

**Содержание**

[Введение 3](#_Toc154333284)

[Глава 1. Предметная область 5](#_Toc154333285)

[Глава 2. Логическое проектирование 7](#_Toc154333286)

[Глава 3. Физическое проектирование 8](#_Toc154333287)

[Глава 4. Примеры sql-запросов 9](#_Toc154333288)

[Заключение 10](#_Toc154333289)

[Список используемой литературы 11](#_Toc154333290)

[Приложение 1. Схема взаимодействий 13](#_Toc154333291)

[Приложение 2. ER-диаграмма 14](#_Toc154333292)

[Приложение 3. Листинг кода 15](#_Toc154333293)

# ВВЕДЕНИЕ

Вместе с развитием технологий и общей доступности Интернета круг деятельности начал переноситься на онлайн-платформы. Одним из них стала сфера торговли, именно здесь индустрия услуг получила новый импульс развития. Уже сейчас мы можем приобрести практически любой товар, не вставая со стула, помогают нам в этом различные - Интернет-магазины.

Сегодня, покупая товары через Интернет, мы ожидаем не только удобства и быстроты доставки, но и качественного обслуживания. В связи с этим важным требованием все больше компаний, занимающихся продажами одежды, открывают собственные Интернет-магазины, чтобы удовлетворить растущие потребности современных покупателей.

Интернет-торговля имеет массу преимуществ, которые особо ощутимы на этапе создания бизнеса. Открытие Интернет-магазина обойдется гораздо дешевле, чем запуск традиционного магазина. Снижение издержек на открытие позволит предпринимателю снизить наценку на товар и обеспечить более низкие цены, чем в обычных магазинах.

Существует великое множество причин, по которым компания может решить выйти в Интернет, однако все эти причины можно разбить на несколько категорий:

 экономия денежных средств;

 экономия времени;

 предоставление возможности клиентам выбирать и приобретать необходимые товары 24 часа в сутки 7 дней в неделю.

Разработка базы данных «Интернет-магазин одежды» является актуальной и востребованной темой, поскольку эффективная организация и управление данными в такой сфере является ключевым фактором успеха бизнеса.

В данной курсовой работе будет рассмотрена разработка базы данных для Интернет-магазина одежды, позволяющая оптимизировать процессы управления товарными запасами, заказами, доставкой и клиентскими данными, а также обеспечить более качественное обслуживание клиентов.

Целью курсовой работы является повышение эффективности управления информацией о товарах, клиентах, заказах и других важных аспектах функционирования Интернет-магазина.

Задачи курсовой работы включают в себя следующее:

* изучение предметной области и создание информационной системы для поддержки управленческих решений;
* проектирование логической структуры базы данных;
* проектирование физической структуры базы данных;
* реализация базы данных и заполнение её тестовыми данными.

Задачи проектирования базы данных включают:

* определение структуры базы данных;
* проектирование нормализованной структуры данных;
* определение связей между таблицами;
* разработка схемы базы данных, создание ER-модели.

# ГЛАВА 1. ПРЕДМЕТНАЯ ОБЛАСТЬ

Любая база данных является ключевым компонентом информационной системы. Анализ предметной области является необходимым начальным шагом в разработке любой информационной системы. Именно на этом этапе определяются информационные потребности всех пользователей будущей системы, которые, в свою очередь, определяют содержание ее базы данных. [[4]](#_СПИСОК_ИСПОЛЬЗУЕМОЙ_ЛИТЕРАТУРЫ)

В рамках данной курсовой работы изучается процесс управления Интернет-магазином одежды, который предоставляет услуги по продаже одежды онлайн. Тема курсовой работы направлена на разработку базы данных для такого Интернет-магазина.

Структура и процессы управления Интернет-магазином ориентированы на удобство и безопасность пользователей, а также на эффективную организацию процесса заказа и поставки товара. Взаимодействие между участниками Интернет-магазина начинается с момента регистрации пользователя и продолжается до момента получения заказанной одежды.

Отсюда имеем несколько основных субъектов – товар, покупатель и фирмы, занимающиеся поставкой и/или доставкой товара покупателю. Также имеется объект в виде Интернет-магазина – веб-сайта, с которого осуществляется непосредственно покупка товара покупателем.

Рассмотрим более подробно следующую схему: покупатель посещает сайт Интернет-магазина. С помощью пользовательского интерфейса сайта, покупатель выбирает товары из доступного каталога, где каждый товар имеет свою страницу с ценой, описанием, поставщиком, количеством и доступностью, а также рейтингом и изображениями.

Если покупатель доволен выбранным набором товаров, он осуществляет оплату заказа, в процессе которой передаются данные о заказе и информация о пользователе (ФИО, телефон, электронная почта, почтовый индекс). Затем происходит обработка заказа и его оплата со стороны покупателя, который также получает электронный чек за оплату.

Затем заказ передается поставщику, который упаковывает товар и организует его доставку покупателю. В качестве альтернативы, покупатель может забрать товар лично, одновременно с этим отправляя владельцу магазина счет за купленные товары в случае, если доставка необходима.

После завершения доставки, магазин также получает счет за доставку, а покупатель получает заказанный товар или забирает его самостоятельно.

Схема взаимодействий тех или иных действий на этапах по работе с товаром представлена в виде диаграммы в Приложении 1.

# ГЛАВА 2. ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ

ER-модель - модель данных, позволяющая описывать концептуальные схемы предметной области.

ER-модель представляет собой формальную конструкцию, которая сама по себе не предписывает никаких графических средств её визуализации.

Основные понятия ER-диаграммы включают сущность (объект реального мира), атрибут (свойство сущности) и связь. Атрибут, который уникально идентифицирует экземпляры сущности, называется ключом.

Реляционная база данных представляет собой набор данных с предопределенными связями между ними, организованный в виде набора таблиц, состоящих из столбцов и строк. В таблицах хранится информация об объектах, представленных в базе данных, где каждый столбец таблицы содержит определенный тип данных, а каждая ячейка - значение атрибута. Каждая строка таблицы представляет собой набор связанных значений, относящихся к одному объекту или сущности. Каждая строка в таблице может быть помечена уникальным идентификатором, называемым первичным ключом, а строки из нескольких таблиц могут быть связаны с помощью внешних ключей [[8]](#_СПИСОК_ИСПОЛЬЗУЕМОЙ_ЛИТЕРАТУРЫ).

Данная модель показывает, как осуществляется процесс взаимодействия между покупателем и Интернет-магазином. Сотрудники магазина формируют заказ, занося все необходимые данные (номер заказа, даты заказа и доставки и т.д.), а затем передают его в технический отдел, который потом передает его в отдел доставки. Сотрудники отдела передают заказ курьеру, который в свою очередь берет необходимый товар на складе и доставляет его покупателю.

Во время анализа сферы предметной области для создания модели базы данных были выделены следующие объекты (сущности):

* покупатель;
* заказ;
* корзина;
* каталог;
* тип;
* курьер;
* склад;
* закупка;
* поставщик.

В таблице «Покупатель» (таблица 1), должны быть представлены столбцы: № покупателя, ФИО, телефон, адрес доставки, e-mail.

Таблица 1. Структура таблицы «Покупатель»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| № покупателя | Числовой |
| ФИО | Символьный |
| Телефон | Символьный |
| Адрес доставки | Символьный |
| e-mail | Символьный |

В таблице «Заказ» (таблица 2), должны быть представлены столбцы: № заказа, № покупателя, № курьера, id корзины, дата заказа, дата доставки, способ оплаты, статус оплаты.

Таблица 2. Структура таблицы «Заказ»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| № заказа | Числовой |
| № покупателя | Числовой |
| № курьера | Числовой |
| id корзины | Числовой |
| Дата заказа | Символьный |
| Дата доставки | Символьный |
| Способ оплаты | Символьный |
| Статус оплаты | Символьный |

В таблице «Корзина» (таблица 3), должны быть представлены столбцы: id корзины, № заказа, артикул, количество.

Таблица 3. Структура таблицы «Корзина»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| id корзины | Числовой |
| № заказа | Числовой |
| Артикул | Числовой |
| Количество | Числовой |

В таблице «Каталог» (таблица 4), должны быть представлены столбцы: артикул, наименование, цвет, стоимость, кодтипа.

Таблица 4. Структура таблицы «Каталог»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| Артикул | Числовой |
| Наименование | Символьный |
| Цвет | Символьный |
| Стоимость | Числовой |
| КодТипа | Числовой |

В таблице «Тип» (таблица 5), должны быть представлены столбцы: кодтипа, наименование одежды, пол, кол-во продано.

Таблица 5. Структура таблицы «Тип»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| КодТипа | Числовой |
| Наименование одежды | Символьный |
| Пол | Символьный |
| Кол-во продано | Числовой |

В таблице «Курьер» (таблица 6), должны быть представлены столбцы: № курьера, ФИО, телефон.

Таблица 6. Структура таблицы «Курьер»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| № курьера | Числовой |
| ФИО | Символьный |
| Телефон | Символьный |

В таблице «Закупки» (таблица 7), должны быть представлены столбцы: № закупки, артикул, количество необходимое.

Таблица 8. Структура таблицы «Закупка»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| № закупки | Числовой |
| Артикул | Числовой |
| Количество необходимое | Числовой |

В таблице «Поставщик» (таблица 8), должны быть представлены столбцы: № поставщика, адрес, № закупки, телефон, e-mail.

Таблица 9. Структура таблицы «Поставщик»

|  |  |
| --- | --- |
| Название поля | Тип данных |
| № поставщика | Числовой |
| Адрес | Символьный |
| № закупки | Числовой |
| Телефон | Символьный |
| e-mail | Символьный |

ER-модель для базы данных «Интернет-магазин одежды» представлена в Приложении 2.

# ГЛАВА 3. ФИЗИЧЕСКОЕ ПРОЕКТИРОВАНИЕ

Создание базы данных и её заполнение велось при использовании СУБД PostrgeSQL 16.1 [[7]](#_СПИСОК_ИСПОЛЬЗУЕМОЙ_ЛИТЕРАТУРЫ). Преимуществами за которые была выбрана именно данная СУБД стали:

* бесплатная и открытая СУБД;
* поддержка расширенных функций;
* активное и растущее сообщество пользователей;
* поддержка многих языков программирования;
* высокая производительность.

Рассмотрим структуру отношений, находящихся в базе данных «Движение товаров на складах в строительной организации».

Отношение «Покупатель», изображенное на таблице 1, состоит из полей: customer\_id (код покупателя) – первичный ключ типа bigint, fio (ФИО) типа varchar(100), tel (телефон) типа bigint, delivery\_address (адрес доставки) типа varchar(100), email (email) типа varchar(100).

Таблица 1. Структура отношения "Покупатель"

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Первичный ключ |
| customer\_id | bigint | первичный ключ |
| fio | character varying (100) | - |
| tel | bigint | - |
| delivery\_address | character varying (100) | - |
| email | character varying (100) | - |

Отношение «Курьер», изображенное на таблице 2, состоит из полей: courierr\_id (код курьера) – первичный ключ типа bigint, fio (ФИО) типа varchar(100), tel (телефон) типа bigint.

Таблица 2. Структура отношения "Курьер"

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Первичный ключ |
| courier\_id | bigint | первичный ключ |
| fio | character varying (100) | - |
| tel | bigint | - |

Отношение «Заказ», изображенное на таблице 3, состоит из полей: order\_id (номер заказа) – первичный ключ типа bigint, customer\_id (код покупателя) типа bigint, courierr\_id (код курьера) типа bigint, cart\_id (код записи) типа bigint, other\_date (дата заказа) типа timestamp, delivery\_date (дата доставки) типа timestamp, payment\_method (способ оплаты) типа varchar(20), payment\_status (статус оплаты) типа varchar(20).

Таблица 3. Структура отношения "Заказ"

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Первичный ключ |
| order\_id | bigint | первичный ключ |
| customer\_id | bigint | - |
| courier\_id | bigint | - |
| cart\_id | integer | - |
| order\_date | date | - |
| delivery\_date | date | - |
| payment\_method | character varying (20) | - |
| payment\_status | character varying (20) | - |

Отношение «Корзина», изображенное на таблице 4, состоит из полей: cart\_id (код записи) – первичный ключ типа bigint, order\_id (номер заказа) типа bigint, product\_article (артикул) типа integer, quantity (количество) типа integer.

Таблица 4. Структура отношения "Корзина"

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Первичный ключ |
| cart\_id | bigint | первичный ключ |
| order\_id | bigint | - |
| product\_article | integer | - |
| quantity | integer | - |

Отношение «Каталог», изображенное на таблице 5, состоит из полей: product\_article (артикул) – первичный ключ типа bigint, product\_name (наименование товара) типа varchar(100), color (цвет) типа varchar(50), cost (стоимость) типа integer, product\_type\_code (код типа товара) типа bigint, quantity (кол-во в наличии) типа integer.

Таблица 5. Структура отношения "Каталог"

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Первичный ключ |
| product\_article | bigint | первичный ключ |
| product\_name | character varying (100) | - |
| color | character varying (50) | - |
| cost | integer | - |
| product\_type\_code | bigint | - |
| quantity | bigint | - |

Отношение «Тип товара», изображенное на таблице 6, состоит из полей: product\_type\_code (код типа товара) – первичный ключ типа bigint, product\_type\_name (наименование типа товара) типа varchar(50), gender (пол) типа varchar(15).

Таблица 6. Структура отношения "Тип товара"

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Первичный ключ |
| product\_type\_code | bigint | первичный ключ |
| product\_type\_name | character varying (50) | - |
| gender | character varying (15) | - |

Отношение «Закупка», изображенное на таблице 7, состоит из полей: purchase\_id (номер закупки) – первичный ключ типа bigint, product\_article (артикул) типа bigint, required\_quantity (кол-во необходимое) типа integer.

Таблица 7. Структура отношения "Закупка"

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Первичный ключ |
| purchase\_id | bigint | первичный ключ |
| product\_article | bigint | - |
| required\_quantity | integer | - |

Отношение «Поставщик», изображенное на таблице 8, состоит из полей: supplier\_id (номер поставщика) – первичный ключ типа bigint, address (адрес) типа varchar(100), purchase\_id (номер закупки) типа bigint, tel (телефон) типа bigint, email (email) типа varchar(100).

Таблица 8. Структура отношения "Поставщик"

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Первичный ключ |
| supplier\_id | bigint | первичный ключ |
| address | character varying (100) | - |
| purchase\_id | bigint | - |
| tel | bigint | - |
| email | character varying (100) | - |

# ГЛАВА 4. ПРИМЕРЫ SQL-ЗАПРОСОВ

Чтобы получить данные из таблиц базы данных, необходимо использовать SQL-запросы, которые представляют собой набор команд для работы с реляционными базами данных. При создании SQL-запроса для работы с базами данных в СУБД необходимо указывать следующие параметры:

* Таблицы, из которых необходимо извлечь данные.
* Поля, значения которых необходимо вернуть к исходным после внесения изменений в базу данных.
* Связи между таблицами, если необходимо получить данные из нескольких связанных таблиц.
* Условия выборки, которые определяют, какие записи необходимо вернуть в результате запроса.
* Вспомогательные критерии отбора, такие как ограничения, способы представления информации и тип сортировки.

SQL-запросы позволяют эффективно получать и изменять данные в реляционных базах данных.

Запрос 1. Выведите номер, дату и общую стоимость заказов, которые были сделаны в прошлом месяце.

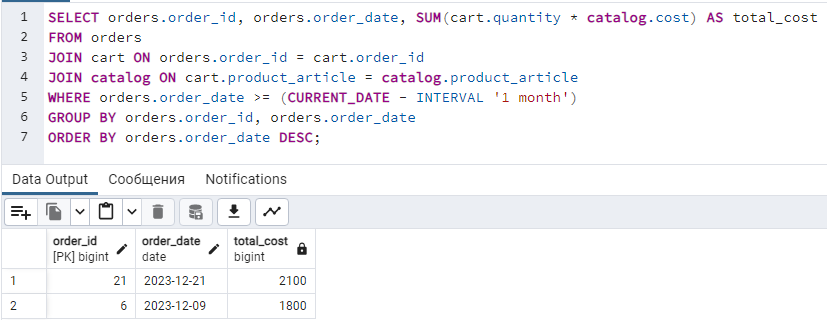


Рисунок 1: Выполнение запроса 1

Запрос 2. Выведите ФИО покупателей, которые сделали более трех заказов в прошлом году, с указанием количества этих заказов.

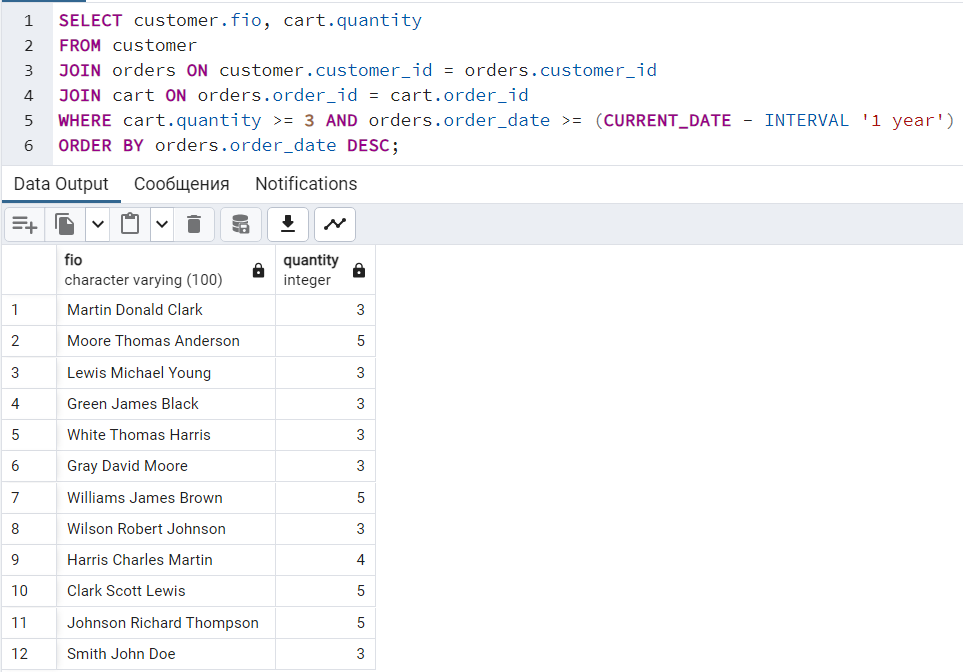


Рисунок 2: Выполнение запроса 2.

Запрос 3. Выведите название, категорию и стоимость товаров, тип которых принадлежит к полу «мужской».

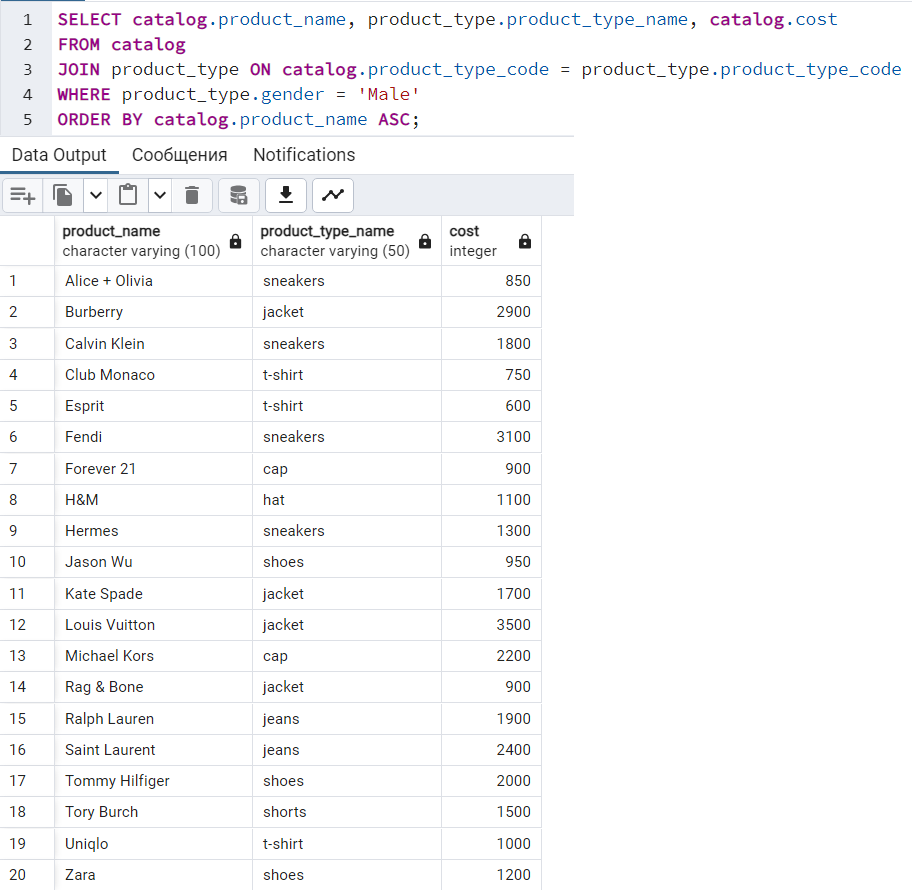


Рисунок 3: Выполнение запроса 3

Запрос 4. Выведите название, категорию и стоимость, самых дорогих женских и мужских футболок.

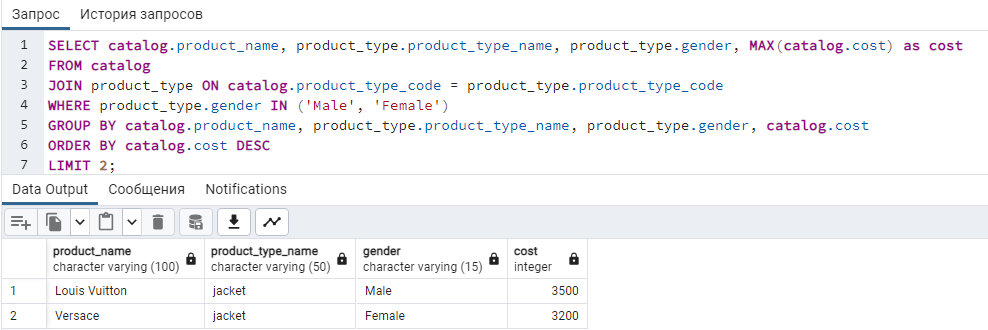


Рисунок 4: Выполнение запроса 4

Запрос 5. Выведите номер, дату и общую стоимость всех заказов, в которых присутствует товар из категории "платье".

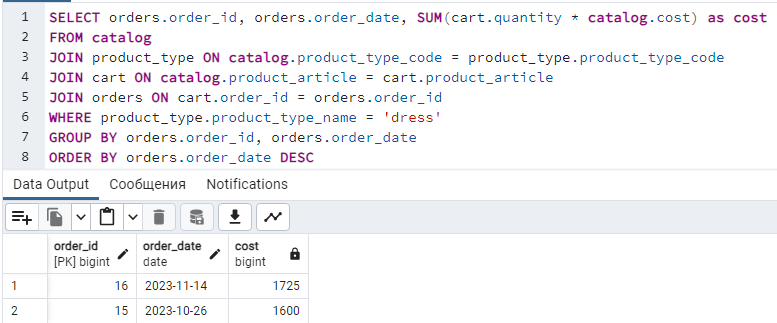


Рисунок 5: Выполнение запроса 5

Запрос 6. Выведите ФИО и сумму заказов всех покупателей, которые никогда не делали заказов на сумму более 1500 рублей.

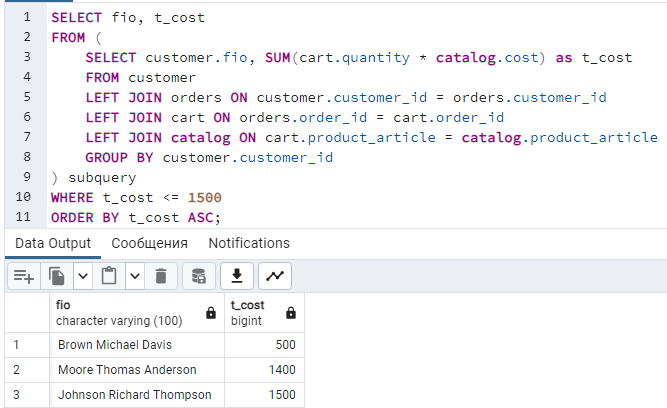


Рисунок 6: Выполнение запроса 6

Запрос 7. Выведите название, категорию и количество проданных единиц всех товаров, которые были проданы хотя бы 1 раз в прошлом месяце.

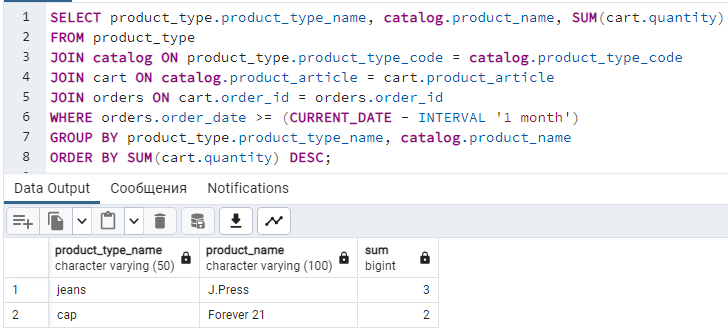


Рисунок 7: Выполнение запроса 7

Запрос 8. Выведите название и общую стоимость всех товаров категории, в которых общая стоимость всех товаров превышает 3000 рублей.

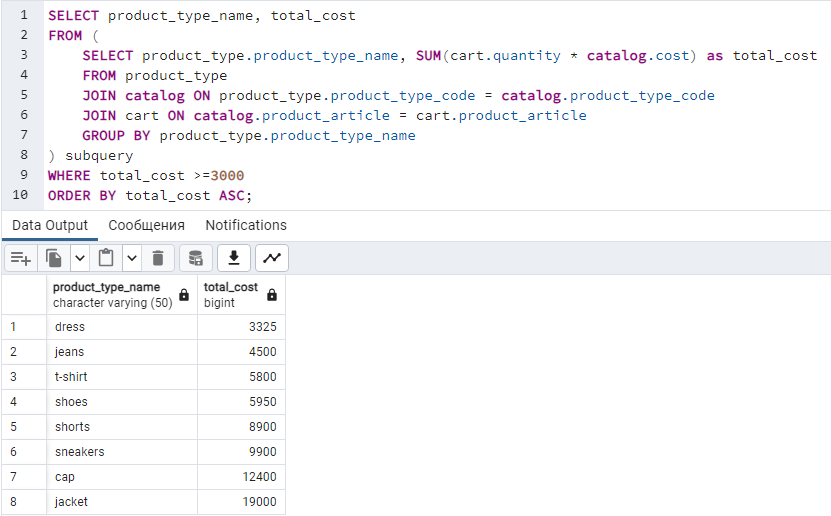


Рисунок 8: Выполнение запроса 8

Запрос 9. Выведите номер, дату и общую стоимость всех заказов, в которых присутствуют товары с ценой более 800 рублей.

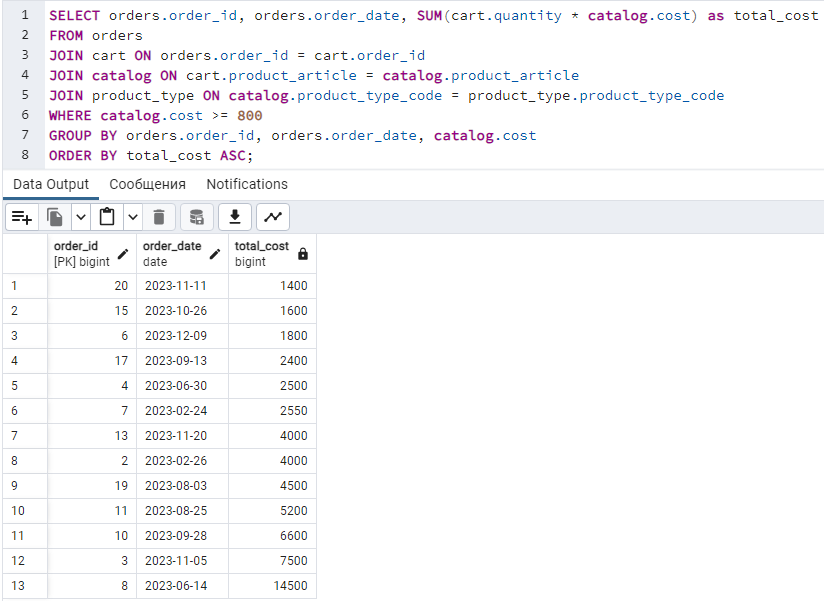


Рисунок 9: Выполнение запроса 9

Запрос 10. Выведите ФИО, количество заказов и общую стоимость заказов покупателей, которые делали заказы в прошлом месяце.

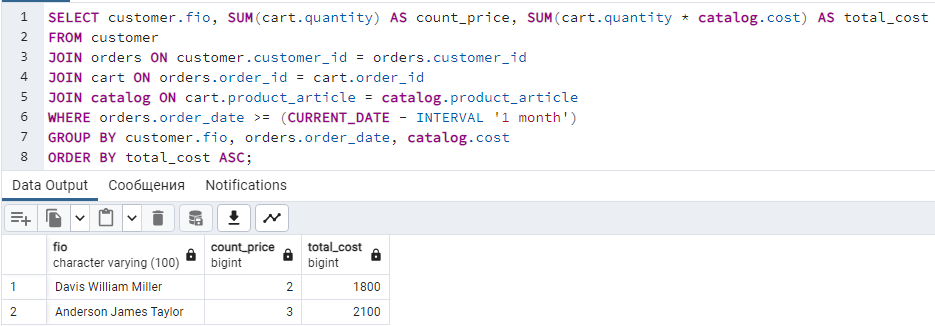


Рисунок 10: Выполнение запроса 10

# ЗАКЛЮЧЕНИЕ

При написании курсовой работы была проанализирована предметная область, связанная с организацией работы Интернет-магазина одежды, были построены логическая и физическая структуры базы данных, было выполнено заполнение базы данных тестовыми данными и были разработаны SQL-запросы для вывода необходимой информации. В результате выполнения курсовой работы были закреплены теоретические основы и практические навыки проектирования структуры базы данных.

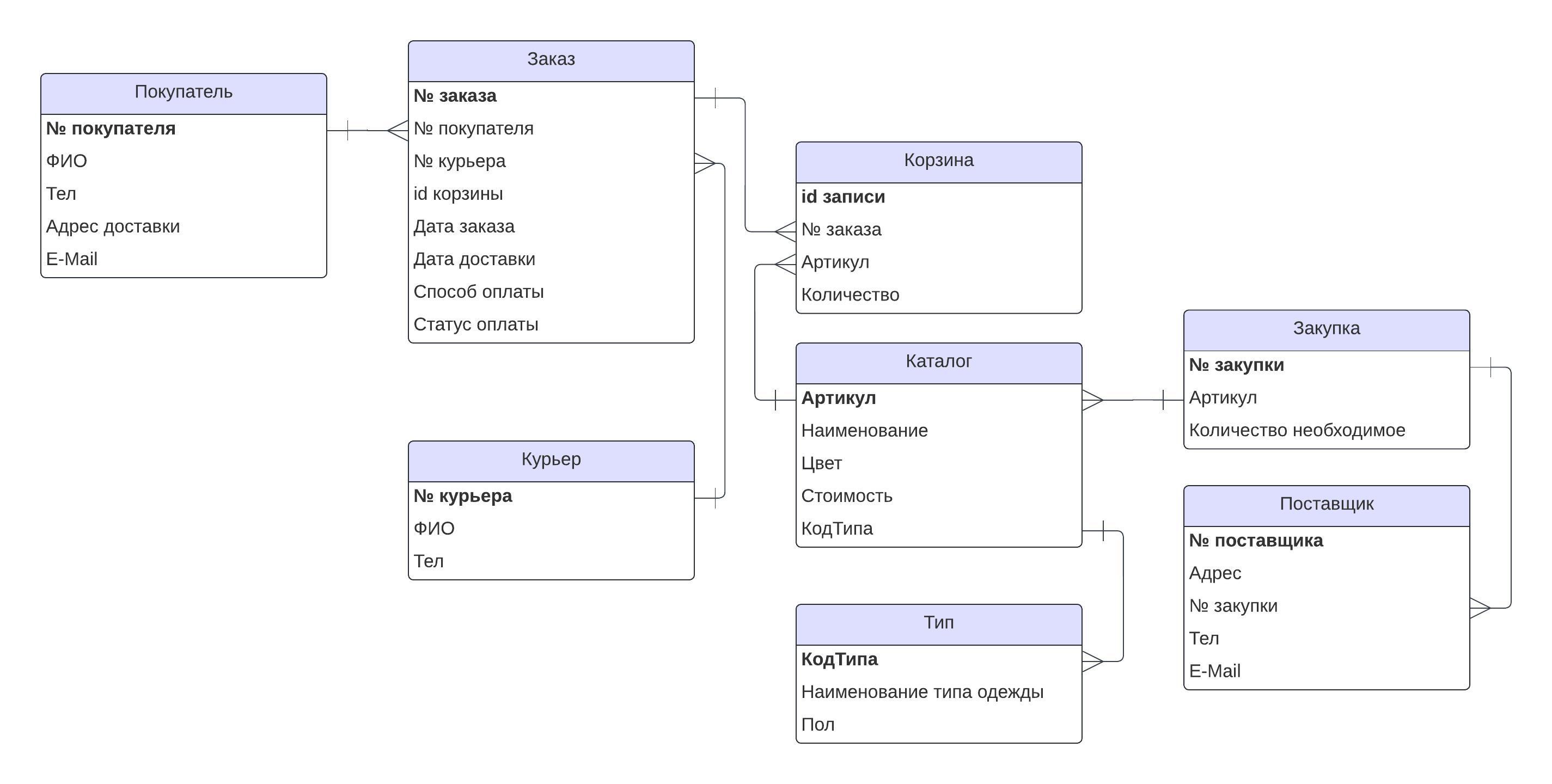
# СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Postgrespro. Документация к PostgreSQL 15 [Электронный ресурс] – Режим доступа: <https://postgrespro.ru/docs/postgresql/15/datatype-datetime>, свободный. – (дата обращения 24.12.2023)
2. Википедия. Контрагент [Электронный ресурс] – Режим доступа: https://ru.wikipedia.org/wiki/Контрагент, свободный. – (дата обращения 24.12.2023)
3. Кафедра информационных и коммуникационных технологий РГПУ им.А.И.Герцена. Лекционный материал [Электронный ресурс]. – Режим доступа: <http://flash-library.narod.ru/Ch-Informatics/lektion/lektion7.html>, свободный. – (дата обращения 24.12.2023)
4. Лекциопедия. Области применения баз данных [Электронный ресурс] – Режим доступа: <https://lektsiopedia.org/lek-10364.html>, свободный. – (дата обращения 24.12.2023)
5. Карпов А.П. Область применения баз данных [Электронный ресурс] – Режим доступа: <https://scienceforum.ru/2016/article/2016020613>, свободный. – (дата обращения 24.12.2023)
6. Оскерко В.С. Проектирование базы данных [Электронный ресурс] – Режим доступа: http://bseu.by/it/tohod/lekcii4\_3.htm, свободный. – (дата обращения 24.12.2023)
7. Amazon Web Services. Реляционные базы данных [Электронный ресурс] – Режим доступа: <https://aws.amazon.com/ru/relational-database/>, свободный. – (дата обращения 24.12.2023)
8. Блог web-программиста. Что такое базы данных и для чего они используются [Электронный ресурс] – Режим доступа: <http://juice-health.ru/programming/419-chto-takoe-bazy-dannykh>, свободный. – (дата обращения 24.12.2023)

# Приложение 1. Схема взаимодействий



# Приложение 2. ER-диаграмма



# Приложение 3. Листинг кода SQL

-- Таблица Покупатель

CREATE TABLE customer (

customer\_id BIGSERIAL PRIMARY KEY,

fio VARCHAR(100) NOT NULL,

tel BIGINT NOT NULL,

delivery\_address VARCHAR(100) NOT NULL,

email VARCHAR(100) NOT NULL

);

INSERT INTO customer (fio, tel, delivery\_address, email)

VALUES

('Smith John Doe', 7308513295, 'Moscow, Tverskaya St, 10', 'john.doe@example.com'),

('Johnson Peter Smith', 6312019579, 'St. Petersburg, Nevsky Prospect, 20', 'peter.smith@example.com'),

('Williams James Brown', 8903035555, 'Novosibirsk, Lenin Ave, 30', 'james.brown@example.com'),

('Brown Michael Davis', 2014553472, 'Yekaterinburg, 8 Marta St, 40', 'michael.davis@example.com'),

('Davis William Miller', 5758831489, 'Nizhny Novgorod, Bolshaya Pokrovskaya St, 50', 'william.miller@example.com'),

('Miller David Wilson', 1142831743, 'Samara, Kuibyshev St, 60', 'david.wilson@example.com'),

('Wilson Robert Johnson', 3596613764, 'Kazan, Bauman St, 70', 'robert.johnson@example.com'),

('Johnson Richard Thompson', 5848019961, 'Chelyabinsk, Kirovka St, 80', 'richard.thompson@example.com'),

('Thompson James White', 7034337405, 'Omsk, Lenin Ave, 90', 'james.white@example.com'),

('White Thomas Harris', 4812441483, 'Rostov-on-Don, Bolshaya Sadovaya St, 100', 'thomas.harris@example.com'),

('Harris Charles Martin', 4914134361, 'Ufa, Pushkin St, 110', 'charles.martin@example.com'),

('Martin Donald Clark', 2106416654, 'Volgograd, Lenin Ave, 120', 'donald.clark@example.com'),

('Clark Scott Lewis', 3512329772, 'Perm, Mira St, 130', 'scott.lewis@example.com'),

('Lewis Michael Young', 1778922497, 'Vladivostok, Svetlanskaya St, 140', 'michael.young@example.com'),

('Young Kevin Green', 2707591323, 'Saratov, Moskovskaya St, 150', 'kevin.green@example.com'),

('Green James Black', 3242709615, 'Krasnoyarsk, Lenin Ave, 160', 'james.black@example.com'),

('Black William Gray', 4441191607, 'Irkutsk, Karl Marx St, 170', 'william.gray@example.com'),

('Gray David Moore', 2168794298, 'Yaroslavl, Volkov St, 180', 'david.moore@example.com'),

('Moore Thomas Anderson', 2011554973, 'Krasnodar, Krasnaya St, 190', 'thomas.anderson@example.com'),

('Anderson James Taylor', 5777923229, 'Voronezh, Lenin Ave, 200', 'james.taylor@example.com');

-- Таблица Курьер

CREATE TABLE courier (

courier\_id BIGSERIAL PRIMARY KEY,

fio VARCHAR(100) NOT NULL,

tel BIGINT NOT NULL

);

INSERT INTO courier (fio, tel)

VALUES

('Lissie McChesney', 1685454012),

('Sergei Lesurf', 8491154022),

('Ricca Patemore', 7924511106),

('Ruthann Drepp', 5357416456),

('Mitzi Teodorski', 7351269259);

-- Таблица Тип одежды

CREATE TABLE product\_type (

product\_type\_code BIGSERIAL PRIMARY KEY,

product\_type\_name VARCHAR(50) NOT NULL,

gender VARCHAR(15) NOT NULL

);

INSERT INTO product\_type (product\_type\_name, gender)

VALUES

('t-shirt', 'Female'),

('dress', 'Female'),

('jacket', 'Female'),

('shoes', 'Female'),

('shorts', 'Female'),

('hat', 'Female'),

('jeans', 'Female'),

('sneakers', 'Female'),

('cap', 'Female'),

('t-shirt', 'Male'),

('jacket', 'Male'),

('shoes', 'Male'),

('shorts', 'Male'),

('hat', 'Male'),

('jeans', 'Male'),

('sneakers', 'Male'),

('cap', 'Male');

-- Таблица Каталог товаров

CREATE TABLE catalog (

product\_article INT PRIMARY KEY,

product\_name VARCHAR(100) NOT NULL,

color VARCHAR(50) NOT NULL,

cost INT NOT NULL,

product\_type\_code INT NOT NULL REFERENCES product\_type(product\_type\_code)

);

INSERT INTO catalog (product\_name, color, cost, product\_type\_code)

VALUES

('Gucci','Black',2500,1),

('Prada','White',3000,1),

('Chanel','Red',2800,4),

('Louis Vuitton','Blue',3500,11),

('Dolce & Gabbana','Green',2700,5),

('Burberry','Yellow',2900,11),

('Versace','Pink',3200,3),

('Armani','Gray',2600,5),

('Fendi','Brown',3100,16),

('Saint Laurent','Beige',2400,15),

('Michael Kors','Orange',2200,17),

('Tommy Hilfiger','Navy',2000,12),

('Calvin Klein','Maroon',1800,16),

('Ralph Lauren','Olive',1900,15),

('Hugo Boss','Purple',2100,7),

('Kate Spade','Teal',1700,11),

('Coach','Lavender',1600,2),

('Tory Burch','Gold',1500,13),

('Valentino','Silver',1400,5),

('Hermes','Bronze',1300,16),

('Zara','Coral',1200,12),

('H&M','Turquoise',1100,14),

('Uniqlo','Magenta',1000,10),

('Forever 21','Cobalt',900,17),

('Topshop','Lime',800,9),

('Mango','Aqua',700,8),

('Esprit','Ivory',600,10),

('Gap','Chocolate',500,8),

('Banana Republic','Crimson',550,8),

('J.Crew','Cerulean',575,2),

('Ann Taylor','Periwinkle',600,5),

('Brooks Brothers','Mustard',650,4),

('J.Press','Sage',700,7),

('Club Monaco','Scarlet',750,10),

('Theory','Plum',800,4),

('Alice + Olivia','Peacock',850,16),

('Rag & Bone','Mint',900,11),

('Jason Wu','Rose',950,12);

-- Таблица Заказ

CREATE TABLE orders (

order\_id BIGSERIAL PRIMARY KEY,

customer\_id INT NOT NULL REFERENCES customer(customer\_id),

courier\_id INT NOT NULL REFERENCES courier(courier\_id),

cart\_id INT NOT NULL,

order\_date DATE NOT NULL,

delivery\_date DATE NOT NULL,

payment\_method VARCHAR(20) NOT NULL,

payment\_status VARCHAR(20) NOT NULL

);

INSERT INTO orders (customer\_id, courier\_id, cart\_id, order\_date, delivery\_date, payment\_method, payment\_status)

VALUES

(1, 4, 15, '26.02.2023', '01.03.2023', 'cash', 'paid'),

(2, 1, 7, '05.11.2023', '21.11.2023', 'card', 'paid'),

(3, 2, 19, '30.06.2023', '20.07.2023', 'cash', 'paid'),

(4, 5, 17, '04.03.2023', '16.03.2023', 'card', 'paid'),

(5, 2, 20, '09.12.2023', '26.12.2023', 'cash', 'paid'),

(6, 1, 3, '24.02.2023', '02.03.2023', 'card', 'unpaid'),

(7, 5, 7, '14.06.2023', '05.07.2023', 'cash', 'paid'),

(8, 3, 1, '10.04.2023', '16.04.2023', 'card', 'paid'),

(9, 2, 19, '28.09.2023', '02.10.2023', 'cash', 'paid'),

(10, 4, 10, '25.08.2023', '08.09.2023', 'card', 'paid'),

(11, 5, 6, '28.05.2023', '04.06.2023', 'card', 'unpaid'),

(12, 2, 10, '20.11.2023', '29.11.2023', 'card', 'unpaid'),

(13, 3, 5, '21.04.2023', '30.04.2023', 'card', 'paid'),

(14, 3, 6, '26.10.2023', '07.11.2023', 'card', 'paid'),

(15, 3, 20, '14.11.2023', '19.11.2023', 'cash', 'paid'),

(16, 1, 12, '13.09.2023', '22.09.2023', 'cash', 'unpaid'),

(17, 1, 9, '16.05.2023', '28.05.2023', 'cash', 'paid'),

(18, 2, 4, '03.08.2023', '21.08.2023', 'card', 'paid'),

(19, 3, 16, '11.11.2023', '02.12.2023', 'card', 'unpaid'),

(20, 3, 6, '21.12.2023', '31.12.2023', 'cash', 'unpaid');

-- Таблица Корзина

CREATE TABLE cart (

cart\_id BIGSERIAL PRIMARY KEY,

order\_id INT NOT NULL REFERENCES orders(order\_id),

product\_article INT NOT NULL REFERENCES catalog(product\_article),

quantity INT NOT NULL

);

INSERT INTO cart (order\_id, product\_article, quantity)

VALUES

(1, 33, 3),

(2, 12, 2),

(3, 18, 5),

(4, 1, 1),

(5, 28, 1),

(6, 24, 2),

(7, 36, 3),

(8, 6, 5),

(9, 34, 2),

(10, 11, 3),

(11, 20, 4),

(12, 29, 3),

(13, 25, 5),

(14, 27, 3),

(15, 21, 1),

(16, 30, 3),

(17, 10, 1),

(18, 32, 3),

(19, 37, 5),

(20, 19, 1);

-- Таблица Закупка

CREATE TABLE purchase (

purchase\_id BIGSERIAL PRIMARY KEY,

product\_article INT NOT NULL REFERENCES catalog(product\_article),

required\_quantity INT NOT NULL

);

INSERT INTO purchase (product\_article, required\_quantity)

VALUES

(10, 125),

(31, 125),

(4, 110),

(26, 131),

(17, 131),

(7, 126),

(33, 176),

(37, 104),

(11, 173),

(12, 122),

(14, 118);

-- Таблица Поставщик

CREATE TABLE supplier (

supplier\_id BIGSERIAL PRIMARY KEY,

address VARCHAR(100) NOT NULL,

purchase\_id INT NOT NULL REFERENCES purchase(purchase\_id),

tel BIGINT NOT NULL,

email VARCHAR(100) NOT NULL

);

INSERT INTO supplier (address, purchase\_id, tel, email)

VALUES

('Stockholm', 3, '9218325155', 'slinden0@tumblr.com'),

('Jijia', 11, '9009244246', 'bhallad1@miitbeian.gov.cn'),

('Huayang', 1, '8648599800', 'cweekland2@jalbum.net'),

('Liquica', 9, '2783473773', 'ltellenbrok3@kickstarter.com'),

('Tocota', 5, '8986435736', 'fethridge4@independent.co.uk'),

('Sidu', 10, '8603230543', 'fphippard5@marriott.com'),

('Moussoro', 2, '2477225201', 'jdowdall6@seesaa.net'),

('Trzebiatów', 6, '4599941415', 'estert7@home.pl'),

('San Rafael', 4, '3657040223', 'egrunwall8@google.com.au'),

('Blois', 7, '3766641000', 'tsibbald9@ftc.gov');