|  |
| --- |
| **МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ** |
| **ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ** |
| **УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ** |
| **«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»** |
|  |
| Институт информационных технологий и управления в технических системах |
| (полное название института) |
|  |
| кафедра «Информационные системы» |
| (полное название кафедры) |

**ОТЧЁТ**

по лабораторной работе №3

на тему«Создание схемы БД. Ссылочная целостность»

по дисциплине **«**Теория баз данных»

Вариант 16

Выполнил

студент ИИТУТС

группы ИС/б-18-2-о

Радыгина Екатерина

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | | | | | | | | |
|  | (должность, учёная степень преподавателя) | | | | | | | | |
|  |  | | | | | | | | |
|  | (ФИО преподавателя) | | | | | | | | |
|  | « |  | » |  |  |  | 20 | 20 | г. |
|  |  | | | | | | | | |
|  | (оценка) | | | | | | | | |
|  |  | | | | | | | | |
|  |  | | | | | | | | |

Севастополь 2020

**3.1 Цель работы**

Научиться анализировать предметную область с целью создания схемы БД, учитывая ссылочную целостность набора.

**3.2 Вариант задания**

Вариант 16

На рисунке 3.1 изображена структура системы, которая содержит информацию о грузовых перевозках, осуществляемых различными фирмами.

Категория водительских прав – A, B, C.

Характер груза – твердый, жидкий и т.д.

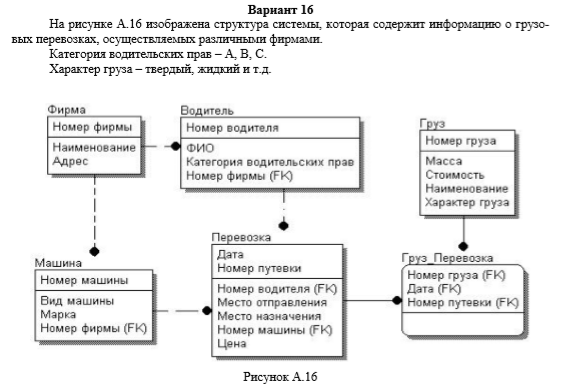


Рисунок 3.1 – Вариант задания

**3.3 Ход работы**

3.3.1 Проанализируем схему БД, выделим и классифицируем все существующие связи, определим необходимые ограничения целостности.

Таблицы связаны отношением 1:М.

Водитель:

«Номер водителя» – целочисленное, не может принимать значение «null»

«ФИО» – строка переменной длины с максимальной длиной в 25 символов

«Категория водительских прав» ­– строка переменной длины с максимальной длиной в 3 символа

«Номер фирмы» – целочисленное, не может принимать значение «null»

Фирма:

«Номер фирмы» – целочисленное, не может принимать значение «null»

«Наименование» – строка переменной длины с максимальной длиной в 20 символов

«Адрес» – строка переменной длины с максимальной длиной в 20 символов

Машина:

«Номер машины» – целочисленное, не может принимать значение «null»

«Вид машины» – строка переменной длины с максимальной длиной в 15 символов

«Марка» – строка переменной длины с максимальной длиной в 15 символов

«Номер фирмы» – целочисленное, не может принимать значение «null»

Перевозка:

«Дата» – имеет тип «date»

«Номер путевки» – целочисленное, не может принимать значение «null»

«Номер водителя» – целочисленное, не может принимать значение «null»

«Место отправления» – строка переменной длины с максимальной длиной в 20 символов

«Место назначения» – строка переменной длины с максимальной длиной в 20 символов

«Номер машины» – целочисленное, не может принимать значение «null»

«Цена» – переменная с фиксированными точностью 2 и масштабом 6, не может принимать значение «null»

Груз\_Перевозка:

«Номер груза» – целочисленное, не может принимать значение «null»

«Дата» – имеет тип «date»

«Номер путевки» – целочисленное, не может принимать значение «null»

Груз:

«Номер груза» – целочисленное, не может принимать значение «null»

«Масса» – целочисленное, не может принимать значение «null»

«Стоимость» – переменная с фиксированной точностью 2 и масштабом 6, не может принимать значение «null»

«Наименование» – строка переменной длины с максимальной длиной в 20 символов

«Характер груза» – строка переменной длины с максимальной длиной в 15 символов

Ограничения:

– объявляются первичные ключи в полях

– описываются внешние ключи, ссылающиеся на поля с аналогичными названиями

3.3.2 Создадим все таблицы, изменим существующие таким образом, чтобы они могли участвовать в связях и установим их (рисунок 3.2 – 3.8).

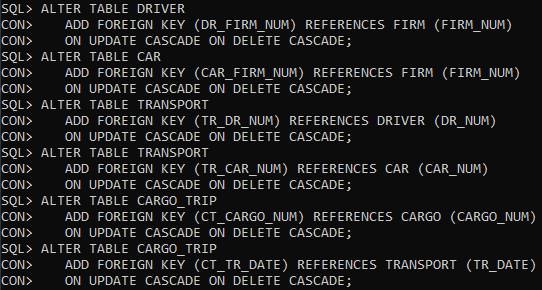
 

Рисунок 3.2 – Установка FOREIGN KEY

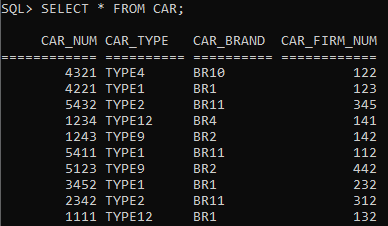


Рисунок 3.3 – Таблица CAR

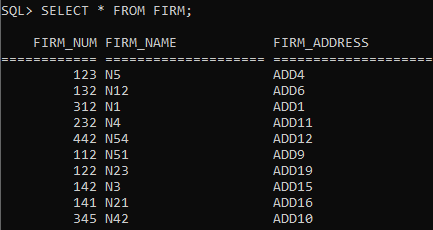


Рисунок 3.4 – Таблица FIRM

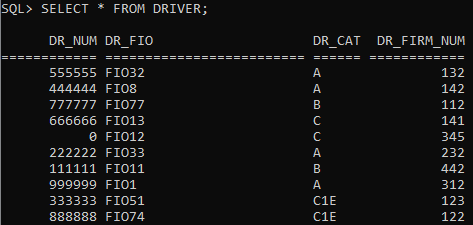


Рисунок 3.5 – Таблица DRIVER

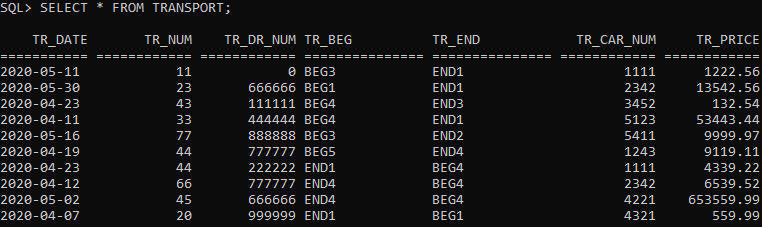


Рисунок 3.6 – Таблица TRANSPORT

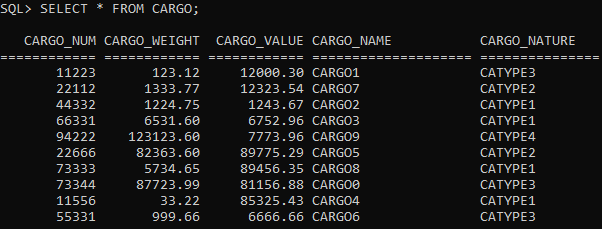


Рисунок 3.7 – Таблица CARGO

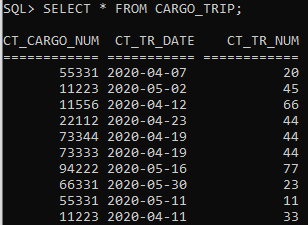


Рисунок 3.8 – Таблица CARGO\_TRIP

3.3.3 Проверим работу ограниченности целостности:

* Каскадирование удаления. Удалим из таблицы TRANSPORT строку. Проверим изменения в таблице CARDO\_TRIP (рисунок 3.9).
* Модификация значения. Изменим номер водителя (DR\_NUM) в таблице DRIVER. Проверим изменения в таблице TRANSPORT (рисунок 3.10).

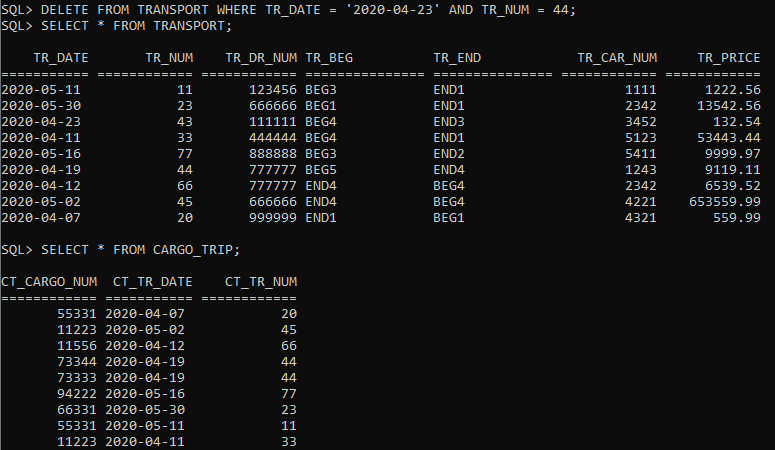


Рисунок 3.9 – Каскадирование удаления

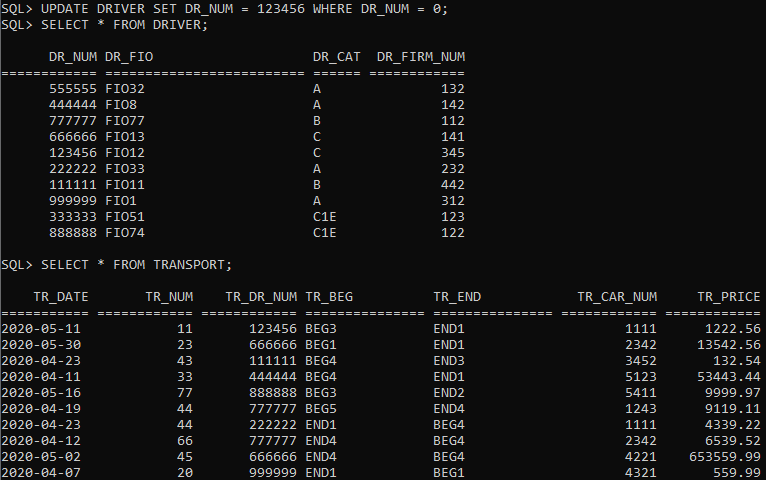


Рисунок 3.10 – Модификация значения

Вывод

В ходе лабораторной работы была проанализирована предметная область с целью создания схемы БД, учитывая ссылочную целостность набора.

**КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. **Требования к ссылочной целостности данных?**

Необходимо устанавливать связь между таблицами по определённому полю. В этом случае сервер базы данных автоматически проследит, чтобы это поле существовало в другой таблице.

1. **Типы связей между отношениями?**

Существует три основных класса связей между таблицами:

– один к одному (1:1);

– один ко многим (1:М);

– многие ко многим (М:М).

На практике связи 1:1 используются редко. Связи М:М не реализуются в реляционных базах данных напрямую – одну связь М:М приводят к двум связям 1:М. Связь 1:М используется наиболее часто.

1. **Стандартная нотация ER-диаграмм?**

Сущность изображается прямоугольником, над которым пишется имя сущности. Одна сущность соответствует одной таблице. Прямоугольник разделяется линией на две части. В верхней части указываются ключевые атрибуты. После имени атрибута могут стоять символы, уточняющие назначение атрибута. Допустимые символы:

* PK ­­– первичный ключ;
* AK – альтернативный ключ;
* FK – внешний ключ;
* IE – по данному полю должен быть создан индекс.

Связь между двумя отношениями изображается с помощью линии. Идентифицирующая связь изображается сплошной линией, не идентифицирующая – пунктирной. Арность связи указывается следующим образом: со стороны «многие» ставится жирная точка, а со стороны «один» точка не ставится. Допустимость Null – значений изображается ромбом с той стороны связи, где позволяются Null – значения.

В случае если у нас есть некоторая общая сущность необходимо хранить информацию о некоторых разновидностях данной сущности имеет место понятие категории.

**4. Нормальные формы для баз данных?**

**Нормальная форма** – требование, предъявляемое к структуре таблиц в теории реляционных баз данных для устранения из базы избыточных функциональных зависимостей между атрибутами (полями таблиц).

**Метод нормальных форм (НФ)** состоит в сборе информации о объектах решения задачи в рамках одного отношения и последующей декомпозиции этого отношения на несколько взаимосвязанных отношений на основе процедур нормализации отношений.

**Первая нормальная форма**. Отношение находится в 1НФ, если все его атрибуты являются простыми, все используемые домены должны содержать только скалярные значения. Не должно быть повторений строк в таблице.

**Вторая нормальная форма.** Отношение находится во 2НФ, если оно находится в 1НФ и каждый не ключевой атрибут неприводимо зависит от Первичного Ключа (ПК).

Неприводимость означает, что в составе потенциального ключа отсутствует меньшее подмножество атрибутов, от которого можно также вывести данную функциональную зависимость.

**Третья нормальная форма.** Отношение находится в 3НФ, когда находится во 2НФ и каждый не ключевой атрибут нетранзитивно зависит от первичного ключа. Проще говоря, второе правило требует выносить все не ключевые поля, содержимое которых может относиться к нескольким записям таблицы в отдельные таблицы.

**Нормальная форма Бойса-Кодда (НФБК) (частная форма третьей нормальной формы)**

Определение 3НФ не совсем подходит для следующих отношений:  
1) отношение имеет два или более потенциальных ключ;

2) два и более потенциальных ключа являются составными;

3) они пересекаются, т.е. имеют хотя бы один атрибут.

Для отношений, имеющих один потенциальный ключ (первичный), НФБК является 3НФ. Отношение находится в НФБК, когда каждая нетривиальная и неприводимая слева функциональная зависимость обладает потенциальным ключом в качестве детерминанта.

**Четвертая нормальная форма.** Отношение находится в 4НФ, если оно находится в НФБК и все нетривиальные многозначные зависимости фактически являются функциональными зависимостями от ее потенциальных ключей.

**Пятая нормальная форма.** Переменная отношения находится в пятой нормальной форме (иначе — в проекционно-соединительной нормальной форме) только тогда, когда каждая нетривиальная зависимость соединения в ней определяется потенциальным ключом (ключами) этого отношения.

**Доменно-ключевая нормальная форма.** Переменная отношения находится в ДКНФ тогда и только тогда, когда каждое наложенное на неё ограничение является логическим следствием ограничений доменов и ограничений ключей, наложенных на данную переменную отношения.

6. Переменная отношения находится в шестой нормальной форме только тогда, когда она удовлетворяет всем нетривиальным зависимостям соединения. Из определения следует, что переменная находится в 6НФ тогда и только тогда, когда она неприводима, то есть не может быть подвергнута дальнейшей декомпозиции без потерь. Каждая переменная отношения, которая находится в 6НФ, также находится и в 5НФ.

**5. Необходимость процесса нормализации базы данных?**

Нормализация предназначена для приведения структуры БД к виду, обеспечивающему минимальную логическую избыточность, и не имеет целью уменьшение или увеличение производительности работы или же уменьшение или увеличение физического объёма базы данных. Конечной целью нормализации является уменьшение потенциальной противоречивости хранимой в базе данных информации. Как отмечает Кристофер Дейт, общее назначение процесса нормализации заключается в следующем:

– исключение некоторых типов избыточности;

– устранение некоторых аномалий обновления;

– разработка проекта базы данных, который является достаточно «качественным» представлением реального мира, интуитивно понятен и может служить хорошей основой для последующего расширения;

– упрощение процедуры применения необходимых ограничений целостности.

**6. Приведение БД к нормальной форме Бойса-Кодда?**

Переменная отношения находится в BCNF только тогда, когда каждая её нетривиальная и неприводимая слева функциональная зависимость имеет в качестве своего детерминанта некоторый потенциальный ключ.

Можно улучшить структуру с помощью декомпозиции отношения на два, получив отношения, удовлетворяющие BCNF.

**7. Обоснование нормальной формы полученной схемы базы данных?**

1. Порядок строк не несёт в себе никакой информации.

2. Порядок столбцов не несёт в себе никакой информации.

3. Нет повторяющихся строк.

4. Каждое пересечение строки и столбца содержит ровно одно значение из соответствующего домена.

5. Все столбцы являются обычными.

**8. Способы повышения надёжности данных?**

Установка связи между таблицами по определённому полю. В этом случае сервер базы данных автоматически проследит, чтобы это поле существовало в другой таблице.