**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«ТЮМЕНСКИЙ индустриальный университет»**

Высшая школа цифровых технологий

Кафедра Математики и прикладных информационных систем

**КУРСОВАЯ РАБОТА**

**Дисциплина:** «Системы хранения и обработки данных»

**Тема:** «Разработка СУБД по предметной области

«Магазин компьютерных комплектующих» на основе PostgreSQL»

**Выполнили:**

Студент 1 курса группы МОмз-23-1

направления 01.04.02 «Прикладная математика и информатика»

Дьяченко Д.А.

**Руководитель:**

Доцент, канд. пед. наук

Спирин И.С.

Тюмень, 2024 г.

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

Для реализации курсовой работы необходимо решить ряд задач:

* провести анализ предметной области: Магазин компьютерных комплектующих;
* произвести выбор и обоснование технологии проектирования для БД «Магазин компьютерных комплектующих»;
* определить сущности, виды и типы связей между ними;
* произвести проектирование информационного обеспечения, построить концептуальную модель, инфологическую модель, физическую модель данных;
* произвести проектирование технологического процесса обработки информации;
* оценить эффективность БД по деятельности бара.

**АННОТАЦИЯ**

Курсовая работа содержит 33 страницы, 12 источников, 21 рисунок

Ключевые слова: база данных, сущности, связи, схема данных, таблицы, запросы, формы, отчеты.

Цель работы – разработка базы данных «Создание базы данных в PostgreSQL» для «Магазина компьютерных комплектующих».

Объект исследования – автоматизированная информационная система «Магазин компьютерных комплектующих».

Предметом является процесс разработки СУБД по выбранной предметной области.

Результатом работы является программа – база данных «Магазин компьютерных комплектующих», которая показывает возможности реляционной базы PostgreSQL.

Курсовая работа выполнена в текстовом редакторе «Microsoft Office Word 2016» и представлена на компакт-диске (в конверте на обороте обложки).

Код работы расположен здесь:

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 5](#__RefHeading___Toc5119_2352249077)

[1 Анализ предметной области 7](#__RefHeading___Toc5121_2352249077)

[1.1 Основные понятия 7](#__RefHeading___Toc5123_2352249077)

[1.2 Анализ существующих программных обеспечений 10](#__RefHeading___Toc5125_2352249077)

[1.3 Анализ предметной области 11](#__RefHeading___Toc5127_2352249077)

[2 Разработка моделей базы данных 13](#__RefHeading___Toc5129_2352249077)

[2.1 Концептуальная модель 13](#__RefHeading___Toc5131_2352249077)

[2.2 Инфологическая модель 16](#__RefHeading___Toc5133_2352249077)

[2.3 Физическая модель 20](#__RefHeading___Toc5135_2352249077)

[3 Практическая реализация базы данных 23](#__RefHeading___Toc5137_2352249077)

[3.1 Создание таблиц и их заполнение информацией 23](#__RefHeading___Toc5139_2352249077)

[3.2 Создание запросов 27](#__RefHeading___Toc5141_2352249077)

[3.3 Создание отчетов 29](#__RefHeading___Toc5143_2352249077)

[ЗАКЛЮЧЕНИЕ 31](#__RefHeading___Toc5145_2352249077)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 32](#__RefHeading___Toc5147_2352249077)

# ВВЕДЕНИЕ

В настоящее время в современном мире электронных технологий практически невозможно представить компанию (фирму или организацию), в которой не требуется обработка некоторого объема информации. Информацию требуется, где-то хранить. Информация может динамически изменяться. Регулярно требуется выборка данных по определенным критериям из всего массива. Базы данных создаются специально для хранения, обработки, проведения расчетов, сортировки, выборки и представления любых массивов данных по любым критериям, и поэтому базы данных очень актуальны, потому что объемы хранимой информации растут с каждым днем.

Курсовая работа посвящается реляционным базам данных. В данной работе будет разрабатываться база данных «Магазин компьютерных комплектующих».

Объектом данной работы является автоматизированная информационная система «Магазин компьютерных комплектующих».

Предметом является процесс разработки СУБД по выбранной предметной области.

Целью является проектирование и разработка базы данных «Магазин компьютерных комплектующих».

Для достижения поставленной цели необходимо решить следующие задачи:

* охарактеризовать предметную область;
* описать основные объекты предметной области;
* провести анализ поставленной задачи;
* разработать концептуальную модель;
* спроектировать инфологическую модель;
* разработать физическую модель;
* создать таблицы и заполнить их;
* разработать запросы;
* сформировать отчеты;

Работа состоит из введения, трех глав, заключения и списка литературы.

В первой главе проводится анализ терминов, технологий и предметной области «Магазина компьютерных комплектующих».

Во второй главе разрабатываются модели базы данных: концептуальная модель, инфологическая модель и физическая модель.

В третьей главе рассматривается методика создания базы данных: создание таблиц и их заполнение данными и информацией, создание запросов, создание отчетов.

# Анализ предметной области

## Основные понятия

Разработка базы данных для магазина компьютерных комплектующих требует глубокого понимания ряда ключевых понятий, которые лежат в основе реляционных систем управления базами данных. В этой главе мы рассмотрим такие понятия, как базы данных, реляционные базы данных, схемы данных, сущности, атрибуты, связи, таблицы, запросы, формы, отчеты и макросы. Понимание этих терминов поможет эффективно проектировать и реализовывать нашу базу данных.

База данных представляет собой организованную коллекцию данных, которая хранится и управляется с помощью системы управления базами данных (СУБД). Базы данных позволяют систематизировать информацию таким образом, чтобы её можно было легко искать, обновлять и анализировать. В контексте магазина компьютерных комплектующих, база данных будет включать информацию о товарах, сотрудниках, поставщиках, продажах и услугах.

Реляционная база данных – это тип базы данных, в которой данные организованы в виде таблиц, называемых также реляциями. Каждая таблица состоит из строк и столбцов, где строки представляют отдельные записи, а столбцы – атрибуты этих записей. Реляционные базы данных основаны на реляционной модели данных, предложенной Эдгаром Ф. Коддом в 1970 году. PostgreSQL, выбранная нами СУБД, является ярким примером реляционной базы данных.

Схема данных – это структура, определяющая организацию данных в базе данных. Схема включает описание таблиц, их атрибутов и связей между ними. Разработка схемы данных является важным этапом в проектировании базы данных, так как она определяет, каким образом данные будут храниться и как они будут взаимосвязаны. В нашем случае схема данных будет включать таблицы для сотрудников, комплектующих, услуг, продаж, поставщиков и поставок.

Сущность – это объект или вещь, о которой необходимо хранить информацию в базе данных. Сущности могут быть как физическими объектами (например, комплектующие или сотрудники), так и абстрактными понятиями (например, продажа или услуга). В реляционной базе данных каждая сущность представляется отдельной таблицей. Например, сущность "Сотрудник" будет представлена таблицей Employee, а сущность "Комплектующие" – таблицей Component.

Атрибуты – это характеристики или свойства, которые описывают сущности. В таблице базы данных атрибуты соответствуют столбцам. Например, таблица Employee будет иметь такие атрибуты, как id, first\_name, last\_name, position, email и phone. Каждый атрибут должен иметь определенный тип данных (например, строка, целое число, дата), который определяет, какие значения могут быть сохранены в этом столбце.

Связи (отношения) определяют, каким образом сущности взаимосвязаны друг с другом. В реляционных базах данных связи устанавливаются с помощью ключей. Первичный ключ (primary key) – это уникальный идентификатор записи в таблице, который позволяет однозначно идентифицировать каждую запись. Внешний ключ (foreign key) – это атрибут, который указывает на первичный ключ другой таблицы, создавая связь между двумя таблицами. Например, в таблице Sale внешний ключ employee\_id указывает на первичный ключ id таблицы Employee, связывая продажи с конкретными сотрудниками.

Таблицы – это основные структурные элементы реляционной базы данных, в которых хранятся данные. Каждая таблица состоит из строк и столбцов, где строки представляют отдельные записи, а столбцы – атрибуты этих записей. Таблицы могут быть связаны между собой с помощью первичных и внешних ключей, что позволяет организовать данные в структурированную и взаимосвязанную систему. В нашей базе данных таблицы будут включать Employee, Component, Service, Sale, Supplier и Supply.

Запросы – это инструкции, которые используются для извлечения, обновления, удаления и добавления данных в базу данных. Запросы на языке SQL (Structured Query Language) позволяют выполнять различные операции с данными, такие как выборка записей, фильтрация, сортировка и агрегирование данных. Например, запрос может извлечь все продажи, совершенные за определенный период, или обновить информацию о наличии комплектующих на складе. В PostgreSQL запросы играют ключевую роль в управлении и анализе данных.

Отчеты – это форматированные представления данных, которые используются для анализа и принятия решений. Отчеты могут включать таблицы, диаграммы и графики, отображающие данные в удобочитаемом виде. В контексте нашего проекта отчеты могут быть использованы для анализа продаж, оценки эффективности работы сотрудников и отслеживания поставок. Создание отчетов позволяет систематизировать и визуализировать данные, предоставляя пользователям ценные инсайты.

Понимание этих основных понятий является необходимым условием для успешного проектирования и разработки базы данных для магазина компьютерных комплектующих. Грамотно спроектированная база данных обеспечивает надежное хранение, быстрый доступ и эффективное управление данными, что в свою очередь способствует повышению эффективности работы магазина и улучшению качества обслуживания клиентов.

## Анализ существующих программных обеспечений

Реляционные базы данных представляют собой основу для хранения и управления структурированными данными в большинстве современных информационных систем. Среди популярных решений для создания реляционных баз данных можно выделить MySQL, PostgreSQL и Microsoft Access, каждое из которых обладает своими особенностями и преимуществами.

MySQL – одна из самых распространенных систем управления базами данных с открытым исходным кодом. Она используется во многих веб-приложениях благодаря своей простоте, высокой производительности и надежности. MySQL поддерживает все основные типы данных и операции, необходимые для создания сложных приложений, и отличается хорошей масштабируемостью. Это делает её подходящей для небольших и средних проектов, где важны скорость и эффективность.

PostgreSQL – еще одна мощная реляционная система управления базами данных с открытым исходным кодом, известная своей надежностью и расширяемостью. Она поддерживает широкий спектр типов данных, включая JSON, XML и пользовательские типы данных. PostgreSQL обладает развитой системой управления транзакциями, что обеспечивает высокую целостность данных. Эта СУБД активно используется в проектах, требующих сложных операций с данными и высокой степени надежности, таких как финансовые системы и крупные корпоративные приложения.

Microsoft Access – настольная система управления базами данных, входящая в пакет Microsoft Office. Она предназначена для быстрого создания и управления небольшими базами данных, интеграции с другими приложениями Microsoft и предоставления простого интерфейса для пользователей без глубоких знаний программирования. Access широко используется в малых предприятиях и для личных проектов, где важны удобство и возможность работы без сложной настройки серверов и сетевой инфраструктуры.

Сравнивая эти решения, мы пришли к выводу, что для нашего проекта, посвященного разработке базы данных для магазина компьютерных комплектующих, наилучшим выбором является PostgreSQL. Это решение было принято на основе ряда критериев, таких как надежность, поддержка сложных типов данных и транзакций, а также возможность масштабирования. PostgreSQL предлагает оптимальное сочетание функциональности и производительности, что делает его идеальным выбором для создания мощной и устойчивой к ошибкам информационной системы, способной справляться с большими объемами данных и сложными запросами.

## Анализ предметной области

Предметной областью данной курсовой работы является автоматизированная информационная система для магазина компьютерных комплектующих. В современном мире, где информационные технологии играют ключевую роль в успешной деятельности бизнеса, магазины компьютерных комплектующих требуют надежной системы управления для обеспечения эффективного функционирования и высокого уровня обслуживания клиентов.

Основная задача такой системы – управление ассортиментом товара, что включает в себя контроль наличия комплектующих на складе, обновление информации о ценах и характеристиках товаров. Важно, чтобы система позволяла своевременно пополнять запасы, избегая как дефицита, так и избыточных запасов, которые могут устареть и потерять свою актуальность.

Управление продажами – еще один важный аспект. Информационная система должна учитывать все продажи, создавая отчеты и проводя анализ по различным критериям, таким как наиболее популярные товары, временные пики продаж и эффективность работы каждого сотрудника. Это позволяет принимать обоснованные решения для увеличения продаж и улучшения качества обслуживания клиентов.

Эффективное управление поставками также является критически важным. Взаимодействие с поставщиками, учет поступивших товаров и контроль за своевременностью поставок способствуют поддержанию оптимального уровня запасов и снижению затрат на хранение. Надежная система управления позволяет быстро и точно отслеживать каждую поставку, включая информацию о поставщиках, количестве и стоимости товаров.

Управление персоналом включает в себя хранение информации о сотрудниках, контроль за их деятельностью и учет данных о продажах, проведенных каждым сотрудником. Это помогает не только в администрировании, но и в мотивации персонала через систему бонусов и премий за достигнутые результаты.

Обслуживание клиентов в магазине компьютерных комплектующих включает предоставление различных услуг, таких как установка и настройка комплектующих. Информационная система должна учитывать заказы на эти услуги, отслеживать их выполнение и обеспечивать высокое качество обслуживания. Это способствует созданию лояльной клиентской базы и повышению репутации магазина.

# Разработка моделей базы данных

## Концептуальная модель

Концептуальная модель базы данных представляет собой высокоуровневое описание структуры данных и взаимоотношений между ними. В данном разделе мы рассмотрим основные сущности и их связи, которые будут включены в базу данных для магазина компьютерных комплектующих. Концептуальная модель помогает визуализировать, как данные будут организованы и как они будут взаимодействовать друг с другом.

Основные сущности в нашей базе данных включают:

Employee (Сотрудники): хранит информацию о сотрудниках магазина, включая их имена, должности, контактные данные и уникальные идентификаторы. Это позволяет отслеживать, кто из сотрудников занимается продажами и предоставляет услуги.

Component (Комплектующие): содержит информацию о компьютерных комплектующих, таких как процессоры, видеокарты, материнские платы и т.д. Для каждого компонента хранятся его название, описание, цена и количество на складе.

Service (Услуги): описывает услуги, предоставляемые магазином, например, установка процессора, замена видеокарты или сборка компьютера. Для каждой услуги указаны её название, описание и стоимость.

Sale (Продажи): представляет информацию о продажах, включая данные о сотруднике, который совершил продажу, услуге или комплектующем, дате продажи и общей стоимости. Эта таблица позволяет отслеживать все транзакции в магазине.

Supplier (Поставщики): хранит данные о поставщиках, от которых магазин получает комплектующие. Это включает название поставщика, телефон и электронную почту.

Supply (Поставки): содержит информацию о поставках комплектующих от поставщиков, включая идентификаторы поставок, комплектующих, поставщиков, даты поставок, количество и общую стоимость поставок. Это позволяет управлять запасами и контролировать поступление товаров.

Концептуальная модель базы данных строится на основе этих сущностей и их взаимосвязей. Связи между сущностями определяются с помощью ключей: первичных ключей (Primary Keys) и внешних ключей (Foreign Keys). Первичный ключ уникально идентифицирует каждую запись в таблице, в то время как внешний ключ указывает на первичный ключ другой таблицы, устанавливая связь между таблицами.

В базе данных магазина компьютерных комплектующих связи между таблицами организованы следующим образом:

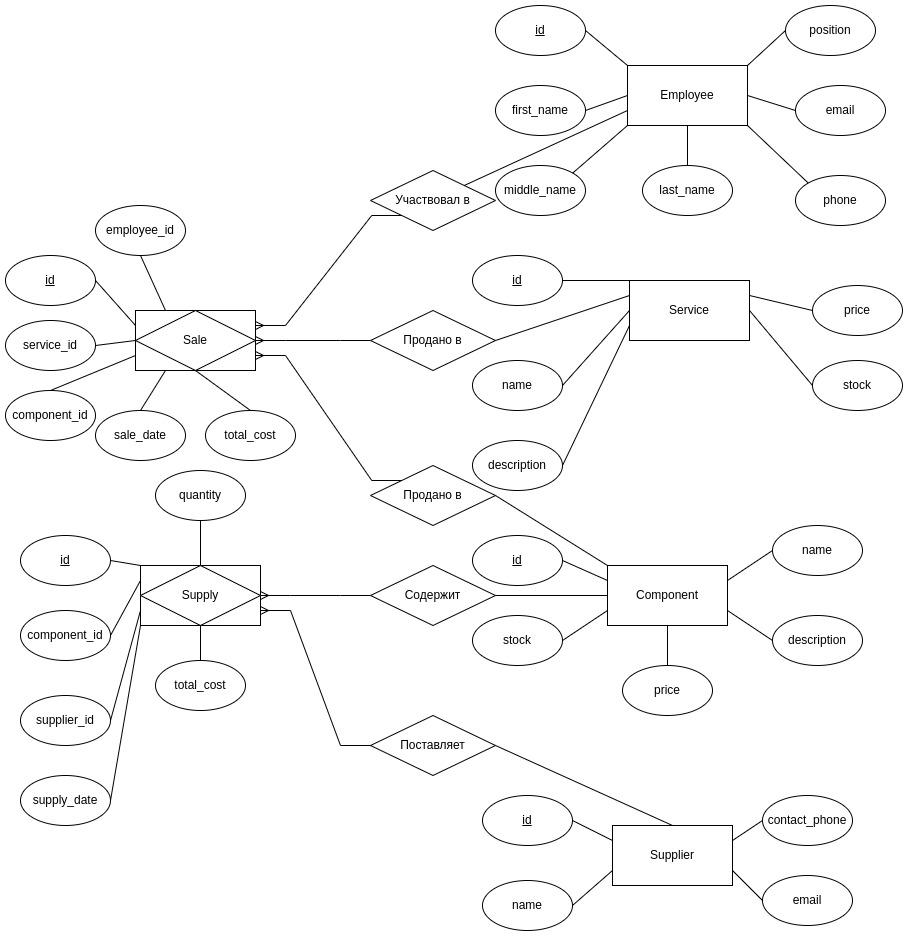
Таблица Sale содержит внешний ключ employee\_id, который ссылается на таблицу Employee, связывая каждую продажу с конкретным сотрудником.

Таблица Sale также содержит внешние ключи service\_id и component\_id, которые ссылаются на таблицы Service и Component соответственно, указывая, какая услуга была предоставлена или какой компонент был продан.

Таблица Supply содержит внешний ключ component\_id, который ссылается на таблицу Component, связывая поставки с конкретными комплектующими.

Таблица Supply также содержит внешний ключ supplier\_id, который ссылается на таблицу Supplier, указывая, какой поставщик осуществил поставку.

Концептуальная модель нашей базы данных представлена на рисунке ниже.

Рисунок 2.1 – Концептуальная модель

На рисунке показаны сущности (таблицы) и связи между ними, что помогает визуализировать структуру базы данных и понять, как данные будут организованы и взаимодействовать друг с другом.

Разработка концептуальной модели является важным этапом в создании базы данных, так как она служит основой для дальнейшего проектирования и реализации инфологической и физической моделей. Четкое понимание концептуальной модели позволяет избежать ошибок и недоразумений в процессе разработки, обеспечивая надежную и эффективную организацию данных в информационной системе магазина компьютерных комплектующих.

## Инфологическая модель

Инфологическая модель базы данных представляет собой детализированное описание структуры данных и взаимосвязей между ними на уровне логической организации. В отличие от концептуальной модели, инфологическая модель более детализирована и включает в себя описание всех атрибутов сущностей и типов данных, а также правил, определяющих взаимодействие данных. Инфологическая модель является переходным этапом между концептуальной моделью и физической реализацией базы данных.

Основные сущности нашей инфологической модели включают Employee, Component, Service, Sale, Supplier и Supply. В данной модели для каждой сущности определяются все атрибуты и их типы данных, а также указываются первичные и внешние ключи.

Сущности и их атрибуты:

* Employee (Сотрудники)
  + id (SERIAL, PRIMARY KEY)
  + first\_name (VARCHAR(50), NOT NULL)
  + middle\_name (VARCHAR(50))
  + last\_name (VARCHAR(50), NOT NULL)
  + position (VARCHAR(50), NOT NULL)
  + email (VARCHAR(100), NOT NULL, UNIQUE)
  + phone (VARCHAR(20))
* Component (Комплектующие)
  + id (SERIAL, PRIMARY KEY)
  + name (VARCHAR(100), NOT NULL)
  + description (TEXT)
  + price (DECIMAL(10, 2), NOT NULL)
  + stock (INT, NOT NULL)
* Service (Услуги)
  + id (SERIAL, PRIMARY KEY)
  + name (VARCHAR(100), NOT NULL)
  + description (TEXT)
  + price (DECIMAL(10, 2), NOT NULL)
* Sale (Продажи)
  + id (SERIAL, PRIMARY KEY)
  + employee\_id (INT, FOREIGN KEY REFERENCES Employee(id))
  + service\_id (INT, FOREIGN KEY REFERENCES Service(id))
  + component\_id (INT, FOREIGN KEY REFERENCES Component(id))
  + sale\_date (TIMESTAMP, DEFAULT CURRENT\_TIMESTAMP)
  + total\_cost (DECIMAL(10, 2), NOT NULL)
* Supplier (Поставщики)
  + id (SERIAL, PRIMARY KEY)
  + name (VARCHAR(100), NOT NULL)
  + contact\_phone (VARCHAR(20))
  + email (VARCHAR(100), NOT NULL, UNIQUE)
* Supply (Поставки)
  + id (SERIAL, PRIMARY KEY)
  + component\_id (INT, FOREIGN KEY REFERENCES Component(id))
  + supplier\_id (INT, FOREIGN KEY REFERENCES Supplier(id))
  + supply\_date (TIMESTAMP, DEFAULT CURRENT\_TIMESTAMP)
  + quantity (INT, NOT NULL)
  + total\_cost (DECIMAL(10, 2), NOT NULL)

Взаимосвязи между сущностями

В инфологической модели взаимосвязи между сущностями определяются с помощью внешних ключей, которые устанавливают отношения между таблицами:

Employee и Sale: Внешний ключ employee\_id в таблице Sale ссылается на первичный ключ id в таблице Employee. Это позволяет связывать каждую продажу с конкретным сотрудником, который её осуществил.

Service и Sale: Внешний ключ service\_id в таблице Sale ссылается на первичный ключ id в таблице Service. Это позволяет связывать продажи с конкретными услугами, которые были предоставлены.

Component и Sale: Внешний ключ component\_id в таблице Sale ссылается на первичный ключ id в таблице Component. Это позволяет связывать продажи с конкретными компонентами, которые были проданы.

Component и Supply: Внешний ключ component\_id в таблице Supply ссылается на первичный ключ id в таблице Component. Это позволяет отслеживать поставки конкретных комплектующих.

Supplier и Supply: Внешний ключ supplier\_id в таблице Supply ссылается на первичный ключ id в таблице Supplier. Это позволяет связывать поставки с конкретными поставщиками.

Инфологическая модель базы данных для магазина компьютерных комплектующих представлена на рисунке ниже.

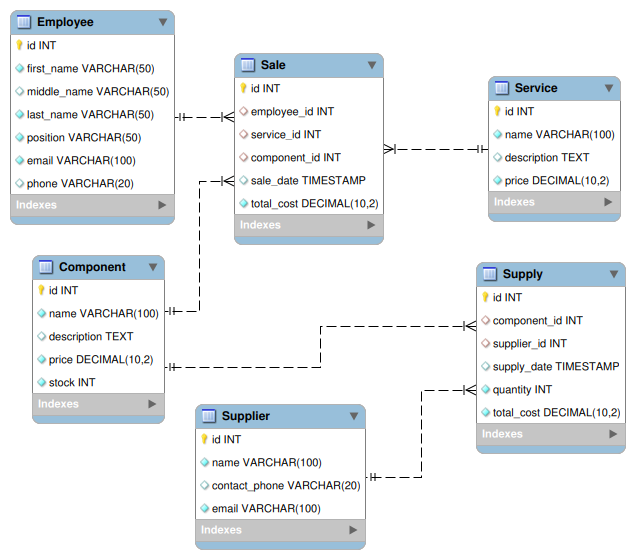


Рисунок 2.2 – Инфологическая модель

Рисунок иллюстрирует структуру таблиц, их атрибуты и взаимосвязи, что помогает визуализировать логическую организацию данных и понять, как данные будут взаимодействовать друг с другом.

Разработка инфологической модели является важным этапом в проектировании базы данных, так как она обеспечивает детальное описание структуры данных, необходимое для дальнейшей физической реализации. Грамотная инфологическая модель позволяет оптимизировать хранение и управление данными, обеспечивая их целостность, согласованность и доступность.

## Физическая модель

Физическая модель базы данных представляет собой описание структуры данных на уровне их физического хранения в системе управления базами данных (СУБД). Она включает в себя детализированное определение таблиц, столбцов, индексов и других объектов базы данных, а также правила их хранения и доступа. В отличие от концептуальной и инфологической моделей, физическая модель отражает технические аспекты реализации базы данных и учитывает особенности конкретной СУБД.

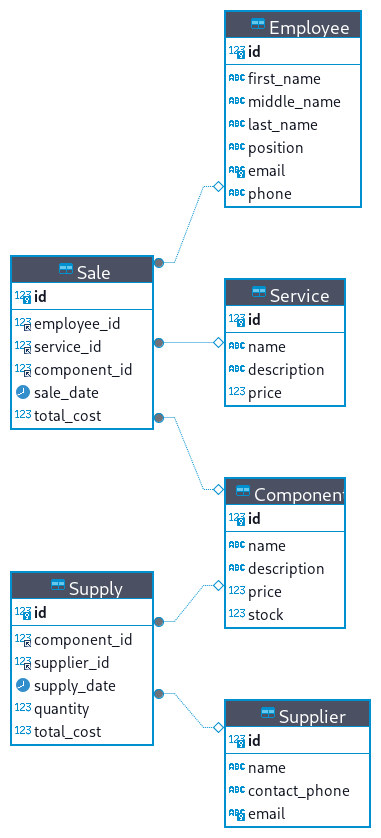
Для реализации базы данных магазина компьютерных комплектующих была выбрана СУБД PostgreSQL. Это решение основано на её высокой производительности, надежности и широких возможностях для работы с большими объемами данных. Физическая модель нашей базы данных включает в себя все необходимые таблицы, атрибуты и связи, описанные в инфологической модели, и определяет, как эти данные будут реально храниться и обрабатываться.

Создание физической модели было выполнено с использованием инструмента DBeaver, который предоставляет мощные средства для работы с базами данных. В DBeaver была автоматически сгенерирована структура таблиц и их атрибутов на основе описанных в инфологической модели данных. Это позволило сократить время на ручное создание таблиц и минимизировать риск ошибок. Сгенерированные таблицы включают все необходимые атрибуты, индексы и ограничения, обеспечивая целостность и оптимальную производительность базы данных.

Процесс генерации физической модели в DBeaver также позволил создать необходимые связи между таблицами с использованием внешних ключей, что обеспечивает корректное взаимодействие данных. Дополнительно были созданы индексы для ускорения выполнения запросов и повышения общей эффективности работы базы данных.

Таким образом, физическая модель представляет собой точное описание того, как данные будут храниться и управляться в PostgreSQL. Она включает в себя все необходимые технические детали для реализации базы данных, обеспечивая её надежность, производительность и удобство использования.

Рисунок ниже иллюстрирует физическую модель базы данных, созданную в Dbeaver.

Рисунок 2.3 – Физическая модель

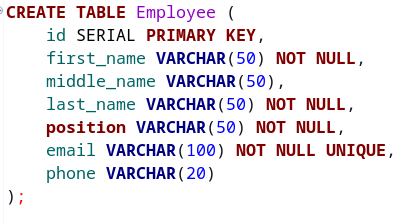
Этот рисунок показывает структуру таблиц, их атрибуты, индексы и связи, что позволяет наглядно представить, как данные организованы на физическом уровне в СУБД PostgreSQL.

# Практическая реализация базы данных

## Создание таблиц и их заполнение информацией

В данном разделе рассматривается процесс создания таблиц и их заполнения информацией для базы данных магазина компьютерных комплектующих. В этом разделе описывается каждая таблица, её структура и данные, которые в неё вставляются.

Таблица Employee предназначена для хранения информации о сотрудниках магазина. В этой таблице содержатся поля для идентификатора сотрудника, его имени, фамилии, отчества, должности, электронной почты и номера телефона.

Рисунок 3.1 – Создание таблицы «Сотрудник»

Для заполнения таблицы Employee используется следующий SQL-запрос:

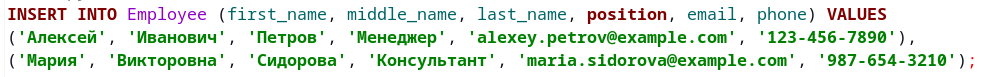
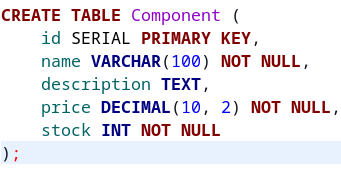
Рисунок 3.2 – Заполнение таблицы «Сотрудник»

Таблица Component хранит информацию о комплектующих, которые продаются в магазине. Поля этой таблицы включают идентификатор комплектующего, его наименование, описание, цену и количество на складе.

Рисунок 3.3 – Создание таблицы «Компонент»

Для заполнения таблицы Component используется следующий SQL-запрос:

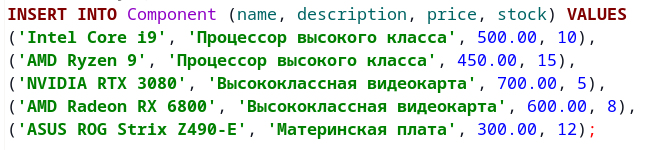
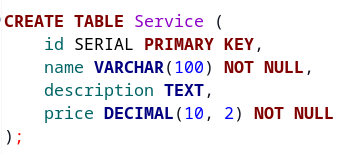
Рисунок 3.4 – Заполнение таблицы «Компонент»

Таблица Service предназначена для хранения информации об услугах, предоставляемых магазином. Поля этой таблицы включают идентификатор услуги, её наименование, описание и цену.

Рисунок 3.5 – Создание таблицы «Услуга»

Для заполнения таблицы Service используется следующий SQL-запрос:

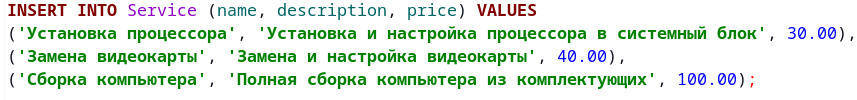
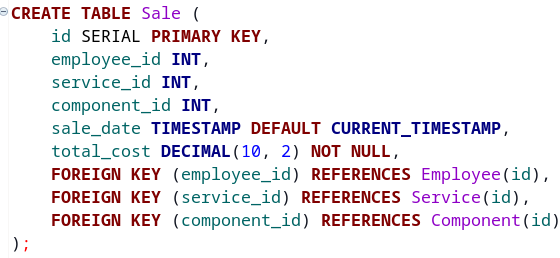
Рисунок 3.6 – Заполнение таблицы «Услуга»

Таблица Sale хранит информацию о продажах в магазине. Поля этой таблицы включают идентификатор продажи, идентификаторы сотрудника, услуги и комплектующего, дату продажи и общую стоимость.

Рисунок 3.7 – Создание таблицы «Продажа»

Для заполнения таблицы Sale используется следующий SQL-запрос:

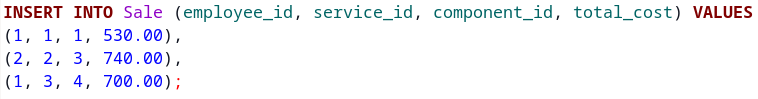
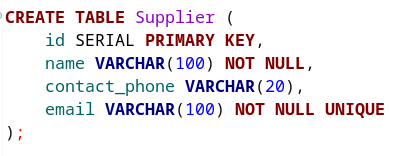
Рисунок 3.8 – Заполнение таблицы «Продажа»

Таблица Supplier хранит информацию о поставщиках комплектующих. Поля этой таблицы включают идентификатор поставщика, его название, контактный телефон и электронную почту.

Рисунок 3.9 – Создание таблицы «Поставщик»

Для заполнения таблицы Supplier используется следующий SQL-запрос:

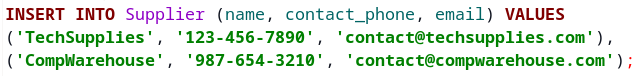
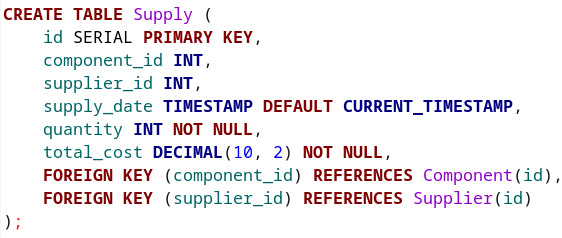
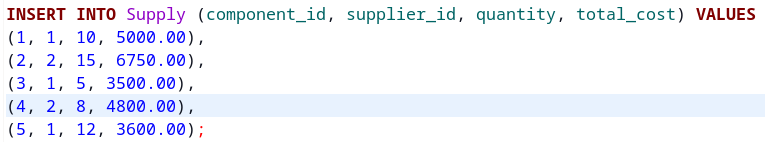


Рисунок 3.10 – Заполнение таблицы «Поставщик»

Таблица Supply хранит информацию о поставках комплектующих. Поля этой таблицы включают идентификатор поставки, идентификаторы комплектующего и поставщика, дату поставки, количество и общую стоимость.

Рисунок 3.11 – Создание таблицы «Поставка»

Для заполнения таблицы Supply используется следующий SQL-запрос:

Рисунок 3.12 – Заполнение таблицы «Поставка»

## Создание запросов

Для работы с базой данных магазина компьютерных комплектующих необходимо уметь выполнять различные запросы для получения, обновления и удаления данных. В данной главе рассмотрим ранее не затронутые примеры запросов CRUD (Create, Read, Update, Delete) к четырем разным таблицам: Employee, Component, Service и Sale.

Для удаления информации о поставке из таблицы Supply, предположим, что мы хотим удалить все записи, где количество (quantity) поставки меньше 10:



Рисунок 3.13 – Запрос удаления

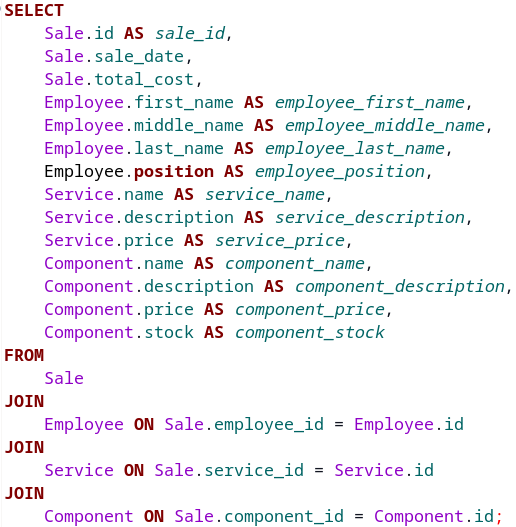
Этот запрос удалит все записи из таблицы Supply, где количество поставки меньше 10.

Для обновления информации о комплектующем в таблице Component, например, уменьшим количество (stock) для компонента с именем "AMD Ryzen 9" на 2 единицы:

Рисунок 3.14 – Запрос изменения

Этот запрос изменит количество компонентов AMD Ryzen 9 на складе, уменьшив его на 2 единицы.

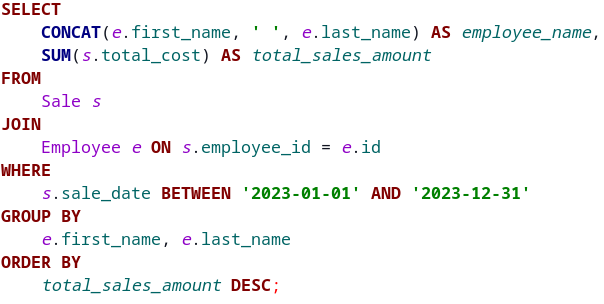
Для получения информации о продажах, включая детали сотрудника, предоставленной услуги и проданного комплектующего, используем запрос с JOIN по таблицам Sale, Employee, Service и Component:

Рисунок 3.15 – Запрос выборки

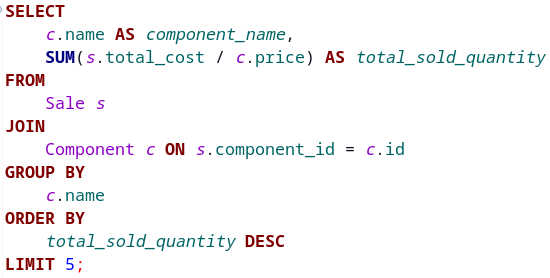
Этот запрос объединяет таблицы Sale, Employee, Service и Component по их внешним ключам, чтобы получить детали о каждой продаже, включая информацию о сотруднике, предоставленной услуге и проданном комплектующем.

## Создание отчетов

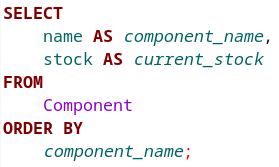
Отчеты в базе данных магазина компьютерных комплектующих играют важную роль в анализе продаж, остатков на складе и других ключевых показателей. Ниже представлены три примера отчетов на SQL, которые могут быть полезны для управления данными и анализа бизнес-процессов.

Рисунок 3.16 – Отчет по продажам

Продажи по сотрудникам за определенный период. Этот отчет позволяет узнать общую сумму продаж, сделанных каждым сотрудником за определенный период времени.

Рисунок 3.17 – Отчет по комлектующим

Этот отчет показывает пять наиболее популярных комплектующих по количеству проданных единиц.

Рисунок 3.18 – Отчет по складу

Общий запас комплектующих на складе. Этот отчет отображает общее количество каждого комплектующего на складе на текущий момент.

Каждый из этих отчетов предоставляет ценную информацию для принятия управленческих решений и анализа текущего состояния бизнеса в магазине компьютерных комплектующих.

# ЗАКЛЮЧЕНИЕ

В ходе выполнения курсовой работы была разработана и реализована база данных для автоматизированной информационной системы "Магазин компьютерных комплектующих". Основной целью работы было проектирование и разработка структуры базы данных, способной эффективно управлять информацией о сотрудниках, комплектующих, услугах, продажах, поставщиках и поставках.

Итоговая база данных предоставляет пользователю возможность эффективного управления информацией о продуктах, услугах, сотрудниках и взаимодействии с поставщиками. Реализованная система позволяет отслеживать состояние складских запасов, анализировать продажи и управлять оперативными данными о комплектующих.

В заключение можно отметить, что выполнение данной работы позволило глубже понять процессы проектирования баз данных, их роль в организации данных и поддержке бизнес-процессов. Разработанная база данных является основой для дальнейшего развития системы учета и управления в магазине компьютерных комплектующих.

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Бекаревич, Ю. Самоучитель PostgreSQL 2010 [Текст] / Ю. Бекаревич, Н. Пушкина. – М.: БХВ-Петербург, 2014. – 432 c.

2. Голицына, О. Л. Базы данных [Текст] / О.Л. Голицына, Н.В. Максимов, И.И. Попов. – М.: Форум, Инфра-М, 2017. – 400 c.

3. Голышева, А. В. Access 2007. Все, что нужно для уверенной работы [Текст] / А. В. Голышева, И. А. Клеандрова, Р. Г. Прокди. – М.: Наука и техника, 2017. – 192 c.

4. Гринченко, Н. Н. Проектирование баз данных. СУБД PostgreSQL [Текст] / Н. Н. Гринченко. – М.: Горячая Линия Телеком, 2012. – 240 c.

5. Гурвиц, Г. PostgreSQL 2010. Разработка приложений на реальном примере [Текст] / Г. Гурвиц. – М.: БХВ-Петербург, 2017. – 496 c.

6. Днепров, А. PostgeSQL [Текст] / А. Днепров. – М.: СИНТЕГ, 2012. – 481 c.

7. Епанешников, А. М. Практика создания приложений в PostgreSQL [Текст] / А. М. Епанешников, В. А. Епанешников. – М.: Гостехиздат, 2016. – 440 c.

8. Золотова, С. И. Практикум по PostgreSQL [Текст] / С. И. Золотова. – М.: Финансы и статистика, 2013. – 354 с.

9. Илюшечкин, В. М. Основы использования и проектирования баз данных. Учебник [Текст] / В.М. Илюшечкин. – М.: Юрайт, 2018. – 214 c.

10. Корнеев, В. В. Базы данных. Интеллектуальная обработка информации [Текст] / В. В. Корнеев, А. Ф. Гареев. – М.: Нолидж, 2011. – 496 с.

11. Кошелев, В. Е. PostgreSQL. Эффективное использование [Текст] / В. Е. Кошелев. – М.: Бином-Пресс, 2015. – 590 c.

12. Кудрявцев, В. Б. Интеллектуальные системы. Учебник и практикум для бакалавриата и магистратуры [Текст] / В.Б. Кудрявцев. – М.: Юрайт, 2016. – 551 c.