МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ государственное БЮДЖЕТНОЕ

образовательное учреждение

высшего образования

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

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

Кафедра защиты информации

**

**ОТЧЁТ**

**по практической работе №2**

**«***Проектирование базы данных***»**

**по дисциплине: «***Технологии и методы программирования***»**

Выполнили:Проверил:

Студенты гр. «АБ-120» и «АБ-124»,

«АВТФ»

*Горбачев Николай Валерьевич*

*Садреев Андрей Олегович* *ассистент кафедры ЗИ*

*Щербинина Дарья Васильевна Медведев М.А.*

«» апреля 2023г«\_\_\_» \_\_\_\_\_\_ 2023 г.

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

(подпись) (подпись)

Новосибирск 2023

**Задачи-требования:**

* Спроектировать базу данных согласно заданному набору атрибутов и сущностей
* На основании проделанной работы необходимо создать отчет, содержащий в себе, все перечисленное выше.

**Описание проекта:**

Спроектировать и создать базу данных для предметной области: деятельность отдела учета нежилых помещений Бюро технической инвентаризации.

В 2000 г. в нашем городе была проведена сплошная инвентаризация, в ходе которой было выявлено 16 000 нежилых помещений. Это магазины (встроенные, пристроенные и отдельно стоящие), офисы, учреждения, мастерские и т.д. Помещение может состоять из отдельных частей (кабинет, проходная, коридор). Составных частей помещений выявлено 265000.

В одном здании может быть несколько помещений, а помещение может состоять из нескольких частей. Любое здание имеет уникальный кадастровый номер, однозначно определяющий его положение в городе.

**Концептуальное проектирование.**

Концептуальное проектирование - это процесс создания модели используемой информации, не зависящей от физических аспектов ее представления.

**Основные этапы:**

* Определение сущностей;

|  |  |
| --- | --- |
| **Сущность** | **Определение** |
| Владельцы | Сведения о владельцах |
| Местоположение | Сведения о адресе здания |
| Кадастр | Сведения о кадастровом номере здания |
| Здание | Сведения о здании |
| Помещение | Сведения о помещении |
| Части помещения | Сведения о части помещения |

* Определение связей между сущностями;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Сущность 1** | **Кратность** | **Тип связи** | **Кратность** | **Сущность 2** |
| Кадастр | 1..1 | Имеет | 1..1 | Владельцы |
| Кадастр | 1..1 | Имеет | 1..1 | Местоположение |
| Здание | 1..1 | Содержит | 1..\* | Помещение |
| Здание | 1..1 | Имеет | 1..1 | Кадастр |
| Помещение | 1..1 | Содержит | 1..\* | Часть помещения |

* Определение атрибутов сущностей;

*Владельцы*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Обязательно к заполнению** |
| Id Владельца | int | Простой | Натуральное число | Да |
| ФИО | Char(50) | Составной | Кириллица | Да |

*Местоположение*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Обязательно к заполнению** |
| Id Местоположения | int | Простой | Натуральное число | Да |
| Улица | Char(20) | Простой | Кириллица,  ‘-‘ | Да |
| Номер дома | Char(10) | Составной | Кириллица, ‘/’, ‘0..9’ | Да |

*Кадастр*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Обязательно к заполнению** |
| Id кадастровый номер | int | Простой | Натуральное число | Да |
| Id владельцы | int | Простой | Натуральное число | Да |
| Стоимость | decimal | Простой | Вещественное положительное число | Да |
| Этажность | int | Простой | Целое число | Да |
| Id Местоположение | int | Простой | Натуральное число | Да |

*Здание*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Обязательно к заполнению** |
| Id Здания | int | Простой | Натуральное число | Да |
| Id Кадастровый номер | int | Простой | Натуральное число | Да |
| Id Помещения | int | Простой | Натуральное число | Да |

*Помещение*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Обязательно к заполнению** |
| Id Помещения | int | Простой | Натуральное число | Да |
| Id Части помещения | int | Простой | Натуральное число | Да |
| Назначение | Char(50) | Простой | Кириллица | Да |
| Тип | Char(50) | Простой | Кириллица | Да |

*Часть помещения*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Обязательно к заполнению** |
| Id Части помещения | int | Простой | Натуральное число | Да |
| Значение | Char(20) | Простой | Кириллица | Да |
| Площадь | decimal | Простой | Вещественное положительное число | Да |

* Построение ER-диаграммы.

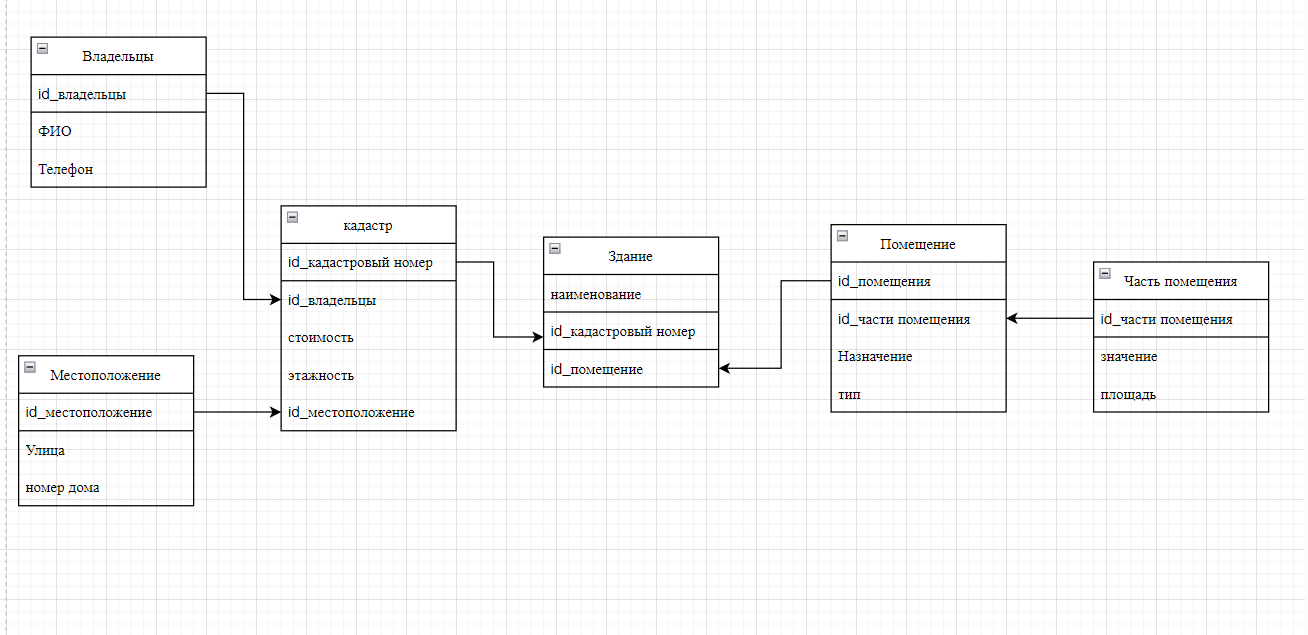


Рисунок 1 – ER-диаграмма концептуального проектирования

**Логическое проектирование**

Логическое проектирование - это конструирование информационной модели на основе существующих моделей данных без учета используемой СУБД и прочих физических условий реализации.

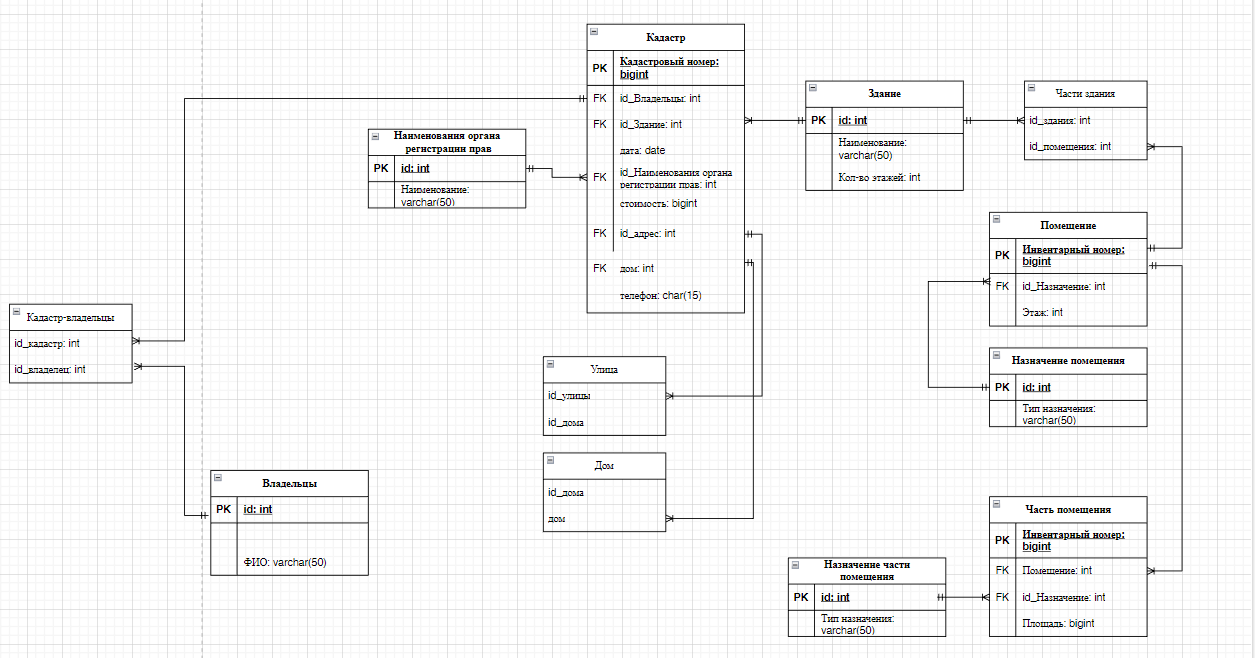


Рисунок 2 – ER-диаграмма логического проектирования

* Описание атрибутов

*Владельцы*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Ключи** | **Обязательно к заполнению** |
| Id Владелец | int | Простой | Натуральное число | PK | Да |
| ФИО | сhar(50) | Составной | Кириллица, ‘-‘ |  | Да |

*Кадастр-владельцы*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Ключи** | **Обязательно к заполнению** |
| id\_владелец | int | Простой | Натуральное число | FK | Да |
| id\_кадастр | int | Простой | Натуральное число | FK | Да |

*Наименование органа регистрации прав*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Ключи** | **Обязательно к заполнению** |
| id\_органа регистрации прав | int | Простой | Натуральное число | PK | Да |
| Наименование | char(50) | Простой | Кириллица |  | Да |

*Кадастр*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Ключи** | **Обязательно к заполнению** |
| Id кадастровый номер | bigint | Простой | Натуральное число | PK | Да |
| id\_Владельцы | int | Простой | Натуральное число | FK | Да |
| id\_Здание | int | Простой | Натуральное число | FK | Да |
| Дата | date(10) | Простой | \*\*. \*\*. \*\*\*\*  X => 0..9 |  | Да |
| Id\_Наименование органа регистрации прав | int | Простой | Натуральное число | FK | Да |
| Стоимость | decimal | Простой | Вещественное положительное число |  | Да |
| id\_адрес | int | Простой | Натуральное число | FK | Да |
| Дом | Char(10) | Простой | Кириллица, натуральное число, ‘/’ | FK | Да |
| Телефон | сhar(15) | Простой | Натуральное число, “+” |  | Да |

*Улица*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Ключи** | **Обязательно к заполнению** |
| id\_улицы | int | Простой | Натуральное число | PK | Да |
| id\_дома | int | Простой | Натуральное число | FK | Да |

*Дом*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Ключи** | **Обязательно к заполнению** |
| id\_дома | int | Простой | Натуральное число | PK | Да |
| дом | Char(10) | Составной | Кириллица, натуральное число, ‘/’ |  | Да |

*Здание*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Ключи** | **Обязательно к заполнению** |
| Id\_Здания | int | Простой | Натуральное число | PK | Да |
| Наименование | char(50) | Простой | Кириллица, ‘-‘ |  | Да |
| Количество этажей | int | Простой | Целое число |  | Да |

*Часть здания*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Ключи** | **Обязательно к заполнению** |
| id\_здания | int | Простой | Натуральное число | FK | Да |
| id\_помещения | int | Простой | Натуральное число | FK | Да |

*Помещение*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Ключи** | **Обязательно к заполнению** |
| Инвентарный номер | bigint | Простой | Натуральное число | PK | Да |
| id\_назначение | int | Простой | Натуральное число | FK | Да |
| Этаж | int | Простой | Целое число |  | Да |

*Назначение помещения*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Ключи** | **Обязательно к заполнению** |
| id\_назначение | int | Простой | Натуральное число | PK | Да |
| Тип назначения | char (50) | Простой | Кириллица |  | Да |

*Часть помещения*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Ключи** | **Обязательно к заполнению** |
| Инвентарный номер | bigint | Простой | Натуральное число | PK | Да |
| Помещение | int | Простой | Натуральное число | FK | Да |
| id\_Назначение | int | Простой | Натуральное число | FK | Да |
| Площадь | decimal | Простой | Вещественное положительное число |  | Да |

*Назначение части помещения*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Вид** | **Домен** | **Ключи** | **Обязательно к заполнению** |
| id\_назначение части помещения | int | Простой | Натуральное число | PK | Да |
| Тип назначения | char (50) | Простой | Кириллица |  | Да |

* Проверка отношений с правилами нормализации

Первое правило нормализации: переменная отношения находится в 1НФ тогда и только тогда, когда в любом допустимом значении отношений каждый его кортеж содержит только одно значение для каждого из атрибутов.

Таблица *Владельцы* находится в 1 нормальной форме, так как ни одна из ее строк не содержит в любом своем поле более одного значения и ни одно из ее ключевых полей не пусто.

Таблица *Кадастр-владельцы* находится в 1 нормальной форме, так как ни одна из ее строк не содержит в любом своем поле более одного значения и ни одно из ее ключевых полей не пусто.

Таблица *Кадастр* находится в 1 нормальной форме, так как ни одна из ее строк не содержит в любом своем поле более одного значения и ни одно из ее ключевых полей не пусто.

Таблица Кадастр находится в 1 нормальной форме, так как ни одна из ее строк не содержит в любом своем поле более одного значения и ни одно из ее ключевых полей не пусто.

Таблица *Наименование органа регистрации прав* находится в 1 нормальной форме, так как ни одна из ее строк не содержит в любом своем поле более одного значения и ни одно из ее ключевых полей не пусто.

Таблица *Улица* находится в 1 нормальной форме, так как ни одна из ее строк не содержит в любом своем поле более одного значения и ни одно из ее ключевых полей не пусто.

Таблица *Дом* находится в 1 нормальной форме, так как ни одна из ее строк не содержит в любом своем поле более одного значения и ни одно из ее ключевых полей не пусто.

Таблица *Здание* находится в 1 нормальной форме, так как ни одна из ее строк не содержит в любом своем поле более одного значения и ни одно из ее ключевых полей не пусто.

Таблица *Помещение* находится в 1 нормальной форме, так как ни одна из ее строк не содержит в любом своем поле более одного значения и ни одно из ее ключевых полей не пусто.

Таблица *Назначение помещения* находится в 1 нормальной форме, так как ни одна из ее строк не содержит в любом своем поле более одного значения и ни одно из ее ключевых полей не пусто.

Таблица *Часть помещения* находится в 1 нормальной форме, так как ни одна из ее строк не содержит в любом своем поле более одного значения и ни одно из ее ключевых полей не пусто.

Таблица *Назначение части помещения* находится в 1 нормальной форме, так как ни одна из ее строк не содержит в любом своем поле более одного значения и ни одно из ее ключевых полей не пусто.

Второе правило нормализации: переменная отношения находится во 2НФ тогда и только тогда, когда она находится в 1НФ и каждый не ключевой атрибут функционально полно зависит от ее потенциального ключа.

Таблица *Владельцы* находится во 2 НФ, так как она имеет простой первичный ключ и все атрибуты зависят от этого ключа.

Таблица *Кадастр* находится во 2 НФ, так как она имеет простой первичный ключ и все атрибуты зависят от этого ключа.

Таблица *Назначение органа регистрации прав* находится во 2 НФ, так как она имеет простой первичный ключ и все атрибуты зависят от этого ключа.

Таблица *Улица* находится во 2 НФ, так как она имеет простой первичный ключ и все атрибуты зависят от этого ключа.

Таблица *Дом* находится во 2 НФ, так как она имеет простой первичный ключ и все атрибуты зависят от этого ключа.

Таблица *Здание* находится во 2 НФ, так как она имеет простой первичный ключ и все атрибуты зависят от этого ключа.

Таблица *Помещение* находится во 2 НФ, так как она имеет простой первичный ключ и все атрибуты зависят от этого ключа.

Таблица *Назначение помещения* находится во 2 НФ, так как она имеет простой первичный ключ и все атрибуты зависят от этого ключа.

Таблица *Часть помещения* находится во 2 НФ, так как она имеет простой первичный ключ и все атрибуты зависят от этого ключа.

Таблица *Назначение части помещения* находится во 2 НФ, так как она имеет простой первичный ключ и все атрибуты зависят от этого ключа.

Третье правило нормализации: переменная отношения находится в 3НФ тогда и только тогда, когда она находится во 2НФ и отсутствуют транзитивные зависимости не ключевых атрибутов от ключевых.

Все таблицы данной базы находятся в 3НФ. Это значит, что все таблицы во 2НФ и все атрибуты таблиц зависят только от первичного ключа, а не от других атрибутов таблицы.

* Проверка соответствия отношений требованиям пользовательских транзакций.

Все отношения соответствуют требованиям.

* Определение требований поддержки бизнес правил и целостности данных.

В ходе работы сформированы следующие бизнес-правила:

Запрет пустого значения.

Этаж, на котором находится помещение, должен быть не больше количества этажей в здании.

**Вывод:** В ходе работы была спроектирована база данных согласно заданному набору атрибутов и сущностей. Работа включала в себя концептуальное и логическое проектирование. Концептуальное проектирование позволяет определить основные объекты, и в целом ознакомится с предметной областью. Целью логического проектирования является получение точной структуры данных для поддержки функциональных требований, выделенных на этапе концептуального проектирования. Оба этапа необходимы для успешной разработки БД и должны выполнятся последовательно.