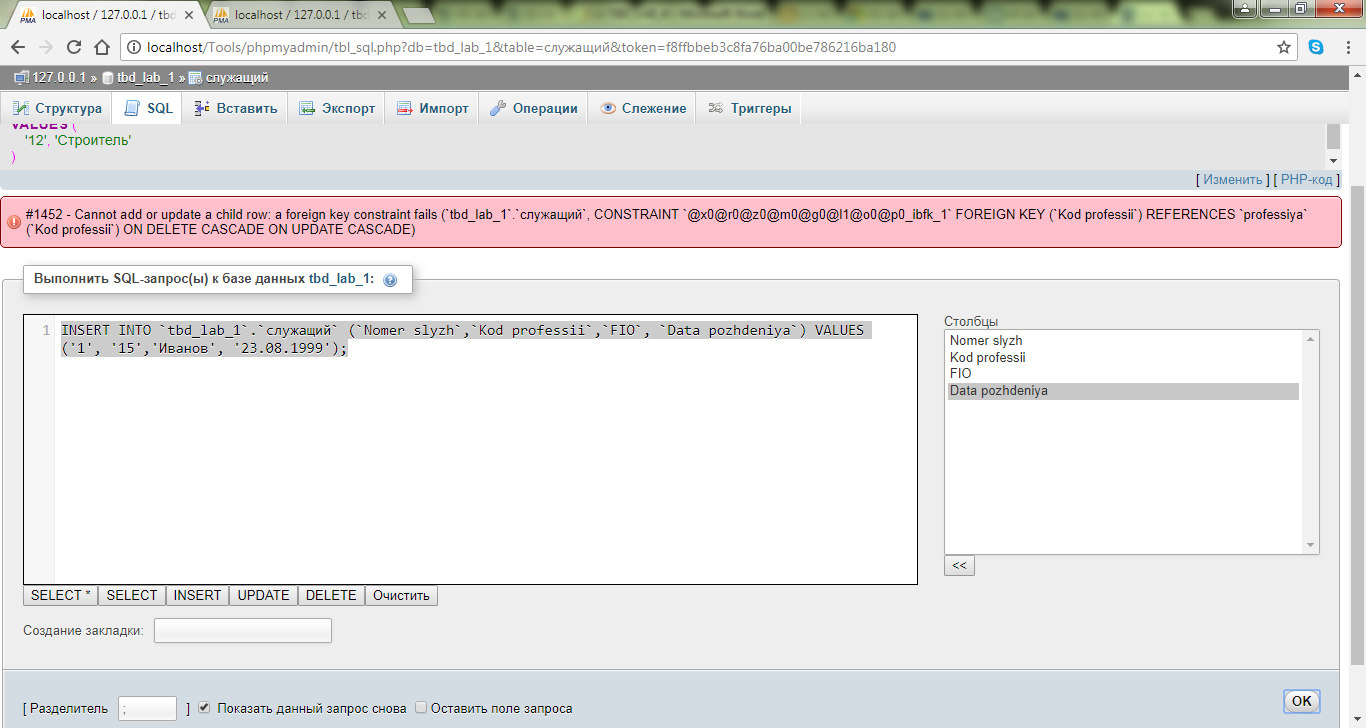


Рисунок 3.4 – Попытка добавления записи в порожденную таблицу с несуществующим кодом профессии

1. Добавление в порожденную таблицу кортеж с существующим кодом профессии в родительской таблице (ON DELETE CASCADE ON UPDATE CASCADE):

INSERT INTO `tbd\_lab\_1`.`служащий` (`Nomer slyzh`,`Kod professii`,`FIO`, `Data pozhdeniya`) VALUES ('1', '12','Иванов', '23.08.1999');

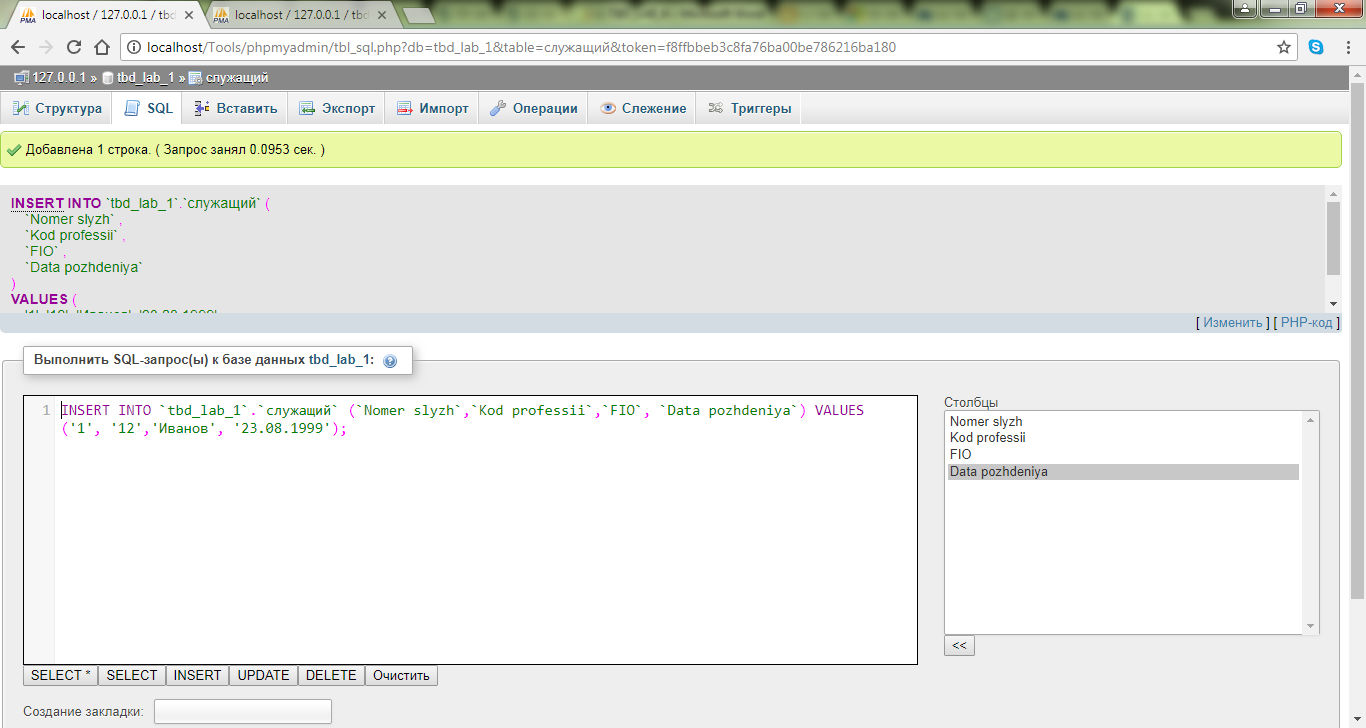


Рисунок 3.5 – Добавление записи в порожденную таблицу с существующим кодом профессии

1. Попытка обновления в порожденной таблице:

UPDATE `tbd\_lab\_1`.`служащий` set `Kod professii`='15';

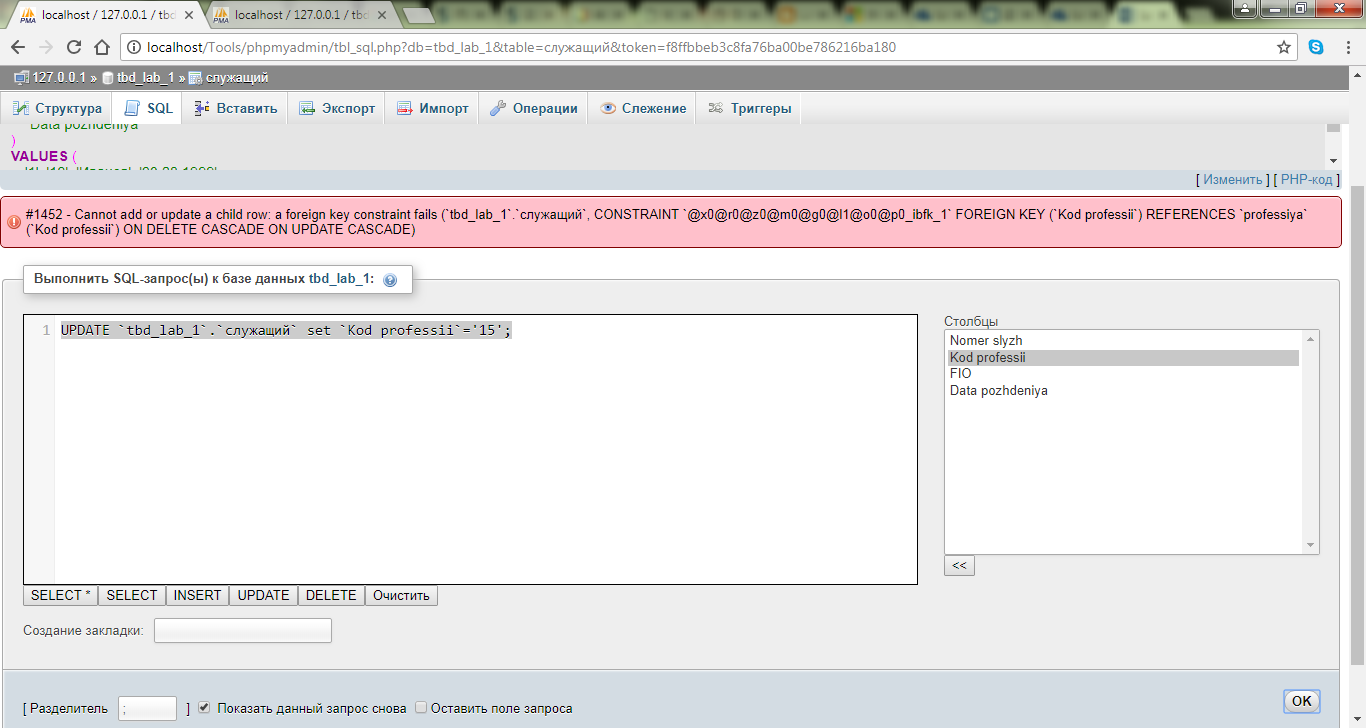


Рисунок 3.6 – Попытка обновления в порожденной таблице

1. Обновление в родительской таблице:

UPDATE `tbd\_lab\_1`.`professiya` set `Kod professii`='15';

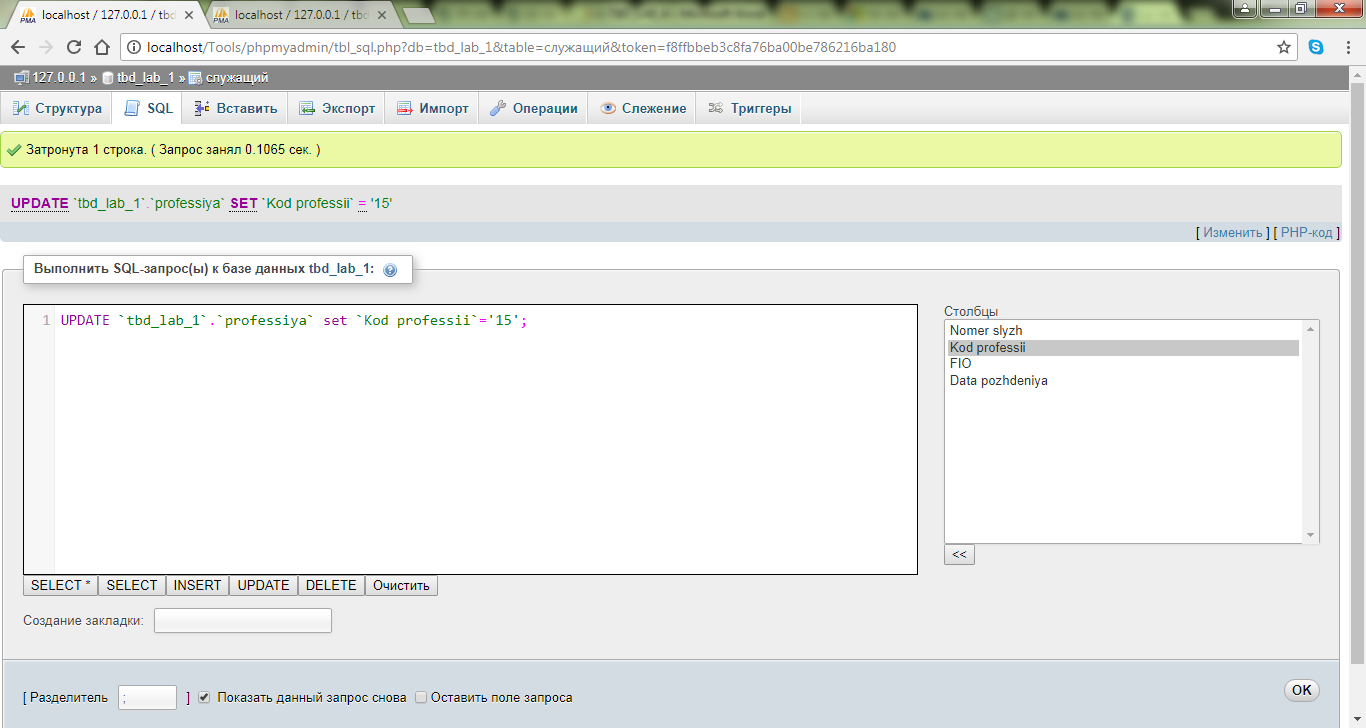


Рисунок 3.7 – Обновление в родительской таблице

1. Удаление из родительской таблицы:

DELETE FROM `professiya` WHERE `Kod professii`=15;

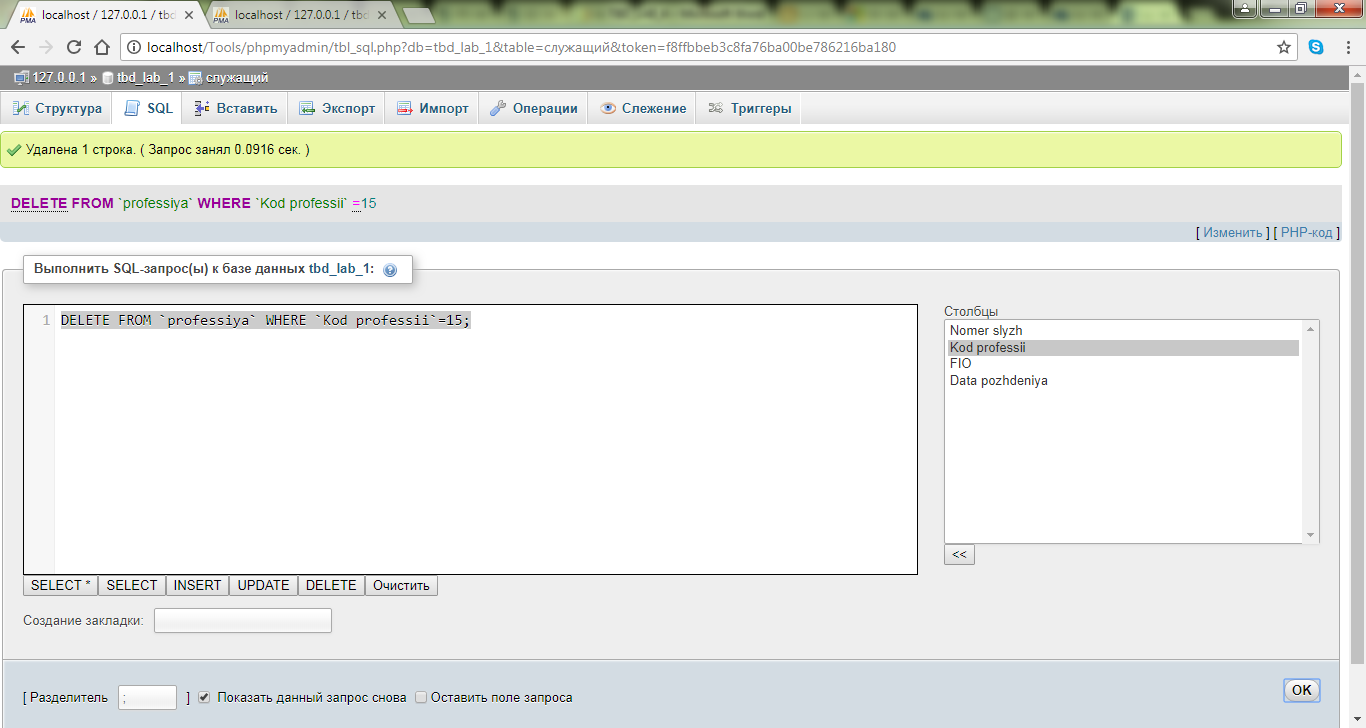


Рисунок 3.8 – Удаление из родительской таблицы

1. В случае если выставить ограничения на: ON DELETE NO ACTION ON UPDATE NO ACTION − запросы на удаление или изменение данных в родительские таблице не будут реализованы из-за ограничений.
2. Созданные таблицы:

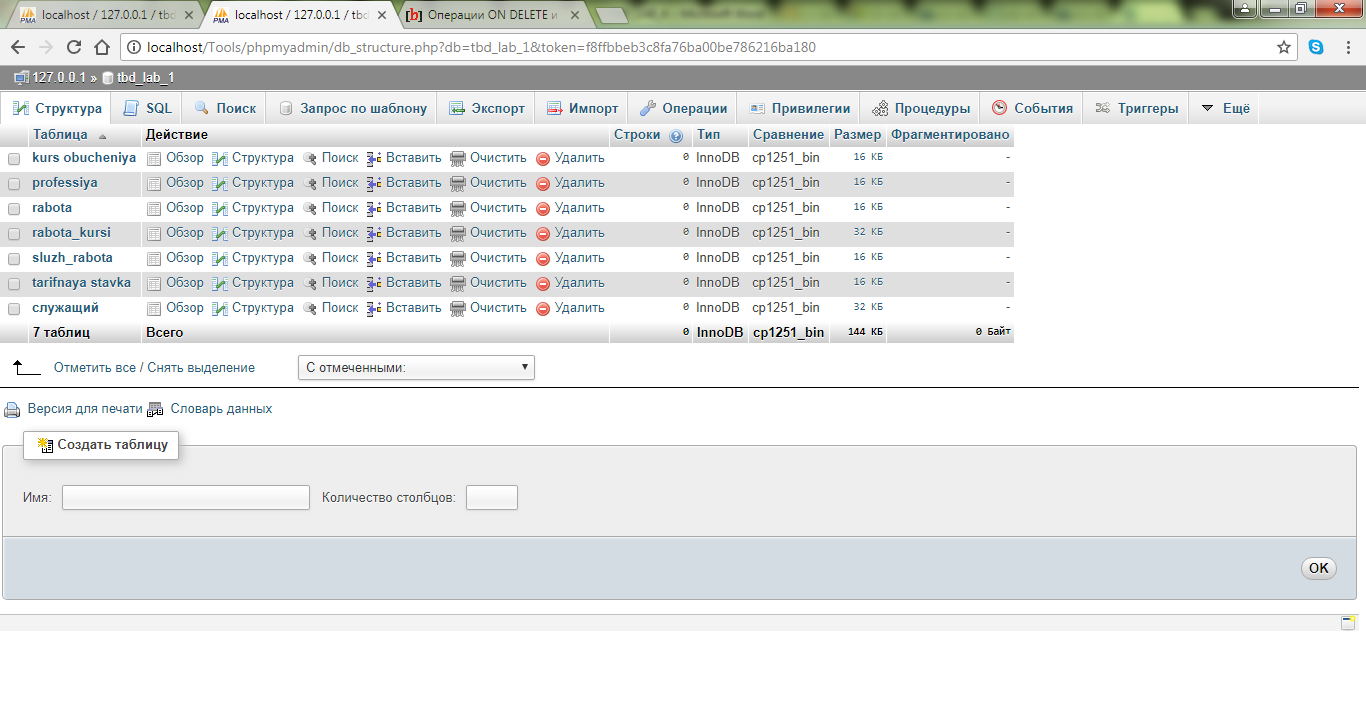


Рисунок 3.9 – Созданные таблицы

1. Связь таблицы rabota\_kursi с rabota и kurs obucheniya:

ALTER TABLE `rabota\_kursi` ADD FOREIGN KEY (`Kod raboti`) REFERENCES `tbd\_lab\_1`.`rabota`(`Kod raboti`) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE `rabota\_kursi` ADD FOREIGN KEY (`Kod kursa`) REFERENCES `tbd\_lab\_1`.`kurs obucheniya`(`Kod kursa`) ON DELETE CASCADE ON UPDATE CASCADE;

1. Связь таблицы rabota\_kursi с rabota и kurs obucheniya:

ALTER TABLE `sluzh\_rabota` DROP FOREIGN KEY ` ADD FOREIGN KEY (`Nomer stavki`) REFERENCES `tbd\_lab\_1`.`tarifnaya stavka`(`Nomer stavki`) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE `sluzh\_rabota` ADD FOREIGN KEY (`Nomer slyzh`) REFERENCES `tbd\_lab\_1`.`служащий`(`Nomer slyzh`) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE `sluzh\_rabota` DROP FOREIGN KEY ` ADD FOREIGN KEY (`Kod raboti`) REFERENCES `tbd\_lab\_1`.`Rabota`(`Kod raboti `) ON DELETE CASCADE ON UPDATE CASCADE;

1. Структура связей

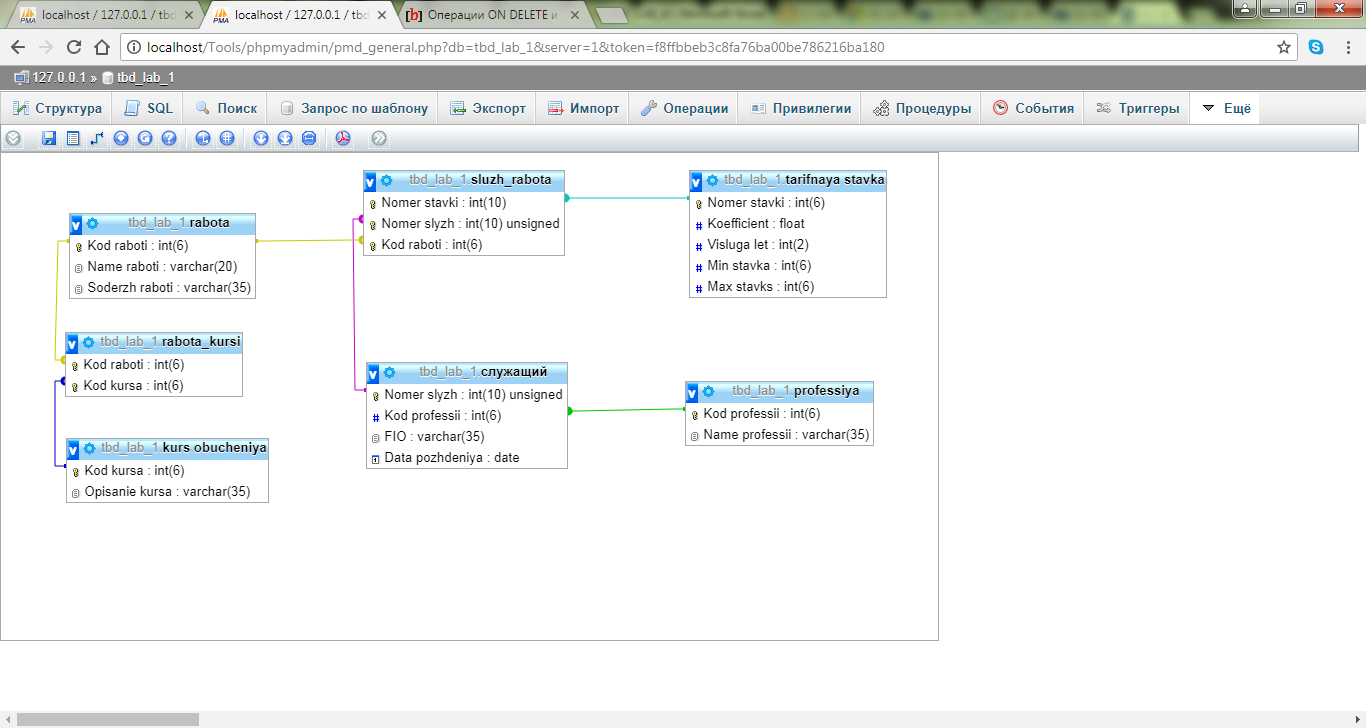


Рисунок 3.10 – Структура связей

ВЫВОДЫ

В ходе проделанной лабораторной работы был изучен принципы построения схем баз данных. Приобретены навыки обеспечения целостности данных в связанных таблицах. Освоены основные принципы работы с родительскими и порожденными таблицами.