Министерство науки и высшего образования РФ

федеральное государственное автономное  
образовательное учреждение высшего образования

«Омский государственный технический университет»

Факультет информационных технологий и компьютерных систем

Кафедра «Прикладная математика и фундаментальная информатика»

|  |
| --- |
| **Лабораторная работа №1** |
| по дисциплине **Системы управления базами данных**  **Тема: ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ С ИСПОЛЬЗОВАНИЕМ ER-ТЕХНОЛОГИИ** |

|  |  |  |  |
| --- | --- | --- | --- |
| Студента | Кузиной Варвары Михайловны | | |
| Курс | 3 | Группа | МО-231 |
| Направление | 02.03.03 Математическое обеспечение и администрирование информационных систем | | |
| Руководитель | доц., к.н. | | |
|  | Моисеева Н.А. | | |
| Выполнил | 16.09 | | |
| Проверил |  | | |

Омск 2025

Для решения задачи управления данными процесса продажи товаров в компании была спроектирована база данных, которая будет отслеживать информацию о сотрудниках, чеках, товарах и продажах. В базе данных выделены следующие сущности: **сотрудник**, **чек**, **товар**, **каталог товаров**, **состав чека**, **накладная**, **состав накладной**.

В рамках предметной области действуют следующие бизнес-правила:

1. Каждый **чек** оформляется одним сотрудником, а один сотрудник может оформить несколько чеков.
2. Каждый **чек** может содержать несколько товаров.
3. Каждый товар может быть в нескольких чеках.
4. **Накладная** отражает информацию о поставке товаров, и один товар может быть в нескольких накладных.

**Логическая модель (изображена на рисунке 1):**

В логической модели выделены следующие сущности:

* **Заявки (Orders)**
* **Состав заявки (Order composition)**
* **Чеки (Receipts)**
* **Состав чека (Check composition)**
* **Каталог товаров (Product catalog)**
* **Накладные (Invoices)**
* **Состав накладной (Invoice composition)**

**Связи между сущностями:**

1. **Заявка** может содержать несколько товаров. Связь между **Заявкой** и **Составом заявки** – 1 ко многим.
2. Каждый **чек** оформляется одним сотрудником, а один сотрудник может оформить множество чеков. Связь между **Чеками** и **Сотрудниками** – 1 ко многим.
3. Каждый **чек** может содержать несколько товаров, а каждый товар может присутствовать в разных чеках. Связь между **Чеком** и **Составом чека** – 1 ко многим.
4. **Каталог товаров** содержит информацию обо всех товарах. Связь между **Каталогом товаров** и **Составом чека** – 1 ко многим.
5. Каждый **товар** может быть в нескольких накладных. Связь между **Накладными** и **Составом накладной** – 1 ко многим.

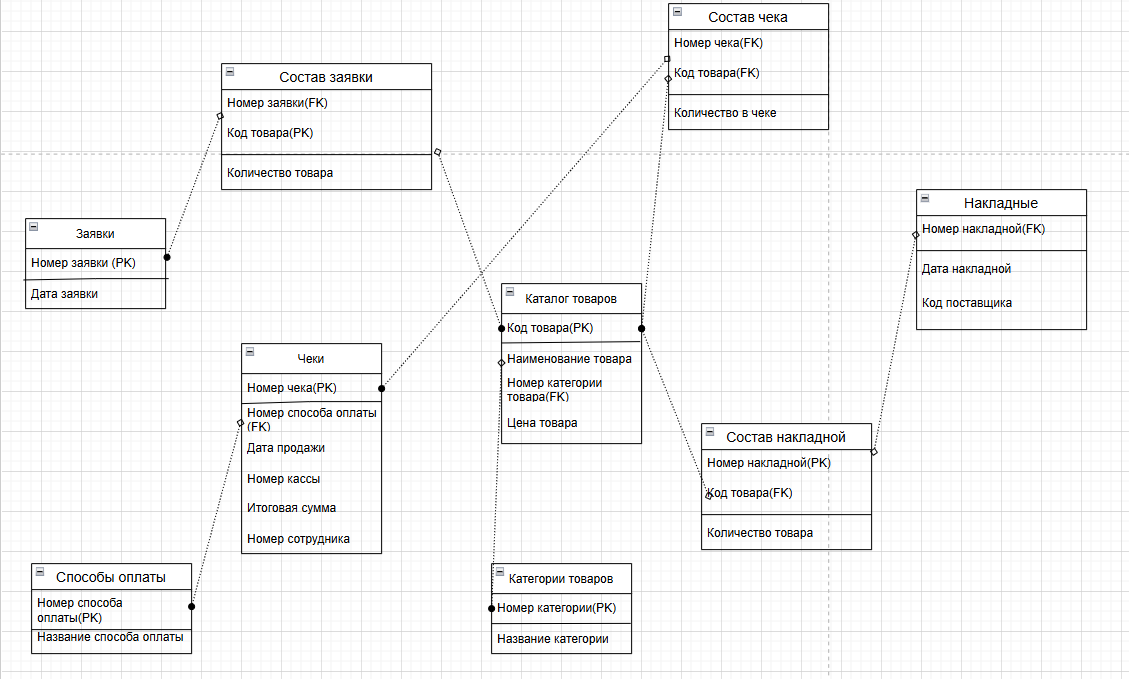


Рисунок 1 — Логическая модель предметной области

**Физическая модель (изображена на рисунке 2):**

**Таблица «Заявки (Orders)»**

* **order\_number** – уникальный номер заявки, тип integer, NOT NULL, Primary Key.
* **order\_date** – дата заявки, тип date, NOT NULL.

**Таблица «Состав заявки (Order composition)»**

* **order\_number** – номер заявки (FK), тип integer, NOT NULL.
* **product\_code** – код товара (FK), тип integer, NOT NULL.
* **product\_quantity** – количество товара в заявке, тип integer, NOT NULL.

**Таблица «Чеки (Receipts)»**

* **receipt\_number** – номер чека, тип integer, NOT NULL, Primary Key.
* **receipt\_date** – дата продажи, тип datetime, NOT NULL.
* **cash\_register\_number** – номер кассового аппарата, тип integer, NOT NULL.
* **employee\_id** – ID сотрудника, оформившего чек (FK), тип integer, NOT NULL.

**Таблица «Состав чека (Check composition)»**

* **receipt\_number** – номер чека (FK), тип integer, NOT NULL.
* **product\_code** – код товара (FK), тип integer, NOT NULL.
* **quantity\_in\_check** – количество товара в чеке, тип integer, NOT NULL.

**Таблица «Каталог товаров (Product catalog)»**

* **product\_code** – код товара, тип integer, NOT NULL, Primary Key.
* **product\_name** – наименование товара, тип text, NOT NULL.
* **product\_category** – категория товара, тип text, NOT NULL.
* **product\_price** – цена товара, тип decimal, NOT NULL.

**Таблица «Накладные (Invoices)»**

* **invoice\_number** – номер накладной, тип integer, NOT NULL, Primary Key.
* **invoice\_date** – дата накладной, тип date, NOT NULL.
* **supplier\_code** – код поставщика, тип integer, NOT NULL.

**Таблица «Состав накладной (Invoice composition)»**

* **invoice\_number** – номер накладной (FK), тип integer, NOT NULL.
* **product\_code** – код товара (FK), тип integer, NOT NULL.
* **product\_quantity** – количество товара в накладной, тип integer, NOT NULL.

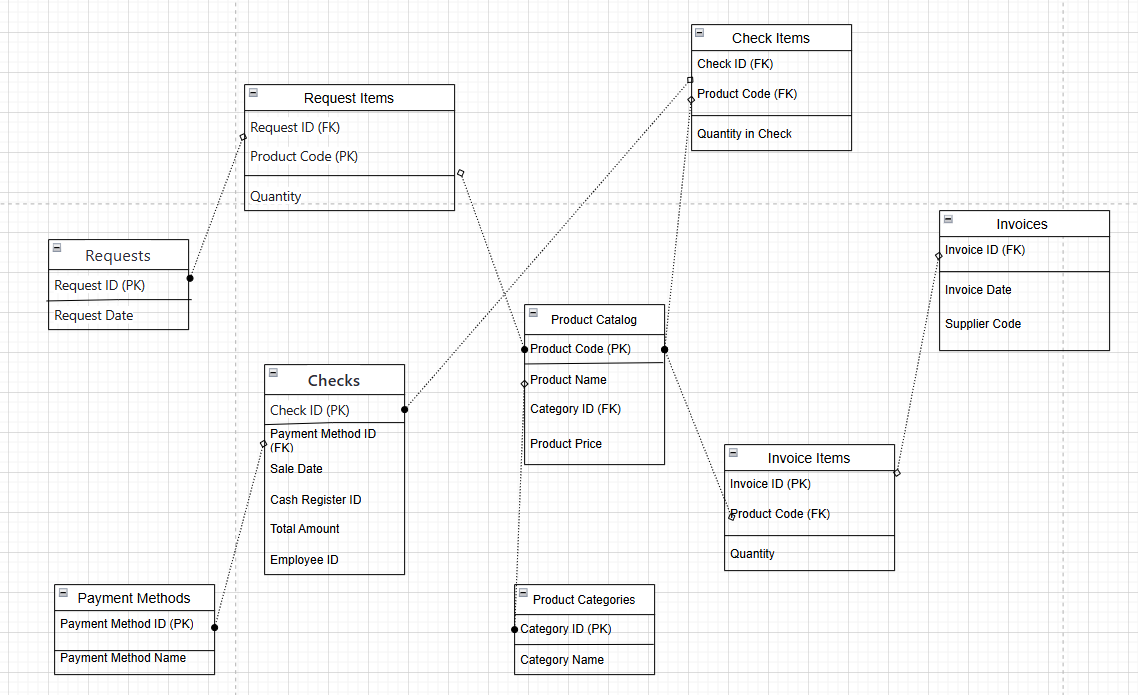
****

Рисунок 2 — Физическая модель предметной области

**Ограничения целостности данных:**

В таблицах базы данных используются **внешние ключи** для связывания сущностей между собой:

* В таблице **Состав заявки** поле **order\_number** связано с таблицей **Заявки** через внешний ключ.
* В таблице **Состав чека** поля **receipt\_number** и **product\_code** связаны с таблицами **Чеки** и **Каталог товаров** через внешние ключи.
* В таблице **Состав накладной** поля **invoice\_number** и **product\_code** связаны с таблицами **Накладные** и **Каталог товаров** через внешние ключи.

Для всех связей предусмотрены **каскадные удаления и обновления**. Это означает, что при удалении записи из родительской таблицы все связанные с ней записи из дочерних таблиц будут удалены автоматически.

**Заключение:**

Проектируемая база данных для управления процессами продажи товаров и учета накладных является логичным и эффективным инструментом для организации данных в компании. Использование логической и физической моделей позволяет эффективно хранить и обрабатывать информацию о товарах, продажах и сотрудниках, соблюдая целостность данных и их актуальность.