<!--1. Создание таблицы "Владельцы" (Owners)

Таблица содержит идентификатор владельца, его имя и телефонный номер.

sql

CREATE TABLE Owners (

owner\_id INT(6) PRIMARY KEY, -- Идентификатор владельца

name VARCHAR(100), -- Имя владельца

phone VARCHAR(20) -- Телефон владельца

);

2. Создание таблицы "Станции метро" (MetroStations)

Таблица содержит идентификатор станции и её название.

sql

CREATE TABLE MetroStations (

station\_id INT(3) PRIMARY KEY, -- Идентификатор станции метро

name VARCHAR(100) -- Название станции метро

);

3. Создание таблицы "Квартиры" (Apartments)

Таблица содержит информацию о квартирах, включая идентификатор, владельца, площадь, количество

комнат и другие данные. Используются внешние ключи для связи с владельцами и станциями метро.

sql

CREATE TABLE Apartments (

apartment\_id INT(5) PRIMARY KEY, -- Идентификатор квартиры

owner\_id INT(6), -- Номер владельца (внешний ключ)

total\_area DECIMAL(4,1), -- Общая площадь

living\_area DECIMAL(4,1), -- Жилая площадь

room\_count INT(2), -- Количество комнат

kitchen\_size DECIMAL(3,1), -- Размер кухни

floor INT(2), -- Этаж

total\_floors INT(2), -- Всего этажей в доме

nearest\_station\_id INT(3), -- Ближайшая станция метро (внешний ключ)

price INT(7), -- Цена

address VARCHAR(30), -- Адрес квартиры

entry\_date DATE, -- Дата поступления

FOREIGN KEY (owner\_id) REFERENCES Owners(owner\_id), -- Связь с владельцем

FOREIGN KEY (nearest\_station\_id) REFERENCES MetroStations(station\_id) -- Связь с ближайшей

станцией метро

);

4. Вставка данных в таблицу "Владельцы" (Owners)

sql

INSERT INTO Owners (owner\_id, name, phone)

VALUES

(1, 'Иван Иванов', '123-456-7890'),

(2, 'Мария Петрова', '987-654-3210'),

(3, 'Петр Сидоров', '555-555-5555');

5. Вставка данных в таблицу "Станции метро" (MetroStations)

sql

INSERT INTO MetroStations (station\_id, name)

VALUES

(1, 'Краснопресненская'),

(2, 'Баррикадная'),

(3, 'Киевская'),

(4, 'Белорусская');

6. Вставка данных в таблицу "Квартиры" (Apartments)

sql

INSERT INTO Apartments (apartment\_id, owner\_id, total\_area, living\_area, room\_count, kitchen\_size,

floor, total\_floors, nearest\_station\_id, price, address, entry\_date)

VALUES

(1, 1, 45.5, 30.2, 2, 9.5, 3, 5, 1, 6000000, 'Улица Ленина, дом 5', '2024-11-01'),

(2, 2, 60.0, 45.0, 3, 12.0, 5, 9, 2, 8500000, 'Проспект Мира, дом 20', '2024-11-10'),

(3, 3, 75.0, 60.0, 4, 15.0, 7, 15, 3, 12000000, 'Площадь Победы, дом 3', '2024-11-15');

Пояснения:

Типы данных:

INT(x) — целочисленные значения с заданной длиной. Длина указывает на максимальное количество цифр,

которые можно хранить в поле.

DECIMAL(p,s) — числовое поле с плавающей точкой, где p — общая длина числа, а s — количество знаков

после запятой. Например, для площади квартиры это поле будет подходящим.

VARCHAR(x) — строка переменной длины, где x указывает максимальную длину строки.

DATE — тип данных для даты (в формате 'YYYY-MM-DD').

Внешние ключи:

В таблице Apartments используются внешние ключи для связи с таблицами Owners (владелец квартиры) и

MetroStations (ближайшая станция метро).

Констрейнты:

PRIMARY KEY — уникальный идентификатор строки в таблице.

FOREIGN KEY — внешний ключ для обеспечения целостности данных, который указывает на связь с другой

таблицей.

Эти SQL-запросы создадут базовую структуру для хранения данных о владельцах, квартирах и станциях

метро, а также заполнят таблицы примерными данными.-->