Министерство образования и науки Российской Федерации

Севастопольский государственный университет

Кафедра ИС

Отчет

По дисциплине: “Теория баз данных”

Лабораторная работа №4

“ССЫЛОЧНАЯ ЦЕЛОСТНОСТЬ ДАННЫХ.

СОЗДАНИЕ СХЕМЫ БАЗЫ ДАННЫХ”

Выполнил:

ст.гр. ИС/б-22

Долженко И.А.

Проверила:

Лебедева М.А.

Севастополь

2019

1 ЦЕЛЬ РАБОТЫ

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

2 ПОСТАНОВКА ЗАДАЧИ

1. Выделить, классифицировать и описать все существующие связи, определить необходимые ограничения целостности.
2. С помощью SQL-запроса изменить существующую таблицу таким образом, чтобы она могла участвовать в связях с другими таблицами (например, с помощью оператора ALTER ТАВLE).
3. С помощью SQL-запросов создать еще не созданные таблицы. В процессе создания таблиц установить связи между таблицами.
4. Обязательно добавить во все таблицы записи (можно с помощью интерфейса phpMyAdmin).
5. Составить SQL-запросы на ввод данных в главную и подчиненную таблицу (выбрать любую пару таблиц). Проверить работу ограничений на значения первичного ключа обеих таблиц и внешнего ключа подчиненной таблицы.
6. Составить SQL-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.

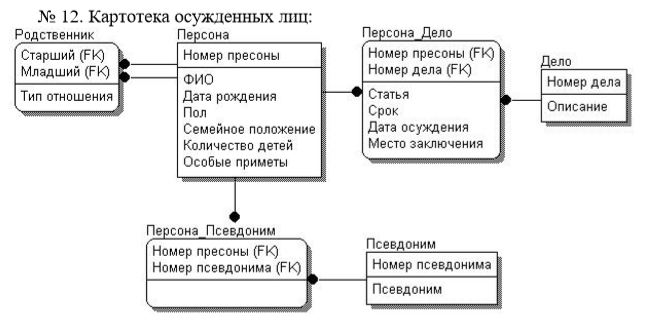


Рисунок 1 – Схема БД по варианту

3 ХОД РАБОТЫ

1. Создание таблицы Родственник

CREATE TABLE `Родственник`(

`Старший` VARCHAR(255) NOT NULL,

`Младший` VARCHAR(255) NOT NULL,

`Тип отношения` VARCHAR(50) NOT NULL,

PRIMARY KEY (`Старший`, `Младший`),

FOREIGN KEY(`Старший`) REFERENCES `Персона`(`ФИО`) ON UPDATE CASCADE ON DELETE CASCADE,

FOREIGN KEY(`Младший`) REFERENCES `Персона`(`ФИО`) ON UPDATE CASCADE ON DELETE CASCADE

)

2. Создание таблицы Дело

CREATE TABLE `Дело`(

`Номер дела` INT NOT NULL PRIMARY KEY,

`Описание` VARCHAR(255) NOT NULL

)

3. Создание таблицы Персона\_дело

CREATE TABLE `Персона\_дело`(

`Номер персоны` INT NOT NULL,

`Номер дела` INT NOT NULL,

`Статья` VARCHAR(255) NOT NULL,

`Срок` INT NOT NULL,

`Дата осуждения` DATE NOT NULL,

`Место заключения` VARCHAR(255) NOT NULL,

PRIMARY KEY (`Номер персоны`, `Номер дела`),

FOREIGN KEY(`Номер персоны`) REFERENCES `Персона`(`Номер персоны`) ON UPDATE CASCADE ON DELETE CASCADE,

FOREIGN KEY(`Номер дела`) REFERENCES `Дело`(`Номер дела`) ON UPDATE CASCADE ON DELETE CASCADE

)

4. Создание таблицы Псевдоним

CREATE TABLE `Псевдоним`(

`Номер псевдонима` INT NOT NULL PRIMARY KEY,

`Псевдоним` VARCHAR(255) NOT NULL

)

5. Создание таблицы Персона\_Псевдоним

CREATE TABLE `Персона\_Псевдоним`(

`Номер персоны` INT NOT NULL,

`Номер псевдонима` INT NOT NULL,

PRIMARY KEY(`Номер персоны`, `Номер псевдонима`),

FOREIGN KEY(`Номер персоны`) REFERENCES `Персона`(`Номер персоны`) ON UPDATE CASCADE ON DELETE CASCADE,

FOREIGN KEY(`Номер псевдонима`) REFERENCES `Псевдоним`(`Номер псевдонима`) ON UPDATE CASCADE ON DELETE CASCADE

)

6. Заполним созданные таблицы данными:



Рисунок 2 – Данные в существующей таблице Персона



Рисунок 3 – Заполнение таблицы Псевдоним

Заполним таблицу Персона-псевдоним используя данные родительских таблиц.

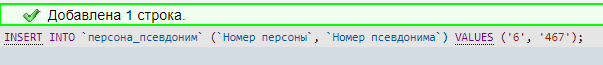


Рисунок 4 – Заполнение таблицы Персона-псевдоним



Рисунок 5 – Заполнение таблицы Дело

Заполним таблицу Персона-дело используя данные родительских таблиц.



Рисунок 6 – Заполнение таблицы Персона-дело

7. Попытка добавить запись в таблицу Персона-дело не удалась, т.к. в родительской для нее таблице Дело нет записи с номером дела 5.

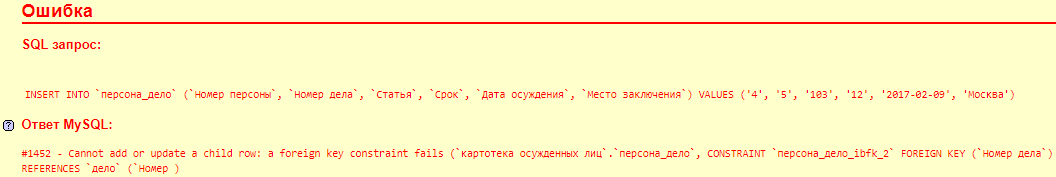


Рисунок 7 – Ошибка при вводе не существующего номера дела

Введем ту же запись, но уже с существующим номером дела. Запись добавлена.



Рисунок 8 – Успешное добавление записи с существующим номером дела

8. Удалим запись в таблице Дело.



Рисунок 9 – Таблица Дело до удаления

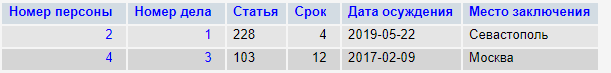


Рисунок 10 – Таблица Персона-дело до удаления

DELETE FROM `дело` WHERE `дело`.`Номер дела` = 1



Рисунок 11 – Таблица Дело после удаления



Рисунок 12 – Таблица Персона-дело после удаления

Видим, что произошло удаление в дочерней таблице Персона-дело после удаления записи в родительской таблице Дело.

9. Изменим запись в таблице Дело.

[UPDATE](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/update.html) `дело` [SET](http://127.0.0.1/openserver/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/set.html) `Номер дела` = '9' WHERE `дело`.`Номер дела` = 3



Рисунок 13 – Таблица Дело после изменения



Рисунок 14 – Таблица Персона-дело после изменения

В результате произошло изменение в дочерней таблице Персона-дело после изменений номера дела в родительской таблице Дело.

ВЫВОД

В ходе данной лабораторной работы были изучены принципы построения схемы баз данных и обеспечение целостности данных в базе. Была создана база данных согласно варианту задания, состоящая из нескольких связанных таблиц. В ходе создания таблиц были описаны ограничения их целостности исходя из анализа структуры данных. Целостность данных была проверена с помощью специально созданных запросов.