**Задание 4.1** Спроектировать реляционную базу данных, все таблицы которой соответствуют 3НФ, на основе исходного набора данных.

НАЧАЛЬНОЕ ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

|  |  |
| --- | --- |
| **Поле** | **Описание** |
| CatalogNum | Каталожный номер товара |
| Product | Название товара |
| Price | Цена за единицу товара (всегда одинакова для одной позиции каталога) |
| OrderNum | Номер заказа. Заказ делается конкретным клиентом и может включать одну или несколько позиций |
| Quantity | Количество единиц указанного товара в заказе |
| CustomerNum | Номер, идентифицирующий заказчика |
| CustomerName | Фамилия и инициалы заказчика |
| CustomerAddress | Адрес заказчика |
| CustomerPhone | Телефон заказчика |

**Считаем что:**

* Заказчик идентифицируется уникальным номером. Среди заказчиков могут быть однофамильцы. У заказчика только один адрес и телефон.
* Товар идентифицируется каталожным номером.
* Цена товара определяется только его номером в каталоге, цена постоянна, скидок и наценок нет.
* Если в одном заказе несколько штук одного товара указывается количество, запись в таблице делается только одна.
* Если в одном заказе несколько разных товаров, делается несколько записей в таблице.
* Заказ идентифицируется уникальным номером и делается определенным заказчиком.

Объединим все поля в единую таблицу “Products”:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Catalog Num | Product | Price | OrderNum | Quantity | Customer Num | Customer Name | CustomerAddress | CustomerPhone |

Устраним избыточность данных.

Вынесем поля Catalog Num, Product, Price в отдельную новую таблицу “CatalogProducts”, но в таблице Products оставим атрибут CatalogNum как связующее звено.

Таблица Products

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Catalog Num | OrderNum | Quantity | Customer Num | Customer Name | CustomerAddress | CustomerPhone |

Таблица CatalogProducts

|  |  |  |
| --- | --- | --- |
| Catalog Num | Product | Price |

Вынесем поля CustomerNum, CustomerName, CustomerAddress, CustomerPhone в отдельную таблицу “CustomerProducts”, а в таблице Products останется атрибут CustomerNum как связующее звено.

Таблица Products

|  |  |  |  |
| --- | --- | --- | --- |
| Catalog Num | OrderNum | Quantity | Customer Num |

Таблица CatalogProducts

|  |  |  |
| --- | --- | --- |
| Catalog Num | Product | Price |

Таблица CustomerProducts

|  |  |  |  |
| --- | --- | --- | --- |
| Customer Num | Customer Name | CustomerAddress | CustomerPhone |

**1НФ** Таблица находится в 1НФ, если все ее поля содержат только простые неделимые значения.

Рассмотрим атрибут CustomerAddress. Он состоит из составной информации. Так как при заказе товара указывается город и можно предположить что если в адресе указывается город, то спортивные товары доставляются не только в рамках одного города но и конкретного адреса. Поэтому атрибут CustomerAddress необходимо разделить на 2 атрибута: City и LocalAddress. Теперь все таблицы приведены к 1НФ.

Таблица CustomerProducts

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Customer Num | Customer Name | City | LocalAddress | CustomerPhone |

**2НФ** Таблица находится в 2НФ, если она удовлетворяет требованиям 1НФ и неключевые поля функционально полно зависят от первичного ключа.

Согласно этому требованию необходимо добавить или определить из имеющихся атрибутов первичный ключ. В таблице Products есть атрибут OrderNum он не уникальный, то есть его не сделать первичным ключом. Но можно использовать составной ключ с атрибутом CatalogNum. И тогда от данной комбинации атрибутов будут зависеть неключевые атрибуты записи. Следовательно таблицы находятся в 2НФ.

Таблица Products

|  |  |  |  |
| --- | --- | --- | --- |
| **Catalog Num** | **OrderNum** | Quantity | Customer Num |

**3НФ** Таблица находится в 3НФ, если она удовлетворяет требованиям 2НФ и не содержит транзитивных зависимостей.

Транзитивная зависимость – функциональная зависимость между не ключевыми полями. При анализе таблиц транзитивных зависимостей не обнаружено, поэтому данные таблицы находятся в 3НФ.

Таблица Products

|  |  |  |  |
| --- | --- | --- | --- |
| **Catalog Num** | **OrderNum** | Quantity | Customer Num |

Таблица CatalogProducts

|  |  |  |
| --- | --- | --- |
| Catalog Num | Product | Price |

Таблица CustomerProducts

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Customer Num | Customer Name | City | LocalAddress | CustomerPhone |

Разработана реляционная БД, таблицы которой соответствуют 3НФ.

1. Первичные ключи и обязательные атрибуты: CustomerNum, CatalogNum, OrderNum.
2. Связи между отношениями:

* Таблица Products и таблица CatalogProducts находится в отношении 1 : М т.к одна позиция из каталога товаров много раз может быть заказана.
* Таблица Products и таблица CustomersProducts находится в отношении 1 : М т.к. один заказчик может сделать много разных заказов

3) Внешние ключи таблицы Products: CustomerNum, CatalogNum.