Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Рязанский государственный радиотехнический университет имени В.Ф. Уткина»

Кафедра «Электронных вычислительных машин»

К защите

Руководитель работы: Тарасова В.Ю.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

дата, подпись

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**К КУРСОВОЙ РАБОТЕ**

по дисциплине

**«Базы данных»**

на тему

«Разработка базы данных для предметной области:

«Бассейн»

|  |  |  |
| --- | --- | --- |
| Выполнил студент группы 945  Епишев М.С. | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  дата сдачи на проверку,  подпись |
| Руководитель работы  ассистент кафедры ЭВМ  Тарасова В.Ю. | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  оценка | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  дата защиты, подпись |

Рязань 2020

**Содержание**

[*Вариант задания.* 3](#_Toc63998777)

[1 Семантическое описание предметной области 3](#_Toc63998778)

[2 Разработка базы данных 4](#_Toc63998779)

[2.1 Разработка логической модели данных 4](#_Toc63998780)

[2.2 Определение типов хранимых данных 10](#_Toc63998781)

[2.3 Задание ограничений целостности данных 12](#_Toc63998782)

[2.4 Построение схемы БД 14](#_Toc63998783)

[2.5 Разработка сценария для создания БД и основных объектов структуры БД 14](#_Toc63998784)

[3. Разработка основных операций реляционной алгебры 17](#_Toc63998785)

[4 Разработка запросов к БД для реализации информационных потребностей пользователя 19](#_Toc63998786)

[Заключение 40](#_Toc63998787)

[Список используемой литературы 41](#_Toc63998788)

# *Вариант задания.*

База данных «Капитальный ремонт» должна хранить информацию о бригадах, выполняющих капитальный ремонт в домах. Требуется хранить данные:

1. О бригадах (Бригадир, контактный номер, количество рабочих, оклад за день работы).
2. Об управляющих компаниях (Название, представитель, год основания, адрес головного офиса, телефон).
3. О домах (Город, улица, номер дома, этажность, наличие лифта, управляющая компания, год постройки).
4. О капитальном ремонте домов бригадами с указанием даты начала капитального ремонта и предварительной даты окончания работ, вида работ и стоимости проведенных работ.

# 1 Семантическое описание предметной области

База данных — совокупность данных, хранимых в соответствии со схемой данных, манипулирование которыми выполняют в соответствии с правилами средств моделирования данных. Система управления базами данных (СУБД) – совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных. В данной курсовой работе используется СУБД MS SQL Server.

Капитальный ремонт многоквартирного дома — проведение работ по устранению неисправностей изношенных конструктивных элементов общего имущества собственников помещений в многоквартирном доме, в том числе по их восстановлению или замене, в целях улучшения эксплуатационных характеристик общего имущества в многоквартирном доме. С 2014 года собственники помещений ежемесячно обязаны уплачивать взносы на капитальный ремонт. Ответственность за проведение такого рода работ в доме несут управляющие компании.

Выполняют всю работу – бригады рабочих. О бригадах рабочих следует хранить следующую информацию: ФИО бригадира, его контактный номер, количество рабочих и оклад за день работы.

Каждая бригада рабочих должна быть закреплена за своей управляющей компанией. Об управляющих компаниях следует хранить информацию: название, ФИО представителя, год основания, адрес головного офиса, телефон.

О домах, в которых должен проходить капитальный ремонт следует хранить информацию: адрес, этажность, наличие лифта, управляющая компания, год постройки. Под адресом мы будем понимать город, улицу, на которой расположен дом и его номер.

О самом капитальном ремонте следует знать дату начала проведения работ, а также предварительную дату окончания ремонта, вид проводимой работы и стоимость проведенных работ.

# 2 Разработка базы данных

## 2.1 Разработка логической модели данных

На основании проведенного анализа предметной области «Капитальный ремонт» можно выделить следующие сущности:

Список сущностей предметной области представлен в таблице 1.

Таблица 1. Список сущностей

|  |  |  |
| --- | --- | --- |
| **№** | **Название** | **Назначение** |
| 1 | Бригада | Описание бригад, выполняющих ремонт в домах |
| 2 | Управляющая компания | Описание управляющих компаний, несущих ответственность за выполнение ремонтных работ в домах |
| 3 | Дома | Перечень домов, в которых проводится капитальный ремонт |

Продолжение таблицы 1

|  |  |  |
| --- | --- | --- |
| 4 | Выполняемые работы | Информация о текущих, либо запланированных работах |

Для каждой сущности необходимо привести описание ее атрибутов в виде таблицы. На данном этапе выделяются первичные, внешние ключи и не ключевые атрибуты. Результаты приведены в таблицах 2-5.

Для каждой сущности приведем описание ее атрибутов в виде таблицы. На данном этапе выделяются ключевые (это первичные, внешние и альтернативные ключи) и не ключевые атрибуты. Результаты приведены в таблицах 2-5.

Следует отметить, что для сущностей Бригада, в отличие от других сущностей, нет атрибута, с помощью которого можно идентифицировать конкретную бригаду. Поэтому требуется ввести дополнительный атрибут, например, ID бригады, который и будет являться первичным ключом.

Атрибут ID бригады является дополнительным служебным полем, добавленным к уже имеющимся информационным полям таблицы. Такие атрибуты называются суррогатными, их единственное предназначение - служить первичным ключом.

Также мы добавляем в сущности Дома и Выполняемые работы дополнительные атрибуты, которые после будут являться первичными ключами.

1. В сущность Дома – ID дома;

2. В сущность Выполняемые работы – ID дома и ID бригады.

Описание атрибутов сущности «Бригада» представлено в таблице 2.

Таблица 2. Список атрибутов сущности «Бригада»

|  |  |  |
| --- | --- | --- |
| **Ключевое поле** | **Название** | **Назначение** |
| PK(первичный ключ) | ID бригады | Ключевое поле. Представляет собой первичный ключ. Это уникальное значение, соответствующее каждой бригаде. Значения автоматически генерируется СУБД при вставке новой записи в таблицу. |
|  | Бригадир |  |
|  | Контактный номер |  |
|  | Кол-во рабочих |  |
|  | Оклад за день |  |

Описание атрибутов сущности «Управляющие компании» представлено в таблице 3.

Таблица 3. Список атрибутов сущности «Управляющие компании»

|  |  |  |
| --- | --- | --- |
| **Ключевое поле** | **Название** | **Назначение** |
| РК | Название | Ключевое поле. Представляет собой первичный ключ. Это уникальное значение, соответствующее каждой управляющей компании. |
|  | Представитель |  |
|  | Год основания |  |
|  | Адрес главного офиса |  |

Описание атрибутов сущности «Дома» представлено в таблице 4.

Таблица 4. Список атрибутов сущности «Дома»

|  |  |  |
| --- | --- | --- |
| **Ключевое поле** | **Название** | **Назначение** |
| PK | ID дома | Ключевое поле. Представляет собой первичный ключ. Это уникальное значение, соответствующее каждому дому. Значения автоматически генерируется СУБД при вставке новой записи в таблицу. |
|  | Город |  |
|  | Улица |  |
|  | Номер дома |  |
|  | Наличие лифта |  |
| FK | Управляющая компания | С помощью данного внешнего ключа можно узнать информацию об управляющей компании |
|  | Год постройки |  |

Описание атрибутов сущности «Выполняемые работы» представлен в таблице 4. Исходя из предметной области, в одном доме не могут проходить несколько одинаковых видов работ. Поэтому в таблице имеет место составной первичный ключ: вид работы, номер бригады и ID дома.

Описание атрибутов сущности «Выполняемые работы» представлено в таблице 5.

Таблица 5. Список атрибутов сущности «Выполняемые работы»

|  |  |  |
| --- | --- | --- |
| **Ключевое поле** | **Название** | **Назначение** |
| PK, FK | ID дома | Является часть составного первичного ключа. С помощью данного внешнего ключа можно узнать информацию о доме, в котором будут проводиться работы |
| РК, FK | ID бригады | Является частью составного первичного ключа. С помощью данного внешнего ключа можно узнать информацию о бригаде, которая будет участвовать в ремонте дома |
|  | Дата начала работ |  |
|  | Предварительная дата окончания работ |  |
| PK | Вид работы | Является частью составного первичного ключа |
|  | Стоимость проведенных работ |  |

**Построение логической модели**

На основании семантического описания предметной области и списка атрибутов из таблиц 2-5 опишем классы объектов (сущностей) и их свойства, расставим существующие связи между ними и приведем обоснование типов связей (1:1, 1:N, N:N).

Результат требуется представить в виде таблицы Список связей с описанием характеристик связей (таблица 6) и логической модели в графическом виде, где будут изображены все таблицы и связи между ними (рисунок 1)

Таблица 6. Список связей

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Сущности, участвующие в связи** | **Тип связи** | **Обоснование** |
| 1 | Бригада – Выполняемые работы | 1:N | Одна бригада может выполнять несколько работ в разных домах. Но конкретная работа, должна быть закреплена за одной бригадой. |
| 2 | Управляющая компания - Дома | 1:N | За одной управляющей компанией закреплено несколько домов. Но у каждого дома может быть только одна управляющая компания . |
| 3 | Дома – Выполняемые работы | 1:N | В одном доме могут проходить сразу несколько видов ремонта. Но запись о конкретном ремонте может относиться только к одному дому |

Логическая модель в графическом виде, где изображены все таблицы, и связи между ними представлена на схеме 1.

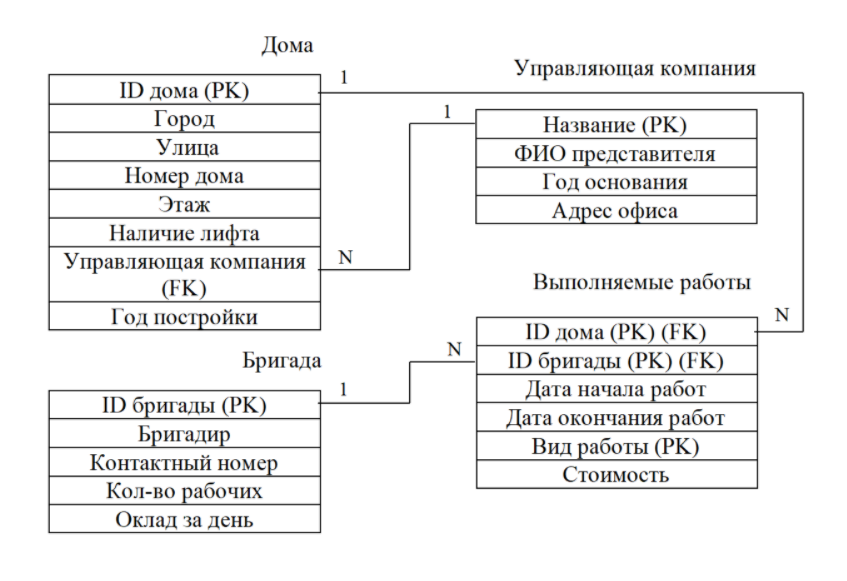


Схема 1. Логическая модель предметной области «Капитальный ремонт»

## 2.2 Определение типов хранимых данных

Для каждой таблицы приведем список всех атрибутов с указанием типа данных. При выборе типов данных следует учитывать, что далее для реализации будет использоваться СУБД Microsoft SQL Server. Результаты приведены в таблицах 7-10.

Таблица 7. Список атрибутов таблицы «Бригада»

|  |  |
| --- | --- |
| **Название** | **Тип данных** |
| ID бригады | int |
| Бригадир | nchar(40) |
| Контактный номер | varchar(20) |
| Кол-во рабочих | int |
| Оклад за день | money |

Таблица 8. Список атрибутов таблицы «Управляющая компании»

|  |  |
| --- | --- |
| **Название** | **Тип данных** |
| Название | nchar(20) |
| ФИО представителя | nchar(40) |
| Год основания | int |
| Адрес | nchar(40) |

Таблица 9. Список атрибутов таблицы «Дома»

|  |  |
| --- | --- |
| **Название** | **Тип данных** |
| ID дома | int |
| Город | nchar(20) |
| Улица | nchar(20) |
| Номер дома | nchar(7) |
| Этаж | int |
| Наличие лифта | nchar(3) |
| Управляющая компания | nchar(20) |
| Год постройки | int |

Таблица 10. Список атрибутов таблицы «Выполняемые работы»

|  |  |
| --- | --- |
| **Название** | **Тип данных** |
| ID дома | int |
| ID бригады | int |
| Дата начала работ | date |
| Дата окончания работ | date |
| Вид работ | nchar(30) |
| Стоимость | money |

## 2.3 Задание ограничений целостности данных

Для каждой таблицы необходимо выделить ограничения, накладываемые на значения атрибутов. Это могут быть ограничения на значения, значения по умолчанию, запрет/разрешение использования NULL-значений, поддержка уникальности значений и др. Результаты представлены в таблицах 11-14.

Таблица 11. Ограничения таблицы «Бригада»

|  |  |
| --- | --- |
| **Название** | **Ограничение** |
| ID бригады | Первичный ключ. Это уникальное значение, не может принимать NULL значения. Значения автоматически генерируются СУБД, начиная от 1 с шагом 1. |
| Бригадир | Должен содержать только русские буквы. Не может принимать NULL значения |
| Контактный номер | +7 (NNN) NNN-NN-NN,  где N – цифра от 0 до 9. |
| Кол-во рабочих | Должен принимать только положительные значения. Не может принимать NULL значения |
| Оклад за день | Больше нуля |

Таблица 12. Ограничения таблицы «Управляющие компании»

|  |  |
| --- | --- |
| **Название** | **Ограничения** |
| Название | Первичный ключ. Это уникальное значение, не может принимать NULL значения. |
| ФИО представителя | Должен содержать только русские буквы. Не может принимать NULL значения |
| Год основания | Принимает значения от 1991 до 2020. Не может принимать NULL значения |
| Адрес главного офиса | Не может принимать NULL значения |

Таблица 13. Ограничения таблицы «Дома»

|  |  |
| --- | --- |
| **Название** | **Ограничение** |
| ID дома | Первичный ключ. Это уникальное значение, не может принимать NULL значения. Значения автоматически генерируются СУБД, начиная от 1 с шагом 1. |
| Город | Должен содержать только русские буквы. Не может принимать NULL значения |
| Улица | Должен содержать только русские буквы. Не может принимать NULL значения |
| Номер дома | Не может принимать NULL значения |
| Этажность | Должен принимать только положительные значения. Не может принимать NULL значения |
| Наличие лифта | Да, нет. Значение по умолчанию - да |
| Управляющая компания | Внешний ключ |
| Год постройки | Принимает значения от 1900 до 2020. Не может принимать NULL значения |

Таблица 14. Ограничения таблицы «Выполняемые работы»

|  |  |
| --- | --- |
| **Название** | **Ограничения** |
| ID дома | Атрибут, входящий в составной первичный ключ. Атрибут, входящий во внешний ключ. |
| ID бригады | Атрибут, входящий в составной первичный ключ. Атрибут, входящий во внешний ключ. |
| Дата начала работ | ДД-ММ-ГГГГ |
| Дата окончания работ | ДД-ММ-ГГГГ |
| Вид работ | Атрибут, входящий во внешний ключ |

Продолжение таблицы 14

|  |  |
| --- | --- |
| Стоимость | Не может принимать NULL-значения. Должен принимать значения больше 1000 |

## 2.4 Построение схемы БД

Схема БД представляет собой набор заголовков таблиц. То есть на схеме БД указываются имена таблиц (сущностей), их атрибуты и связи между таблицами. В нашем случае схема связей представлена на рисунке.

Для уточнения схемы БД для каждой сущности (таблицы) указывают наименование атрибута, идентификатор, тип, длина, формат, ограничения. На рисунке 2 представлена схема базы данных «Капитальный ремонт».

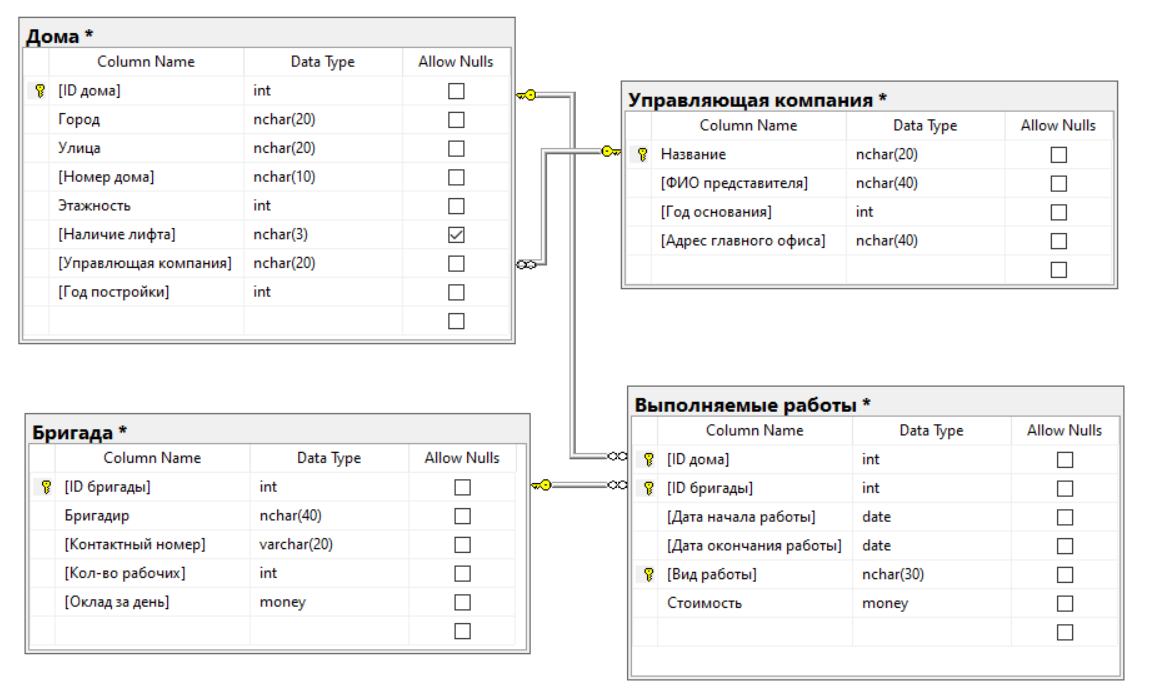


Рисунок 2 – Схема базы данных «Капитальный ремонт»

## 2.5 Разработка сценария для создания БД и основных объектов структуры БД

**--Создание файлов БД**

USE master

CREATE DATABASE [Капитальный ремонт\_945]

ON

(NAME = 'Капитальный ремонт',

FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL14.SQLEXPRESS\MSSQL\DATA\Капитальный ремонт.mdf',

SIZE = 1,

MAXSIZE = 10,

FILEGROWTH = 1)

LOG ON

(NAME = 'Капитальный ремонт\_log',

FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL14.SQLEXPRESS\MSSQL\DATA\Капитальный ремонт\_log.ldf',

SIZE = 1,

MAXSIZE = 5,

FILEGROWTH = 1)

GO

**--Создание таблиц**

GO

USE [Капитальный ремонт\_945]

GO

CREATE TABLE Бригада

([ID бригады] int not null identity (1, 1),

Бригадир nchar(40) not null check (Бригадир like '[А-Я][а-я]%'),

[Контактный номер] varchar(20) not null, check ([Контактный номер] like '+7 ([0-9][0-9][0-9]) [0-9][0-9][0-9]-[0-9][0-9]-[0-9][0-9]'),

[Кол-во рабочих] int not null check ([Кол-во рабочих] > 1 ),

[Оклад за день] money check ([Оклад за день] > 0))

GO

ALTER TABLE Бригада

ADD CONSTRAINT PK\_Бригада PRIMARY KEY ([ID бригады])

GO

CREATE TABLE [Управляющая компания]

(Название nchar(20) not null,

[ФИО представителя] nchar(40) not null check ([ФИО представителя] like '[А-Я][а-я]%'),

[Год основания] int not null check ([Год основания] between 1991 and 2020),

[Адрес главного офиса] nchar(40) not null)

GO

ALTER TABLE [Управляющая компания]

ADD CONSTRAINT PK\_УправКомпания PRIMARY KEY (Название)

GO

CREATE TABLE Дома

([ID дома] int not null identity (1, 1),

Город nchar(20) not null check (Город like '[А-Я][а-я]%'),

Улица nchar(20) not null check (Улица like '[А-Я][а-я]%'),

[Номер дома] nchar(10) not null,

Этажность int not null check (Этажность >0),

[Наличие лифта] nchar(3) not null check ([Наличие лифта] in ('да', 'нет')) default 'да',

[Управляющая компания] nchar(20),

[Год постройки ] int not null,check ([Год постройки ] between 1900 and 2020))

GO

ALTER TABLE Дома

ADD CONSTRAINT PK\_Дома PRIMARY KEY ([ID дома])

GO

ALTER TABLE Дома

ADD CONSTRAINT FK\_Название\_УправКомп foreign key ([Управляющая компания]) references [Управляющая компания](Название) ON DELETE CASCADE

GO

CREATE TABLE [Выполняемые работы]

([ID дома] int not null,

[ID бригады] int not null,

[Дата начала работы] date not null,

[Дата окончания работы] date not null ,

[Вид работы] nchar(30) not null,

Стоимость money check (Стоимость > 1000))

GO

ALTER TABLE [Выполняемые работы]

ADD CONSTRAINT PK\_ВыпРаботы PRIMARY KEY ([ID дома],[ID бригады],[Вид работы])

GO

ALTER TABLE [Выполняемые работы]

ADD CONSTRAINT FK\_IDдома foreign key ([ID дома]) references Дома ([ID дома]) ON DELETE CASCADE

GO

ALTER TABLE [Выполняемые работы]

ADD CONSTRAINT FK\_IDбригады foreign key ([ID бригады]) references Бригада ([ID бригады]) ON DELETE CASCADE

GO

**--Заполнение таблиц**

INSERT INTO Бригада VALUES

('Максимов Платон Всеволодович','+7 (905) 572-78-36',5,1500),

('Фролова Арина Богдановна','+7 (915) 362-17-69',4,2200 ),

('Зайцев Тимур Михайлович','+7 (995) 532-36-44',6, null),

('Барисова Юлия Дмитриевна', '+7 (979) 206-21-39',5,1500),

('Епишев Дмитрий Иванович', '+7 (949) 198-23-45',3, 3000),

('Николаев Евгений Артёмович','+7 (905) 232-98-40',10,2000),

('Масликов Павел Евгеньевич','+7 (920) 456-32-12',8,null),

('Баранчикова Евгения Андреевна','+7 (910) 111-43-21',4,4000),

('Путин Владимир Владимирович','+7 (936) 345-95-15',9,2000),

('Суслов Максим Павлович','+7 (919) 965-15-00',5,1400)

GO

INSERT INTO [Управляющая компания] VALUES

('Мой выбор','Волков Сергей Владимирович',1992,'​г.Рязань, ул.Островского, д.44'),

('Сияние','Овальный Алексей Максимович',2000,'​г.Москва, ​ул.Дзержинского, д.78'),

('Контур','Обцешко Руслан Викторович',2005, '​г.Москва, ​ул.Ленина, д.2'),

('Единство', 'Калинкевич Леонид Сергеевич',2010,'​г.Рязань, ул.Пушкина, д.17'),

('Северная компания', 'Безруков Никита Сергеевич',1995,'​г.Москва, ​ул.Лермонтова, д.23')

GO

INSERT INTO Дома VALUES

('Москва','Строителей','6',9,default,'Единство',1986),

('Москва','Высоковольтная','35',20,default,'Единство',2003),

('Москва','Театральная','11',5,'нет','Единство',1980),

('Саратов','Солдатская','64',12,default,'Северная компания',1989),

('Химки','Солнечная','2',5,'нет','Сияние',1975),

('Рязань','Октябрьская','60',4,'нет','Сияние',1970),

('Рязань','Космонавтов','30 к3',15,default,'Северная компания',2000),

('Орел','Партизанская','122',3,'нет','Мой выбор',1953),

('Рязань','Земельная','10',10,default,'Мой выбор',1990),

('Москва','Новая','4',9,default,'Единство',1988),

('Саратов','Старая','1',19,default,'Контур',2004),

('Орел','Пианистов','15',6,'нет','Мой выбор',2010),

('Рязань','Чикистов','90',10,default,'Северная компания',2012),

('Саратов','Ковровая','13 ',25,default,'Мой выбор',2018),

('Химки','Павлова','47',16,default,'Северная компания',2007),

('Химки','Пекинская','13',16,default,'Сияние',2001),

('Рязань','Бирюзова','88',3,'нет','Контур',1930),

('Рязань','Радистов','100',9,default,'Северная компания',1973),

('Орел','Каширина','4',18,default,'Мой выбор',2015),

('Орел','Мебельная','136',5,'нет','Единство',1994)

GO

INSERT INTO Выполняемые работы VALUES

(1,2,'2000-12-04','2001-03-04','Ремонт лифтового оборудования',100000),

(3,4,'2016-05-01','2016-08-03','Ремонт крыши',200000),

(4,1,'2018-02-15','2018-05-15','Ремонт лифтового оборудования',55000),

(6,7,'2013-03-01','2013-07-01','Ремонт подвальных помещений',70000),

(7,3,'2019-09-01','2019-12-16','Ремонт водоснабжения',100000),

(8,5,'2012-02-01','2012-07-03','Ремонт крыши',300000),

(10,2,'2015-10-01','2016-02-03','Ремонт лифтового оборудования',150000),

(11,6,'2018-05-16','2018-10-11','Ремонт газоснабжения',230000),

(12,5,'2011-10-01','2011-12-01','Ремонт крыши',25000),

(13,3,'2020-05-04','2020-11-04','Ремонт водоснабжения',55000),

(15,7,'2013-06-10','2013-11-10','Ремонт подвальных помещений',100000),

(16,8,'2003-02-20','2003-03-20','Ремонт лифтового оборудования',900000),

(17,9,'2008-06-01','2009-02-01','Ремонт фасада',500000),

(18,3,'1996-06-01','1996-10-01','Ремонт водоснабжения',350000),

(20,10,'2017-03-03','2017-06-06','Ремонт фасада',1500000)

GO

# 3. Разработка основных операций реляционной алгебры

1. Вывести полную информацию о бригадах.

R=Бригада

1. Вывести бригадиров и их контактный номер из таблицы «Бригада».

R=Бригада [Бригадир, Контактный номер]

1. Вывести полную информацию о бригадах, где имя бригадира Евгений.

R=Бригада WHERE Бригадир = ‘Евгений’

1. Вывести полную информацию о бригадах, в которых количество рабочих больше 4.

R=Бригада WHERE Количество рабочих > 4

1. Вывести вид работ у бригад с ID 1-3.

R1= (Выполняемые работы JOIN Бригада) R2=(R1 WHERE ID Бригады BETWEEN 1 and 3)[Вид работ]

1. Вывести бригады, не имеющих оклада за день.

R= (Бригада WHERE [оклад за день] = NULL) [ID бригады]

1. Вывести информацию о домах, которые находятся не в Москве R1=Дома WHERE Город<> ‘Москва’
2. Вывести номера домов, которые находятся только в Рязани.

R1= Дома WHERE город = ‘Рязань’ [Номер дома]

R2= Дома WHERE город <> ‘Рязань’ [Номер дома]

R3=R1 EXCEPT R2

1. Получить номера домов, в которых ведутся работы по ремонту лифтов. R1=(Выполняемые работы WHERE [Вид работы]= ремонт лифтов)[ID Дома] R2=R1 JOIN Номер дома[ID дома]
2. Вывести ID бригады, где кол-во рабочих больше 5, а оклад за день превышает 4000.

R1= Бригада WHERE [Количество рабочих] > 5

R2= Бригада WHERE [Оклад за день]> 4000

R3= (R1 INTERSECT R2) [ID бригады]

1. Вывести информацию об управляющих компаниях с годом основания 1995 и 2000. R1= (Управляющая компании WHERE [Год основания] =1995) R2= (Управляющая компании WHERE [Год основания] =2000) R3= R1 UNION R2
2. Получить информацию о бригадах в следующем виде:

|  |  |  |  |
| --- | --- | --- | --- |
| Вид работы | Бригадир | Контактный номер | Количество рабочих |

R = (Выполняемые работы JOIN Бригада) [Вид работы, Бригадир, Контактный номер, Количество рабочих]

1. Вывести всю информацию об Управляющих компаниях, названия которых начинаются с «C».

R= DISTINCT(Управляющая компания WHERE Название LIKE (‘C%’))

1. Вывести информацию об Управляющих компания, у которых год основания был до 2000

R=Управляющая компания WHERE Год основания<=2000

1. Вывести информацию о представителях управляющих компаний, осуществляющих работу в Москве клиентах, исключив из списка Волкова Сергея Владимировича.

R1 = (Дома WHERE Город = ‘Москва’)[Город]

R2 = R1 JOIN Управляющая компания

R3 = Управляющая компания WHERE ФИО = ‘Волков Сергей Владимирович’

R4 = R2 EXCEPT R3

**4 Разработка запросов к БД для реализации информационных потребностей пользователя**

Для созданной базы данных разработать на языке Transact-SQL следующие типы запросов:

Запросы представлены на рисунках 3 – 50.

1. Запросы с выбором строк по условию:
2. Запрос с использованием операторов сравнения;

Вывести ID бригады, оклад за день в которых больше 1500

Вывести наименования услуг, цена которых больше 2600.

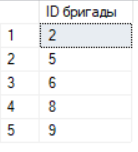


Рисунок 3 – Результат запроса 1.1

1. Запрос с использованием логических операторов AND, OR и NOT;

Вывести бригадиров, которые имеют 4 или 5 рабочих.

select Бригадир

from Бригада

where [Кол-во рабочих] = 4 or [Кол-во рабочих] =5

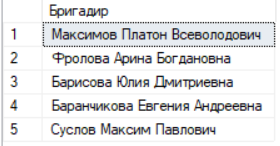


Рисунок 4 – Результат запроса 1.2

1. Запрос на использование комбинации логических операторов;

Получить полную информацию о бригадирах, с кодом оператора 915 или не 920.

select Бригадир

from Бригада

where [Контактный номер] LIKE ('%915%') OR NOT [Контактный номер] LIKE ('%920%')

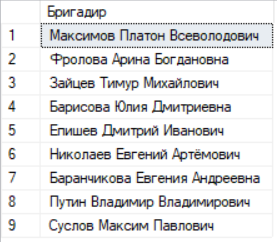


Рисунок 5 – Результат запроса 1.3

1. Запрос на использование выражений над столбцами;

Вывести вид работы и ее стоимость в долларах.

select [Вид работы], ROUND (Стоимость/75,1) AS [Стоимость в Долларах]

from [Выполняемые работы]

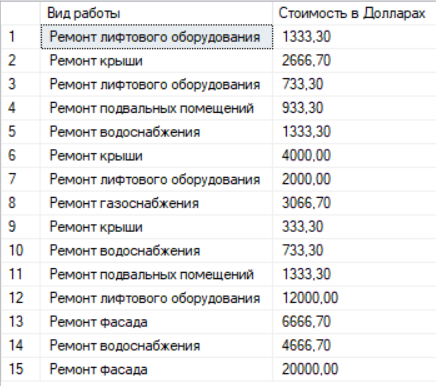


Рисунок 6 – Результат запроса 1.4

1. Запрос с проверкой на принадлежность множеству;

Вывести полную информацию о бригадах, чей ID равен 3,4,5.

select \*

from Бригада

where [ID бригады] IN (3,4,5)

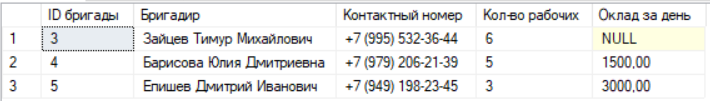


Рисунок 7 – Результат запроса 1.5

1. Запрос с проверкой на принадлежность диапазону значений;

Вывести город, улицу и номер дом этажность которых 10 – 20.

select Город, Улица, [Номер дома]

from Дома

where Этажность between 10 and 20



Рисунок 8 – Результат запроса 1.6

1. Запрос с проверкой на соответствие шаблону;

Вывести ФИО представителя и название компании, чья фамилия начинается на букву «O».

select [ФИО представителя], Название

from [Управляющая компания]

where [ФИО представителя] like 'О%'

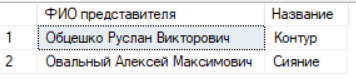


Рисунок 9 – Результат запроса 1.7

1. Запрос с проверкой на неопределенное значение.

Вывести ID бригад, у которых нет оклада за день.

select [ID бригады]

from Бригада

where [Оклад за день] is null

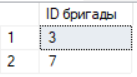


Рисунок 10 – Результат запроса 1.8

1. Запросы на теоретико-множественные реляционные операции:

2.1 Запрос на объединение таблиц;

Вывести список городов и улиц, названия которых начинаются на “П” или ”К”

select Город, Улица

from Дома

where Улица LIKE 'П%'

union

select Город, Улица

from Дома

where Улица LIKE 'К%'

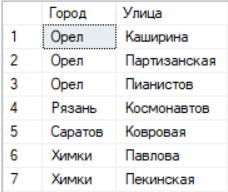


Рисунок 11 – Результат запроса 2.1

* 1. Запрос на пересечение таблиц;

Вывести информацию домах этажность которых от 10 до 20

SELECT \*

FROM Дома

WHERE Этажность >= 10

INTERSECT

SELECT \*

FROM Дома

WHERE Этажность <= 20

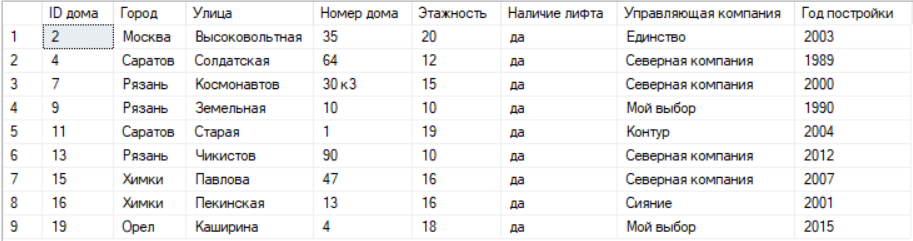


Рисунок 12 – Результат запроса 2.2

* 1. Запрос на вычитание таблиц;

Вывести ID домов, в которых не проводился капитальный ремонт.

SELECT [ID Дома] AS [Дома в которых не проводился кап ремонт]

FROM Дома

EXPERT

SELECT [ID Дома]

FROM [Выполняемые работы]

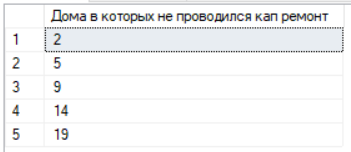


Рисунок 13 – Результат запроса 2.3

2.4 Запрос с использованием декартового произведения таблиц

Вывести бригадиров их вид работы в домах с ID меньше 10.

SELECT Бригада.Бригадир, [Выполняемые работы].[Вид работы]

FROM Бригада, [Выполняемые работы]

WHERE Бригада.[ID бригады]=[Выполняемые работы].[ID бригады] and [ID дома]<10

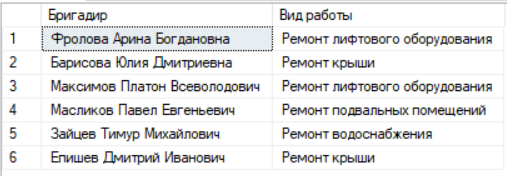


Рисунок 14 – Результат запроса 2.4

1. Многотабличные запросы:
2. Запрос с использованием соединения двух таблиц по равенству и условием отбора;

select Бригадир, [Вид работы], Стоимость

from [Выполняемые работы], Бригада

where [Выполняемые работы].[ID бригады]=Бригада.[ID бригады]

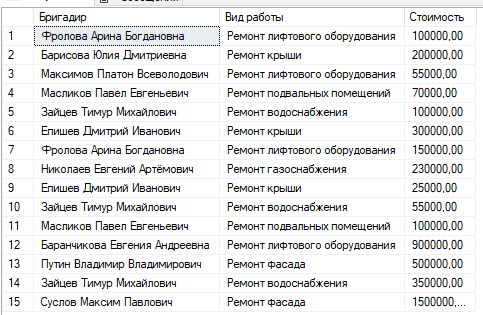


Рисунок 15 – Результат запроса 3.1

1. Запрос с использованием соединения трех таблиц по равенству и условием отбора;

select Бригадир, Город, [Дата начала работы], [Дата окончания работы]

from [Выполняемые работы] as v join Бригада as b on v.[ID бригады]=b.[ID бригады]

join Дома as d on v.[ID дома]=d.[ID дома]

where Стоимость>55000.00

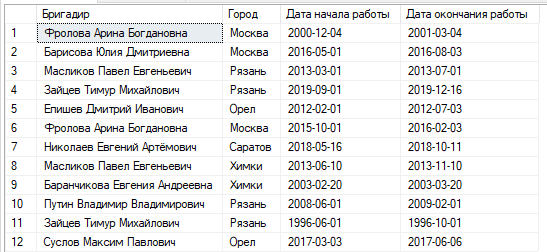


Рисунок 16 – Результат запроса 3.2

1. Запрос с использованием соединения по четырем таблицам;

select Бригадир, Город, [Управляющая компания], [Дата начала работы], [Дата окончания работы]

from [Выполняемые работы] as v join Бригада as b on v.[ID бригады]=b.[ID бригады]

join Дома as d on v.[ID дома]=d.[ID дома]

join [Управляющая компания] on Название=[Управляющая компания]

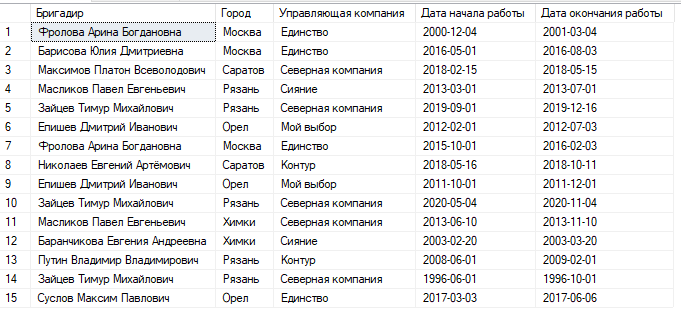


Рисунок 17 – Результат запроса 3.3

1. Запрос с использованием левого внешнего соединения;

select distinct Город, Улица, [Номер дома], Стоимость

from Дома left join [Выполняемые работы] on Дома.[ID дома]=[Выполняемые работы].[ID дома]



Рисунок 18 – Результат запроса 3.4

1. Запрос на использование правого внешнего соединения;

select distinct Город, Улица, [Номер дома], Стоимость

from Дома right join [Выполняемые работы] on Дома.[ID дома]=[Выполняемые работы].[ID дома]

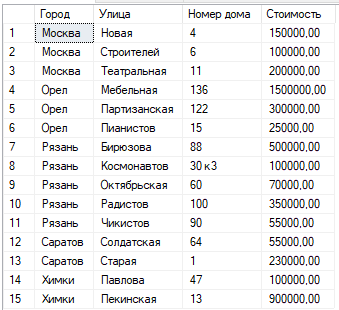


Рисунок 19 – Результат запроса 3.5

1. Запрос с использованием функции COUNT;

select [Управляющая компания], COUNT ([ID дома]) as [Кол-во домов у компании]

from Дома, [Управляющая компания]

where Название=[Управляющая компания]

group by [Управляющая компания]

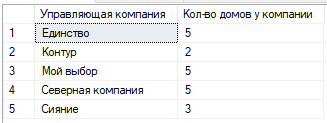


Рисунок 20 – Результат запроса 3.6

1. Запрос с использованием функции SUM;

select [Управляющая компания], sum ([ID дома]) as [Сумма ID домов компании]

from Дома, [Управляющая компания]

where Название=[Управляющая компания]

group by [Управляющая компания]

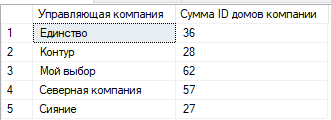


Рисунок 21 – Результат запроса 3.7

1. Три запроса с использованием временных функций
   * 1. Первый запрос

select \*

from [Выполняемые работы]

where month([Дата окончания работы]) < 05

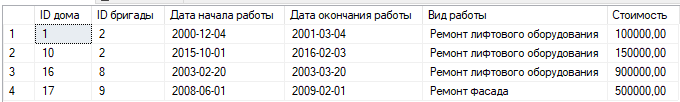


Рисунок 22 – Результат запроса 3.8.1

* + 1. Второй запрос

select \*

from [Выполняемые работы]

where Year([Дата начала работы])<2010

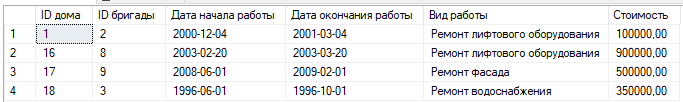


Рисунок 22 – Результат запроса 3.8.2

* + 1. Третий запрос

select \*

from [Выполняемые работы]

where DAY([Дата окончания работы])<10

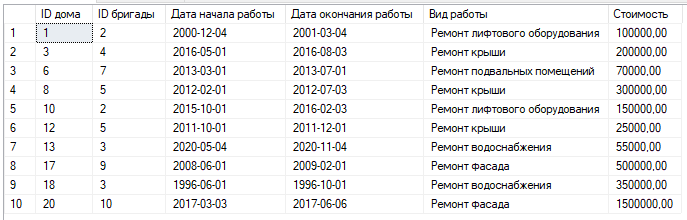


Рисунок 23 – Результат запроса 3.8.3

1. Запрос с использованием группировки по одному столбцу;

select Город, COUNT ([ID дома]) as [Кол-во домов ремонтируемых в городе]

from Дома

group by Город



Рисунок 24 – Результат запроса 3.9

1. Запрос на использование группировки по нескольким столбцам;

select Город,[Управляющая компания], COUNT ([ID дома]) as [Кол-во домов у компании]

from Дома

group by Город, [Управляющая компания]

  
Рисунок 25 – Результат запроса 3.10

1. Запрос с использованием условия отбора групп HAVING;

select Дома.[ID дома], SUM (Стоимость) As Траты

from Дома, [Выполняемые работы]

where Дома.[ID дома]=[Выполняемые работы].[ID дома]

group by [Дома].[ID дома]

having SUM(Стоимость)>=100000

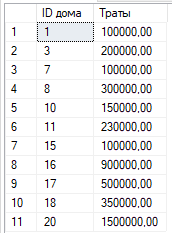


Рисунок 26 – Результат запроса 3.11

1. Запрос с использованием сортировки по столбцу;

select Дома.[ID дома], SUM (Стоимость) As Траты

from Дома, [Выполняемые работы]

where Дома.[ID дома]=[Выполняемые работы].[ID дома]

group by [Дома].[ID дома]

order by Траты DESC

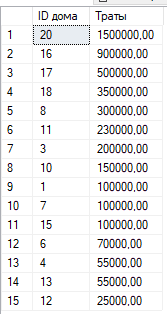


Рисунок 27 – Результат запроса 3.12

1. Запросы на модификацию таблиц:
2. Запрос на добавление новых данных в таблицу;

insert into [Управляющая компания]

values

('Альянс', 'Яшенков Дмитрий Игоревич', 2018, 'г.Рязань,ул.Нововведенская,д.135')



Рисунок 28 – Результат запроса 4.1

1. Запрос на добавление новых данных по результатам выполненного запроса;

insert into Бригада

values

('Тарасов Сергей Сергеевич','+7 (910) 456-00-82',7,null)



Рисунок 29 – Результат запроса 4.2

1. Запрос на обновление существующих данных в таблице;

update [Выполняемые работы]

set Стоимость=Стоимость+5000

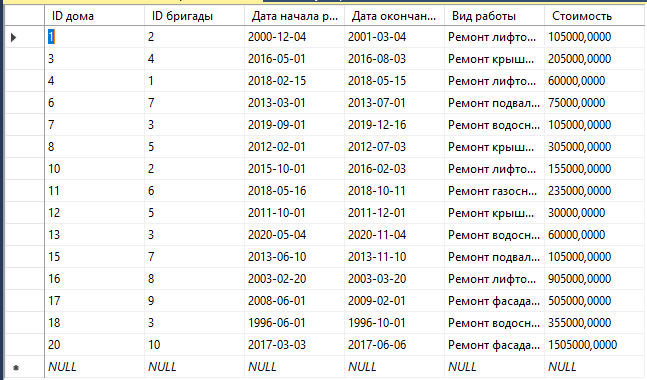


Рисунок 30 – Результат запроса 4.3

1. Запрос на обновление существующих данных по результатам подзапроса во фразе WHERE;

update [Выполняемые работы]

set Стоимость=(Стоимость)\*0.3

where [ID бригады]=(select [ID бригады]

From Бригада

where Бригадир='Суслов Максим Павлович')

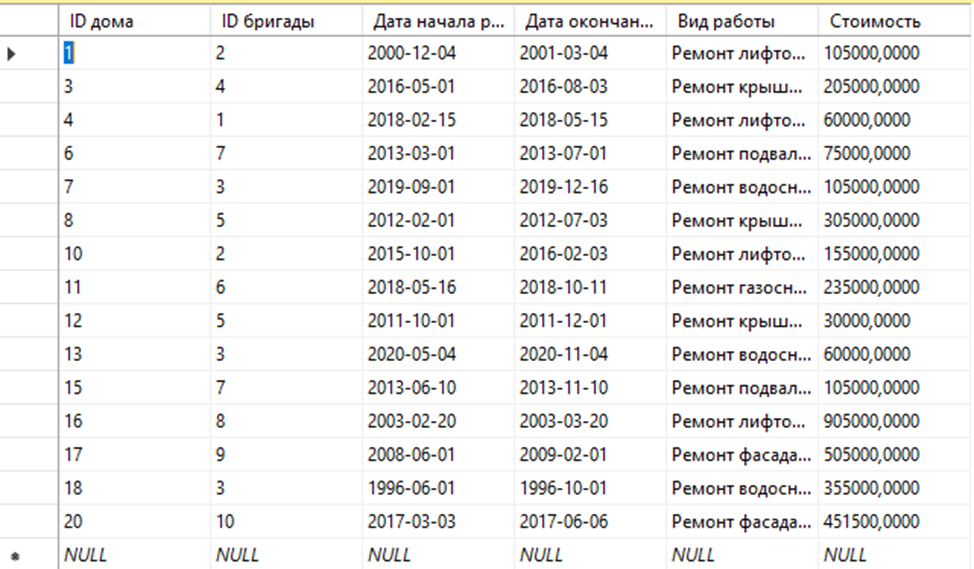


Рисунок 31 – Результат запроса 4.4

1. Запрос на удаление существующих данных

delete

from [Управляющая компания]

where [Год основания]>2015

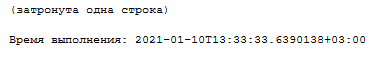


Рисунок 32 – Результат запроса 4.5

1. Запрос на удаление существующих данных из одной таблицы на основе связанных с ней таблиц.

delete [Выполняемые работы]

from [Выполняемые работы] JOIN Дома ON [Выполняемые работы].[ID дома]=Дома.[ID дома]

where Город LIKE 'Х%'



Рисунок 33 – Результат запроса 4.6

1. Запрос на удаление существующих данных по результатам подзапроса во фразе WHERE;

delete

from [Выполняемые работы]

where [ID бригады] =(select [ID бригады]

from Бригада

where Бригадир = 'Максимов%')

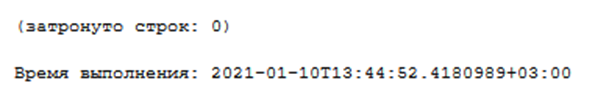


Рисунок 34 – Результат запроса 4.7

1. Многотабличные запросы с использованием подзапросов:
2. Запроса с использованием операций сравнения;

Вывести полную информацию о бригаде, выполняющей работу стоимостью выше 1300000.

select \*

from Бригада

where [ID бригады] = (select [ID бригады]

from [Выполняемые работы]

where Стоимость > 1300000)



Рисунок 36 – Результат запроса 5.1.

1. Запрос с использованием операции ANY;

Вывести города в которых проходит ремонт газоснабжения.

select Город

from Дома

where [ID дома] = Any (select [ID дома]

from [Выполняемые работы]

where [Вид работы]= 'Ремонт газоснабжения')



Рисунок 37 – Результат запроса 5.2

1. Запрос с использованием операции ALL;

Вывести полную информацию об управляющей компании, которая не закреплена ни за одним из домов.

select \*

from [Управляющая компания]

where Название != ALL (select [Управляющая компания]

from Дома )



Рисунок 38 – Результат запроса 5.3

1. Запрос с использованием операции IN;

Вывести дома, в которых не проходят ремонтные работы

select [Номер дома], Улица, Город

from Дома

where [ID дома] not in ( select [ID дома]

from [Выполняемые работы])

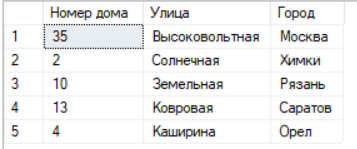


Рисунок 39 – Результат запроса 5.4

1. Запрос с использованием операции EXISTS;

Вывести название управляющей компании и ее представителя, которые осуществляют работу в домах

select Название, [ФИО представителя]

from [Управляющая компания]

where EXISTS

(select \*

from Дома

where [Управляющая компания].Название = Дома.[Управляющая компания])

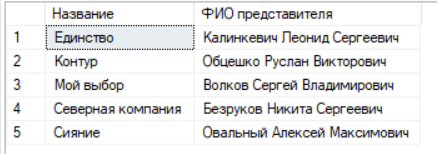


Рисунок 40– Результат запроса 5.5

1. Запрос с использованием двух вложенных друг в друга подзапросов;

Вывести полную информацию о бригаде, работающей на улице Новая

select\*

from Бригада

where [ID бригады] in (select [ID бригады]

from [Выполняемые работы]

where [ID дома] = (select [ID дома]

from Дома

where Улица = 'Новая'))



Рисунок 41 – Результат запроса 5.6

1. Запрос с использованием трех вложенных друг в друга подзапросов;

Вывести информацию об управляющей компании, бригада которой выполняет ремонт в домах под руководством Николаева Евгения Артёмовича.

select Название, [ФИО представителя], [Год основания]

from [Управляющая компания]

where Название in (select [Управляющая компания]

from Дома

where [ID дома] = (select [ID дома]

from [Выполняемые работы]

where [ID бригады]= (select [ID бригады]

from Бригада

where Бригадир= 'Николаев Евгений Артёмович')))



Рисунок 42 – Результат запроса 5.7

1. Запросы на изменение структуры существующей таблицы:
2. Запрос на добавление нового столбца к таблице;

Добавить к таблице «Выполняемые работы» столбец «Скидка»

alter table [Выполняемые работы]

add Скидка int

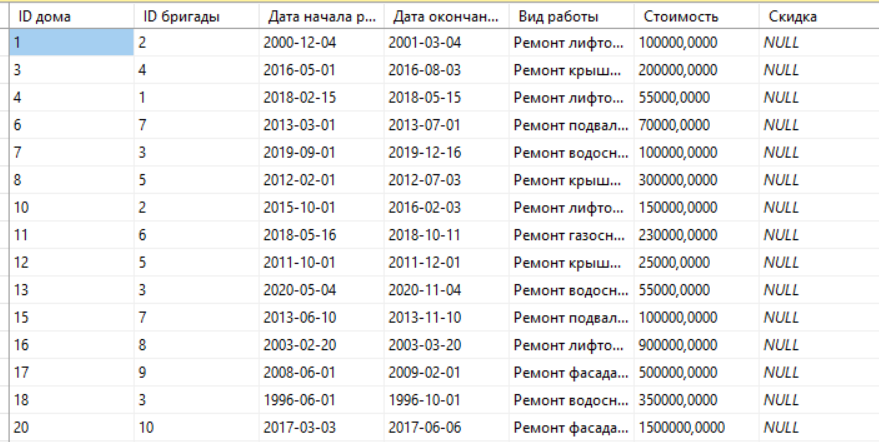


Рисунок 43 – Результат запроса 6.1

1. Запрос на добавление нового ограничения к таблице;

Добавить ограничение check к атрибуту «Скидка» в таблице «Выполняемые работы».

alter table [Выполняемые работы]

add constraint SKIDKA check (Скидка<10)

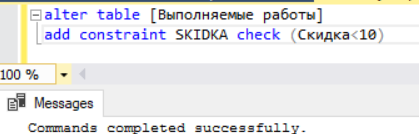


Рисунок 44 – Результат запроса 6.2

1. Запрос на удаление нового ограничения к таблице;

alter table [Выполняемые работы]

drop constraint SKIDKA

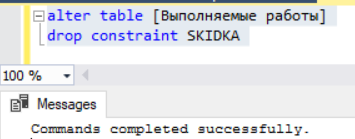


Рисунок 45 – Результат запроса 6.3

1. Запрос на удаление нового столбца к таблице;

alter table [Выполняемые работы]

drop column Скидка

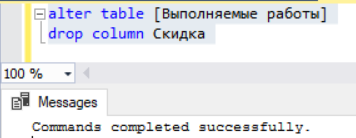


Рисунок 46 – Результат запроса 6.4

1. Запрос на изменение типа данных в таблице;

Изменить тип атрибута «Город» в таблице «Дома» с nchar (20) на nchar (15)

alter table Дома

alter column Город nchar(15)

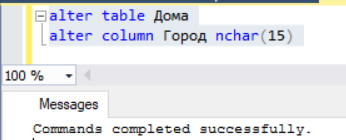


Рисунок 47 – Результат запроса 6.5

1. Запрос на добавление нового первичного ключа к таблице с сохранением целостности таблицы;

alter table [Выполняемые работы]

drop constraint FK\_IDбригады

alter table Бригада

drop constraint PK\_Бригада

alter table Бригада

add constraint PK\_IDбригады\_Бригадир primary key (Бригадир,[ID бригады])

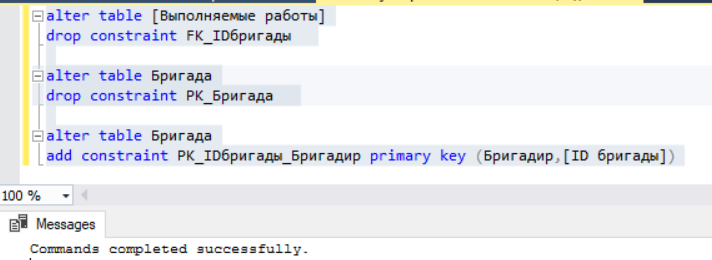


Рисунок 48 – Результат запроса 6.6

**Заключение**

В результате выполнения курсовой работы была создана база данных «Капитальный ремонт».

При создании таблиц были заданы ограничения (первичные ключи, внешние ключи, ограничения на использование NULL-значений и т.д.).

Были разработаны операции реляционной алгебры и запросы к базам данных для реализационных потребностей пользователя.

Для выполнения работы были использованы возможности Microsoft SQL Server 2017 (СУБД).

**Список используемой литературы**

1. Клайн К. SQL. Справочник / Пер. с англ. – М.: КУ-ДИЦ-ОБРАЗ, 2006 – 832 с.
2. Фиайли К. SQL: Пер. с англ. – М.: ДМК Пресс. – 456 с.: ил. (Серия «Quick Start»).
3. Дунаев В.В. Базы данных. Язык SQL. – СПб.: БХВ-Петербург, 2006. – 288 с.: ил.
4. Черноусова А.М. Создание и использование баз данных, - ГОУ ОГУ, 2009. – 244 с.
5. Раздаточный материал Гринченко Н.Н