МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«Севастопольский государственный университет»

Кафедра «Информационные технологии и компьютерные системы»

ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №7

по дисциплине

«Проектирование баз данных»

Вариант 8

Выполнил:

Донец Н.О.

Проверил:

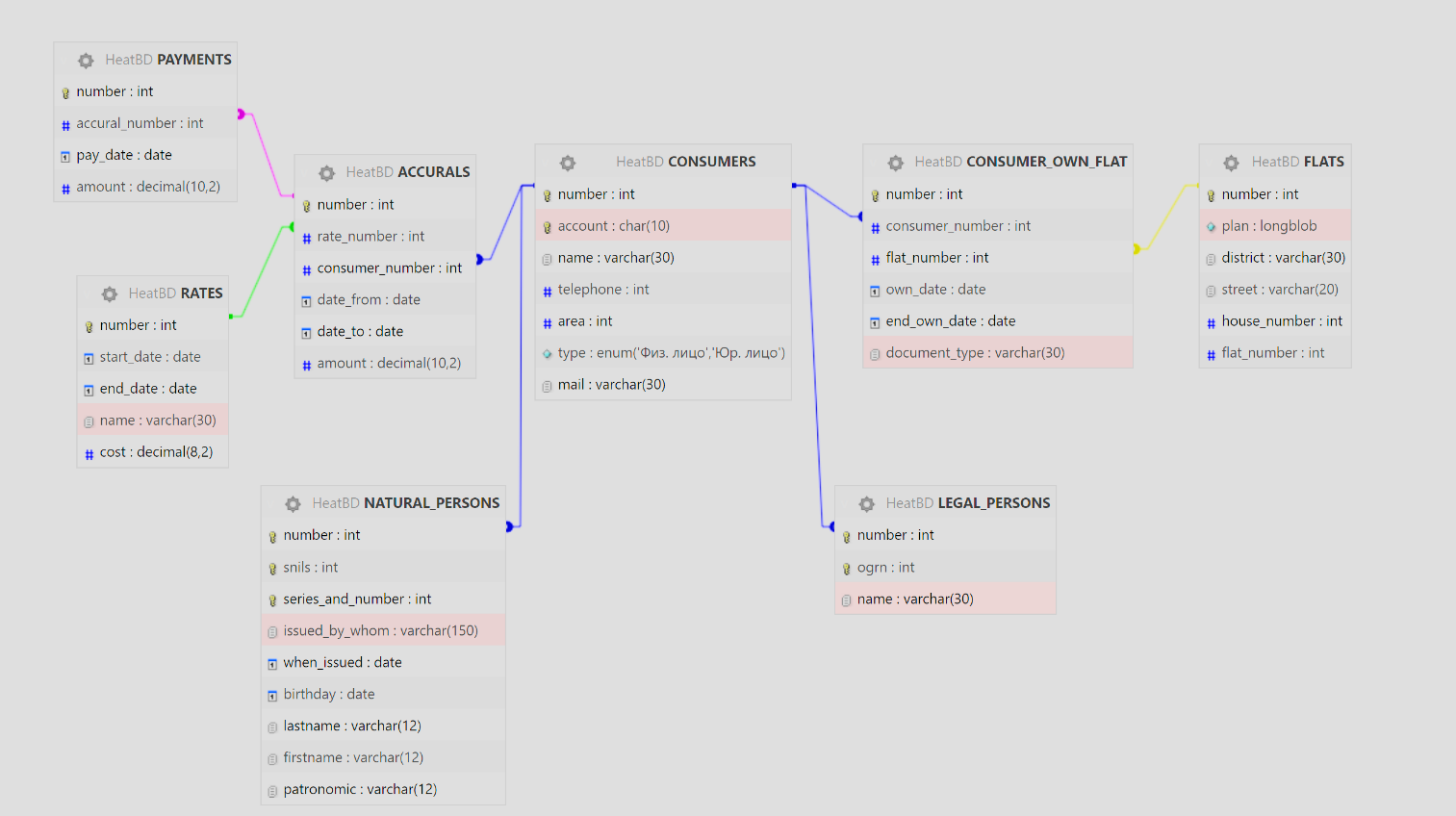
Волкова Т. В.

Севастополь

2024 г.

**Задание**

Разработать БД для ежемесячного учета оплаты услуг за теплоснабжение в теплосети района. БД должна содержать следующие данные: адрес (район, улица, номера дома и квартиры), план и площадь квартиры, месячные тарифы за потребленное тепло из расчета за 1 квадратный метр площади, общая плата за услуги, сведения об уплате с указанием дат и сумм, показатели оплаты услуг за отчетное полугодие по домам.



**5. ТЕСТИРОВАНИЕ БАЗЫ ДАННЫХ**

* 1. **Запросы на добавление данных в таблицы базы данных**

Были созданы запросы на добавление данных во все таблицы. Результаты добавления данных приведены на рисунках 1 - 8.

SQL запрос на добавление данных в таблицу RATES:

INSERT INTO RATES VALUES (0, '2024-01-01', '2025-01-01', 'Землянка', 3080);

INSERT INTO RATES (start\_date, end\_date, number, cost) VALUES ('2024-02-01', '2025-02-01', 1, 2490);

INSERT INTO RATES (number, start\_date, end\_date, name, cost) VALUES (2, '2024-06-01', '2025-06-01', 'Тропики', 3666);

INSERT INTO RATES VALUES (3, '2024-03-01', '2025-01-01', 'Первый круг ада', 4010), (4, '2024-04-01', '2025-01-01', 'Второй круг ада', 4020), (5, '2024-05-01', '2025-01-01', 'Третий круг ада', 4030);

SQL запрос на добавление данных в таблицу CONSUMERS:

INSERT INTO CONSUMERS (number, account, name, telephone, area, type, mail) VALUES (0, '70ББ000584', 'Владимир', '79781195738', 40, 'Физ. лицо', 'misterincognito@mail.ru');

INSERT INTO CONSUMERS VALUES (1, '80ББ000584', 'Anonymous', '79788888888', 350, 'Физ. лицо', 'nevladimir@mail.ru');

INSERT INTO CONSUMERS (area, number, account) VALUES (56, 2, '90ББ000584');

INSERT INTO CONSUMERS VALUES (3, '77ББ000584', 'ООО Красавцы', '79787777777', 40, 'Юр. лицо', 'krasavci@mail.ru'), (4, '79ББ000584', 'ЗАО Свинец', '79783333333', 40, 'Юр. лицо', 'svinec@mail.ru');

SQL запрос на добавление данных в таблицу FLATS:

INSERT INTO FLATS (number, plan, district, street, house\_number, flat\_number) VALUES (0, LOAD\_FILE('C:\Users\k\_dod\Downloads\ггшл.jpeg'), 'Гагаринский', 'Пушкина', 4, 12);

INSERT INTO FLATS VALUES (1, LOAD\_FILE('C:\Users\k\_dod\Downloads\planirovka-3komnatnoi-kvartiry-novostroika.jpeg'), 'Ленинский', 'Колотушкина', 2, 15);

INSERT INTO FLATS (district, number) VALUES ('Киевский', 2);

INSERT INTO FLATS VALUES (3, LOAD\_FILE('C:\Users\k\_dod\Downloads\ггшл.jpeg'), 'Балаклавский', 'Киевская', 14, 15), (4, LOAD\_FILE('C:\Users\k\_dod\Downloads\ггшл.jpeg'), 'Балаклавский', 'Киевская', 14, 16);

SQL запрос на добавление данных в таблицу CONSUMER\_OWN\_FLAT:

INSERT INTO CONSUMER\_OWN\_FLAT (number, consumer\_number, flat\_number, own\_date, end\_own\_date, document\_type) VALUES (0, 0, 0, '2023-01-01', '2025-01-01', 'Договор аренды');

INSERT INTO CONSUMER\_OWN\_FLAT VALUES (1, 0, 1, '2023-01-01', '2025-01-01', 'Выписка из ЕГРН');

INSERT INTO CONSUMER\_OWN\_FLAT (consumer\_number, flat\_number, number) VALUES (1, 2, 2);

INSERT INTO CONSUMER\_OWN\_FLAT (number, consumer\_number, flat\_number) VALUES (3, 2, 1), (4, 3, 3), (5, 4, 4);

SQL запрос на добавление данных в таблицу LEGAL\_PERSONS:

INSERT INTO LEGAL\_PERSONS (number, ogrn, name) VALUES (3, 1027739642281, "ООО Красавцы");

INSERT INTO LEGAL\_PERSONS (ogrn, number) VALUES (8327779642299, 4);

SQL запрос на добавление данных в таблицу NATURAL\_PERSONS:

INSERT INTO NATURAL\_PERSONS (number, snils, series\_and\_number, issued\_by\_whom, when\_issued, birthday, lastname, firstname, patronomic) VALUES (0, '48095351208', '0913599987', 'УМВД России по г. Севастополю', '1985-03-04', '1971-03-04', 'Пейпиво', 'Виктор', 'Альбертович');

INSERT INTO NATURAL\_PERSONS (snils, number, series\_and\_number) VALUES ('55544333208', 1, '0817666333');

Также для корректной работы был создан триггер для автоматического подсчёта суммы начисления.

SQL запрос на создание триггера для таблицы ACCURALS:

CREATE TRIGGER `NEW\_ACCURAL\_BEFORE\_INSERT`

BEFORE INSERT ON `ACCURALS`

FOR EACH ROW

set NEW.amount = (SELECT cost\*(SELECT area FROM CONSUMERS WHERE CONSUMERS.number = NEW.consumer\_number) FROM RATES WHERE RATES.number = NEW.rate\_number)

SQL запрос на добавление данных в таблицу ACCURALS:

INSERT INTO ACCURALS (number, rate\_number, consumer\_number, date\_from, date\_to) VALUES (0, 0, 0, '2024-01-01', '2024-02-01'), (1, 1, 1, '2024-01-01', '2024-02-01');

INSERT INTO ACCURALS (rate\_number, number, consumer\_number) VALUES (2, 2, 2), (3, 3, 3), (4, 4, 4);

SQL запрос на добавление данных в таблицу PAYMENTS:

INSERT INTO PAYMENTS (number, accural\_number, pay\_date, amount) VALUES (0, 0, '2024-02-02', 123200);

INSERT INTO PAYMENTS VALUES (1, 1, '2024-02-02', 871500);

INSERT INTO PAYMENTS (accural\_number, number, amount) VALUES (2, 2, 205296);

INSERT INTO PAYMENTS (number, accural\_number, amount) VALUES (3, 3, 160400), (4, 4, 160800);

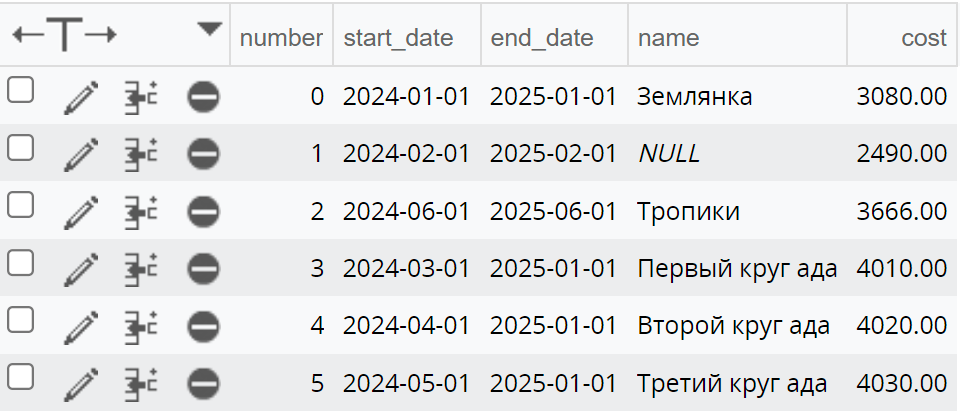


Рисунок 1 – Таблица RATES

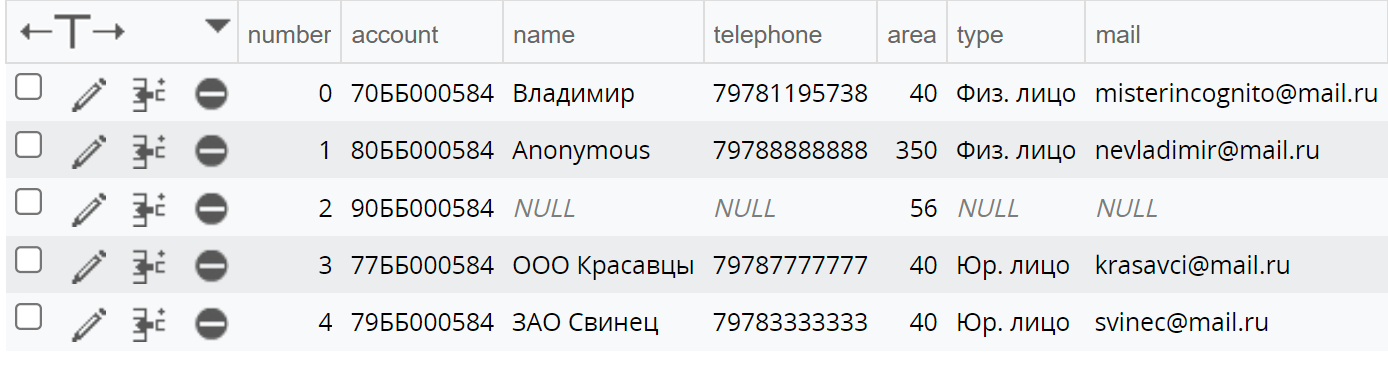


Рисунок 2 – Таблица CONSUMERS

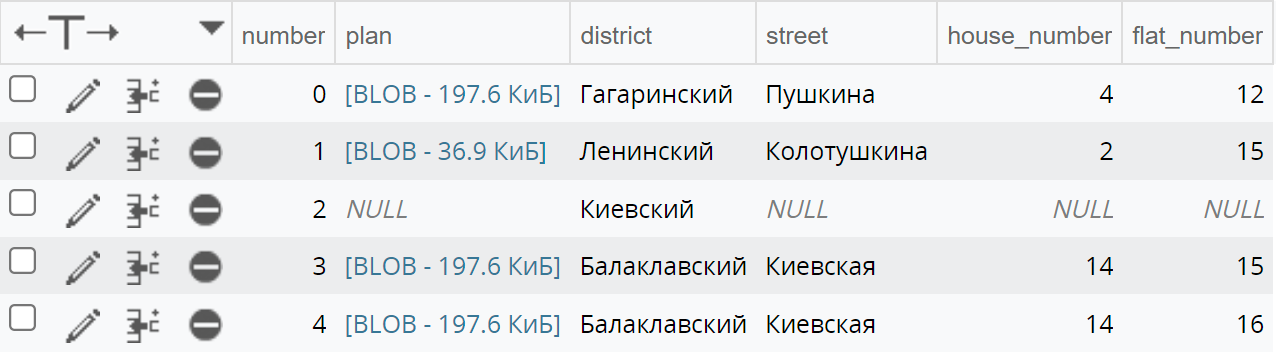


Рисунок 3 – Таблица FLATS

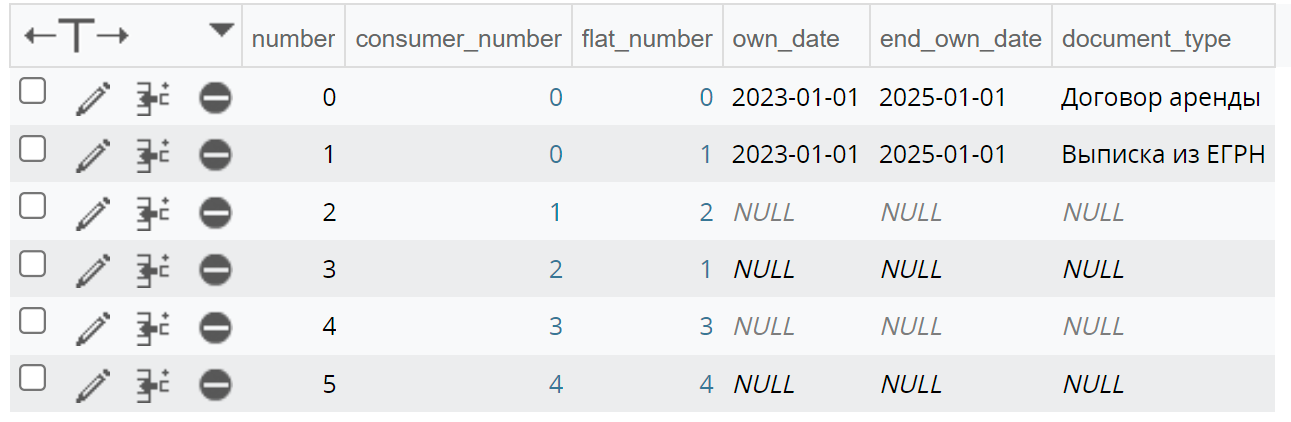


Рисунок 4 – Таблица CONSUMER\_OWN\_FLAT

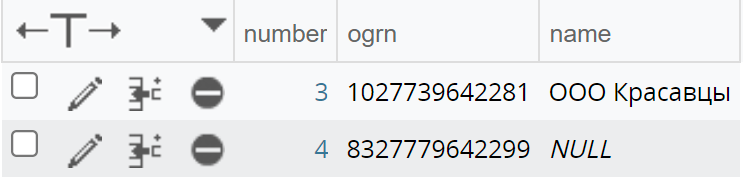


Рисунок 5 – Таблица LEGAL\_PERSON

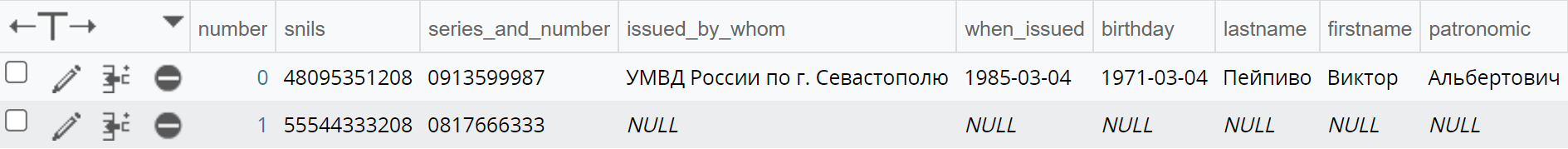


Рисунок 6 – Таблица NATURAL\_PERSON

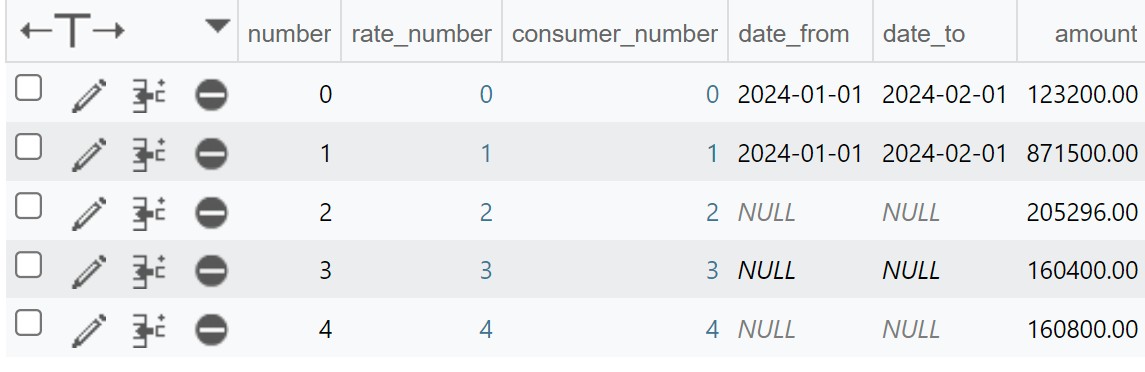


Рисунок 7 – Таблица ACCURALS

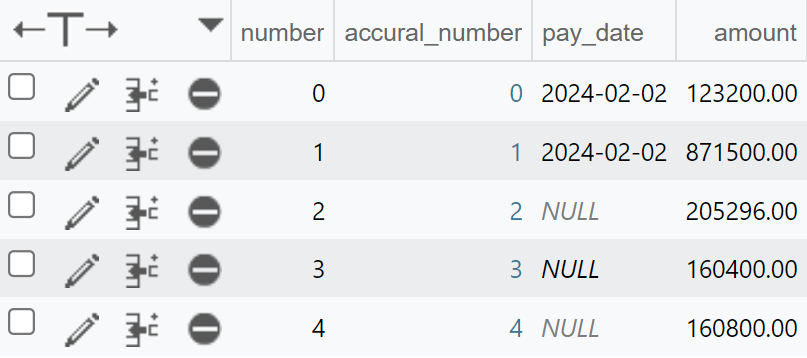


Рисунок 8 – Таблица PAYMENTS

**5.2. Тестирование базы данных на соблюдение ссылочной целостности**

Были созданы запросы на изменение и удаление данных, которые могут привести к нарушениям ссылочной целостности. Также было сохранено состояние таблиц до и после отправки запросов.

1. Изменить номер записи о владении квартирой с 5 на 6 (рисунки 9-10).

UPDATE CONSUMER\_OWN\_FLAT SET number = 6 WHERE number = 5;

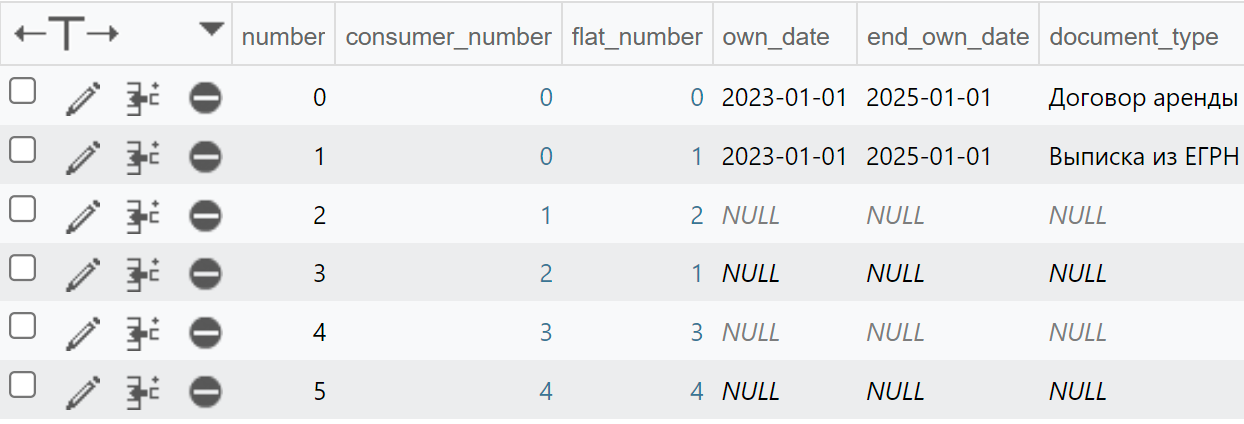


Рисунок 9 – Таблица CONSUMER\_OWN\_FLAT до первого изменения

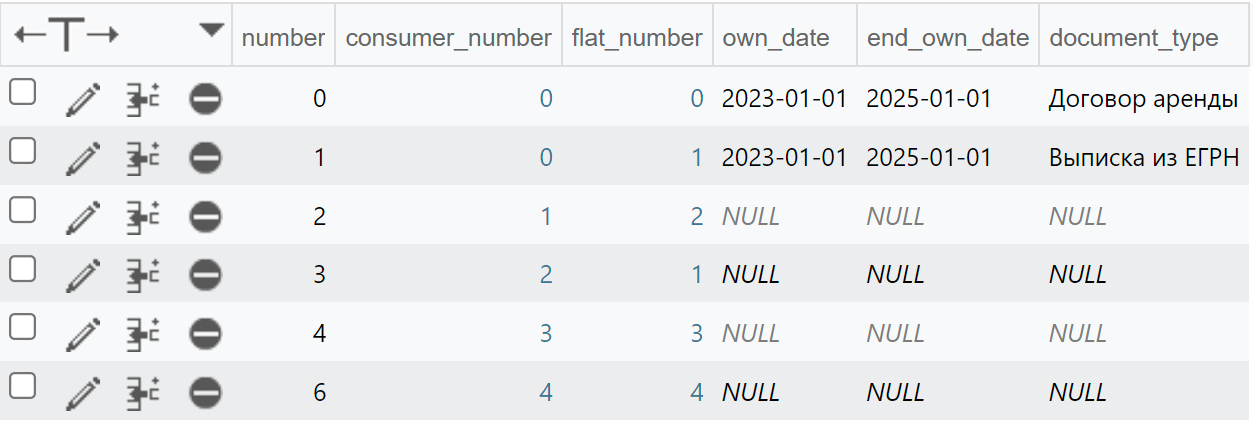


Рисунок 10 – Таблица CONSUMER\_OWN\_FLAT после первого изменения

В данном запросе изменяется первичный ключ. Так как записи с таким ключом в этой таблицы ещё не существовало – такой запрос возможно выполнить.

1. Изменить у записи о владении квартирой с номером 6 номер квартиры на 2 (рисунок 11).

UPDATE CONSUMER\_OWN\_FLAT SET flat\_number = 2 WHERE number = 6;

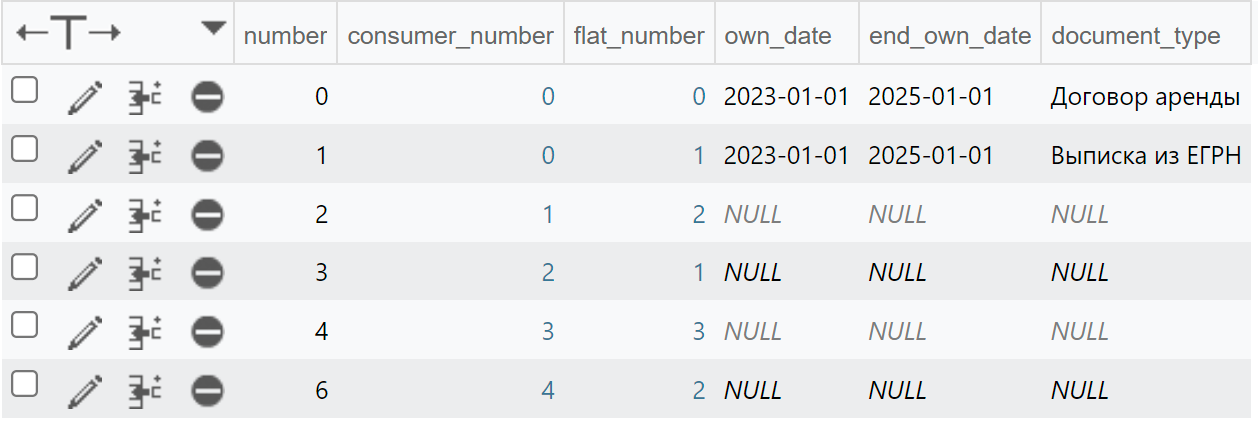


Рисунок 11 – Таблица CONSUMER\_OWN\_FLAT после второго изменения

В данном запросе изменяется внешний ключ. Так как таблица, с которой происходит работа, является зависимой, и запись с таким ключом в основной таблице существует – такой запрос возможно выполнить.

1. Изменить номер нулевой квартиры на 5 (рисунки 12-14).

UPDATE FLATS SET number = 5 WHERE number = 0;

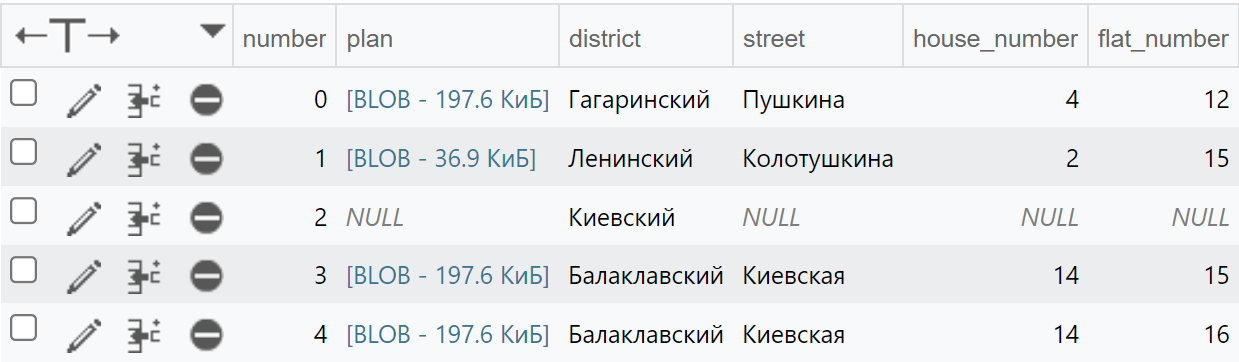


Рисунок 12 – Таблица FLATS до первого изменения

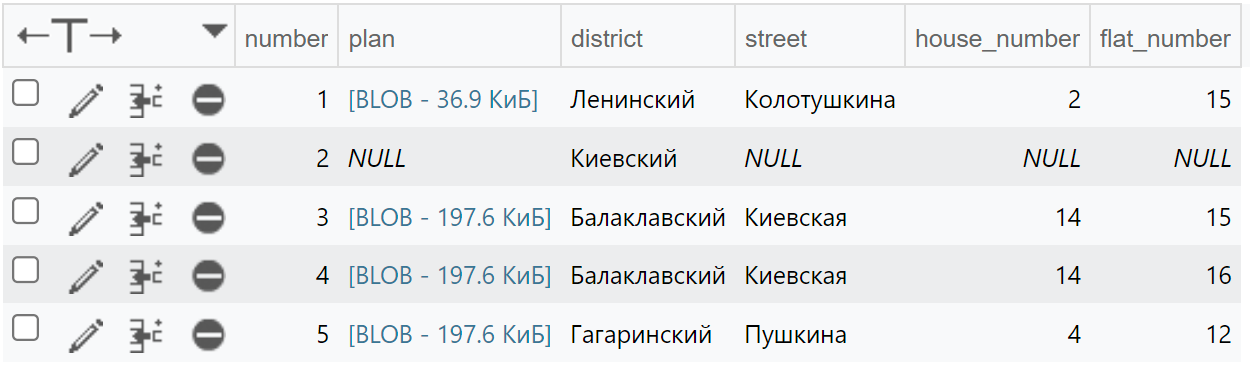


Рисунок 13 – Таблица FLATS после первого изменения

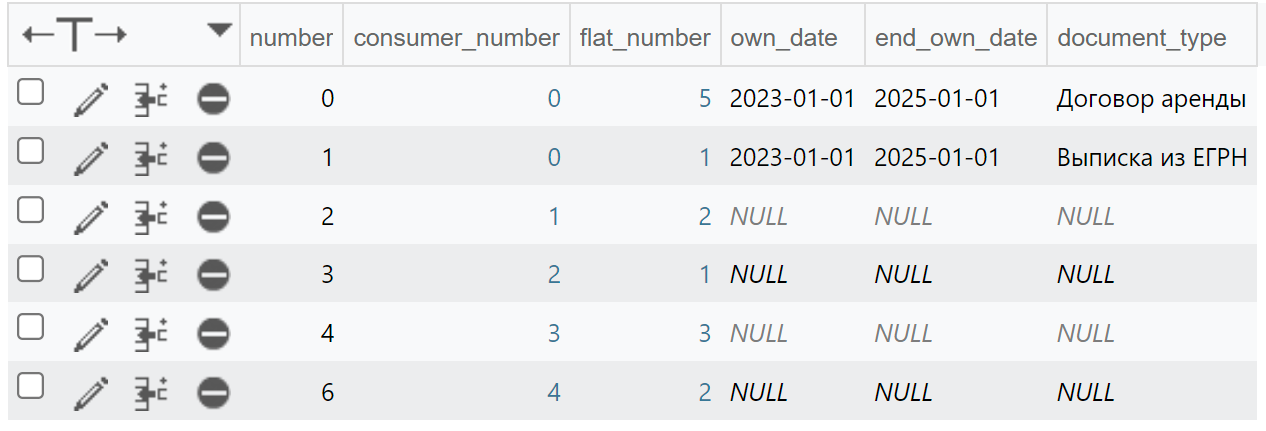


Рисунок 14 – Таблица CONSUMER\_OWN\_FLAT после первого изменения FLATS

В данном запросе изменяется первичный ключ. Так как записи с таким ключом в этой таблицы ещё не существовало – такой запрос возможно выполнить. Так как для дополнительной таблицы установлено каскадное изменения данных – внешний ключ в соответствующей записи дополнительной таблицы автоматически изменится на новый.

1. Изменить номер квартиры c 5 на 6 (рисунки 15-16).

UPDATE FLATS SET number = 6 WHERE number = 5;

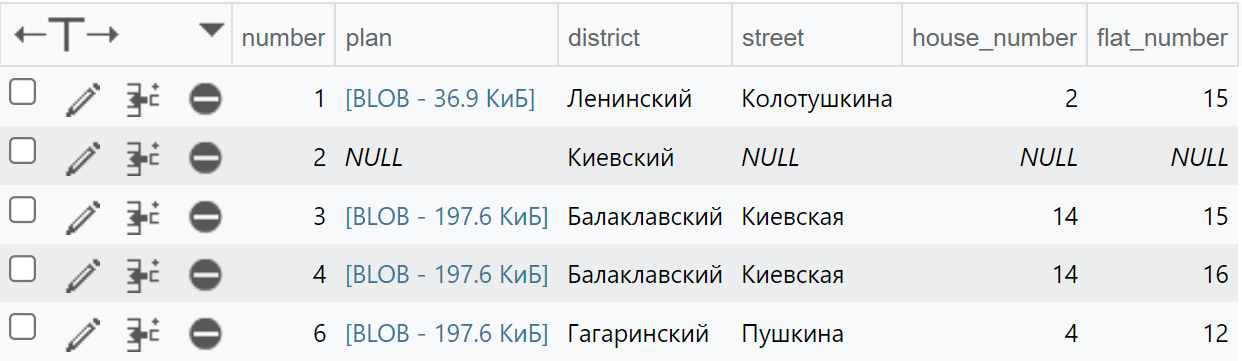


Рисунок 15 – Таблица FLATS после второго изменения

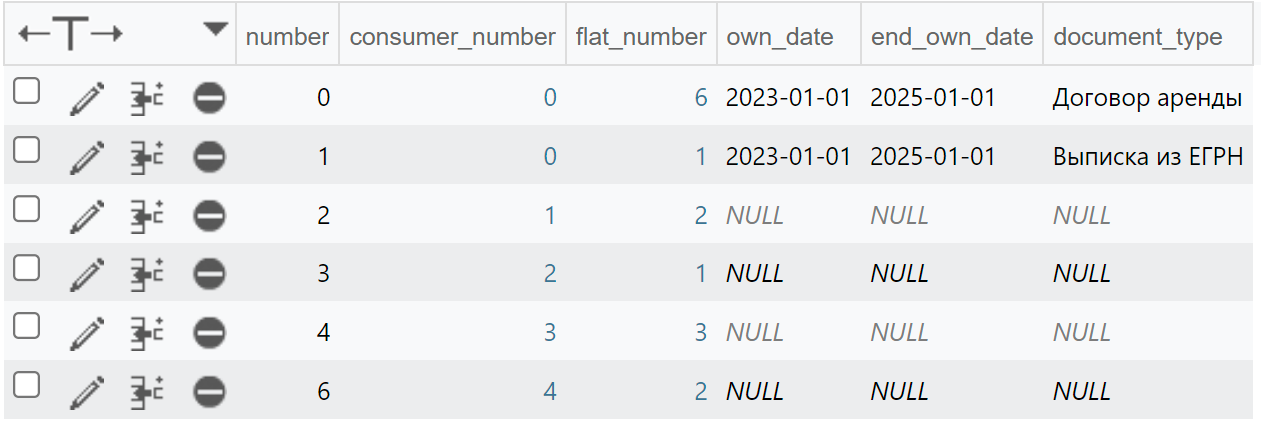


Рисунок 16 – Таблица CONSUMER\_OWN\_FLAT после второго изменения FLATS

В данном запросе изменяется первичный ключ. Так как записи с таким ключом в этой таблицы ещё не существовало – такой запрос возможно выполнить. Так как для дополнительной таблицы установлено каскадное изменения данных – внешний ключ в соответствующей записи дополнительной таблицы автоматически изменится на новый.

1. Удалить квартиру с номером 6 (рисунки 17-18).

DELETE FROM FLATS WHERE number = 6;

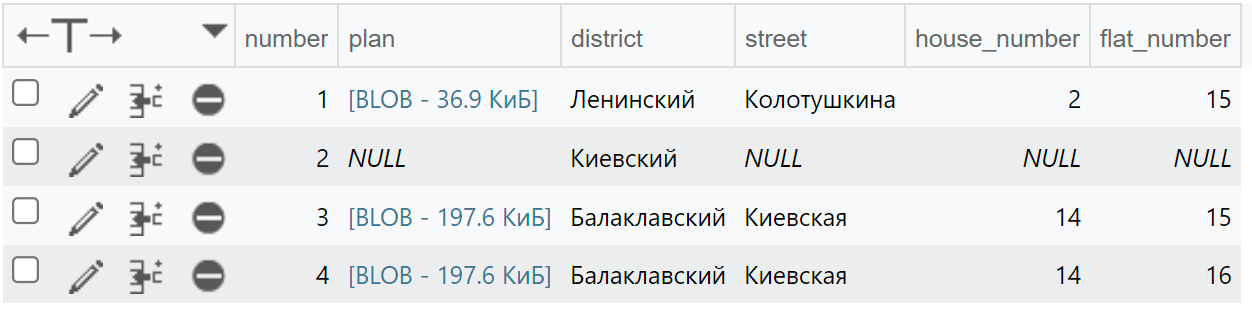


Рисунок 17 – Таблица FLATS после первого удаления

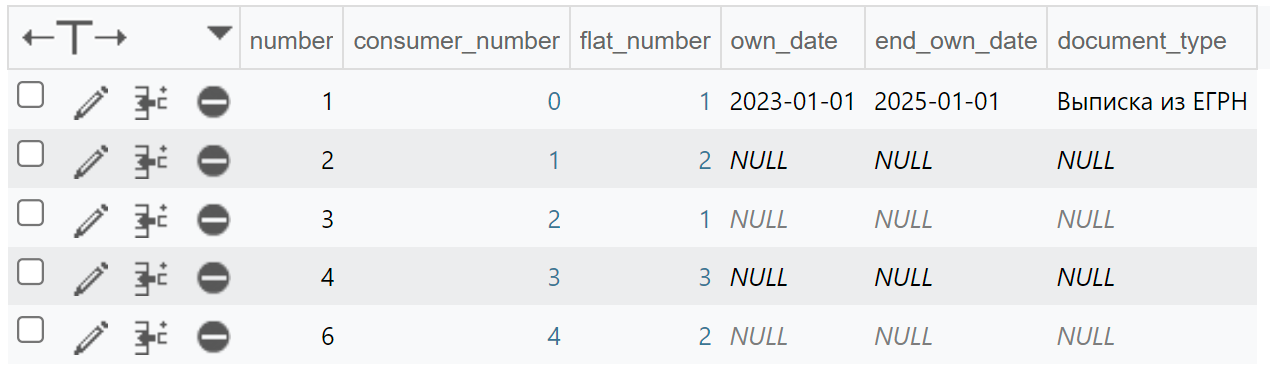


Рисунок 18 – Таблица CONSUMER\_OWN\_FLAT после первого удаления из FLATS

В этом запросе удаляется запись из основной таблицы. Так как для дополнительной таблицы установлено каскадное удаления данных – соответствующая запись в дополнительной таблице автоматически удалится.

1. Удалить запись о третьей квартире (рисунки 19-20).

DELETE FROM FLATS WHERE number = 3;

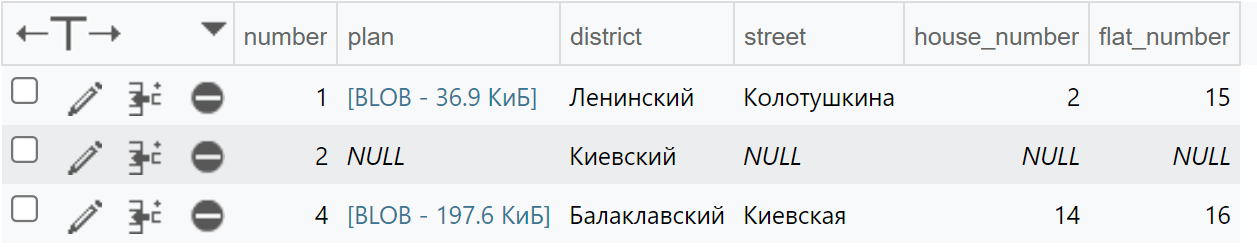


Рисунок 19 – Таблица FLATS после второго удаления

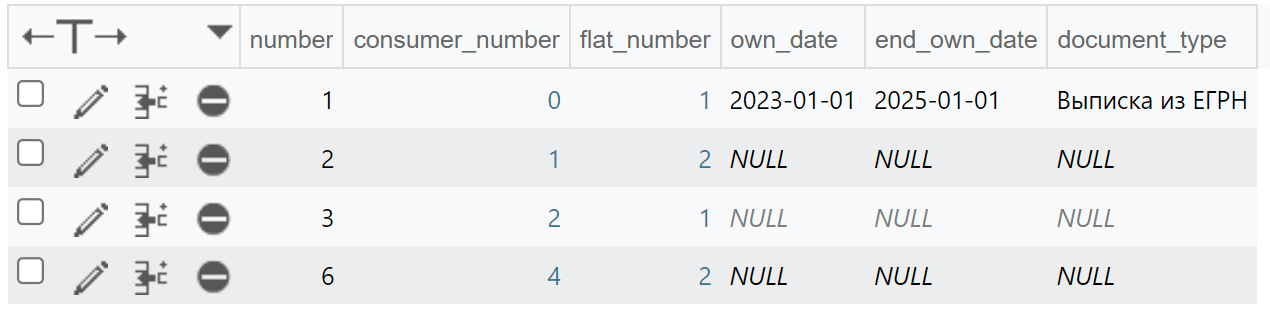


Рисунок 20 – Таблица CONSUMER\_OWN\_FLAT после второго удаления из FLATS

В этом запросе удаляется запись из основной таблицы. Так как для дополнительной таблицы установлено каскадное удаления данных – соответствующая запись в дополнительной таблице автоматически удалится.

1. Удалить шестую запись о владении квартирой (рисунок 21).

DELETE FROM CONSUMER\_OWN\_FLAT WHERE number = 6;

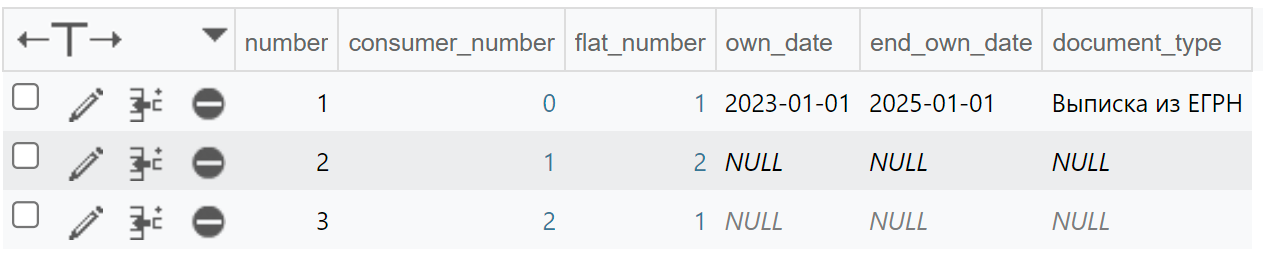


Рисунок 21 – Таблица CONSUMER\_OWN\_FLAT после первого удаления

В этом запросе удаляется запись из дополнительной таблицы. Так как таблица дополнительная – удаление данных из неё никак не затронет основную таблицу.

1. Удалить записи о владении квартирой потребителя с номером 2 (рисунок 22).

DELETE FROM CONSUMER\_OWN\_FLAT WHERE consumer\_number = 2;

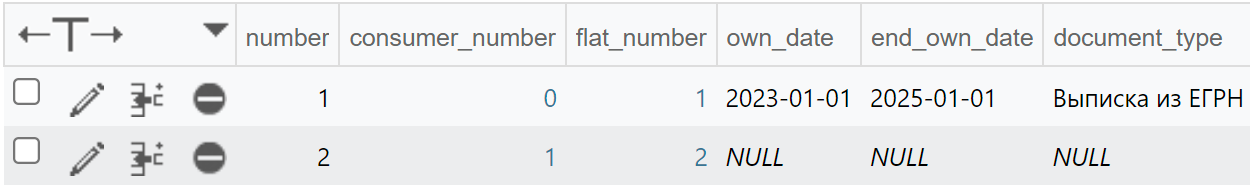


Рисунок 22 – Таблица CONSUMER\_OWN\_FLAT после второго удаления

В этом запросе удаляется запись из дополнительной таблицы. Так как таблица дополнительная – удаление данных из неё никак не затронет основную таблицу.

**Вывод**

В ходе практической работы было проведено тестирование базы данных. Были написаны запросы на удаление, добавление и изменение данных. Также были написаны запросы потенциально нарушающие ссылочную целостность базы данных.