ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ ГОРОДА МОСКВЫ

**Государственное бюджетное профессиональное образовательное учреждение города Москвы**

**«Московский государственный колледж электромеханики**

**и информационных технологий»**

КУРСОВАЯ РАБОТА

Тема: **СОЗДАНИЕ БАЗЫ ДАННЫХ** ≪**САЛОН КРАСОТЫ**≫

Специальность: 09.02.07 Информационные системы и программирование

(код, название)

МДК 11.01 «Технология разработки и защиты баз данных»

|  |  |  |
| --- | --- | --- |
| Выполнил | Студент группы 3ИП-1-11-22 | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) | Хлыстов Р.А. |
| Руководитель  курсовой работы | Преподаватель ГБПОУ МГКЭИТ  (должность, место работы) | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) | Пучкова Д.А |
|  | Оценка «\_\_\_\_\_\_\_\_\_\_\_\_\_\_»  «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_2025 г. | |

Москва, 2025

**Содержание**

[Введение 3](#_Toc192787099)

[1. Теоретическая часть 4](#_Toc192787100)

[1.1 Объект и предмет исследования 4](#_Toc192787101)

[1.2 Описание бизнес-процесса 5](#_Toc192787102)

[1.3 Основные сущности и атрибуты базы данных 7](#_Toc192787103)

[2. Практическая часть 9](#_Toc192787104)

[2.1 Разработка ER-модели и диаграммы базы данных 9](#_Toc192787105)

[2.2 Целостность базы данных 10](#_Toc192787106)

[2.3 Реализация базы данных в Pgadmin4 12](#_Toc192787107)

[2.4 Запросы разных типов 13](#_Toc192787108)

[2.5 Создание представлений 14](#_Toc192787109)

[2.6 Создание хранимых процедур 16](#_Toc192787110)

[Заключение 20](#_Toc192787111)

[Список использованной литературы 21](#_Toc192787112)

# **Введение**

Создание базы данных для салона красоты является важным шагом в автоматизации бизнес-процессов, что, в свою очередь, значительно повышает качество обслуживания клиентов и эффективность управления данными. В условиях современного рынка, где конкуренция становится все более жесткой, использование информационных технологий для оптимизации работы становится необходимостью. База данных позволяет систематизировать информацию о клиентах, услугах, сотрудниках и финансовых операциях, что способствует более быстрому и качественному обслуживанию. Автоматизация процессов записи клиентов, управления запасами и обработки платежей позволяет снизить вероятность ошибок и повысить уровень удовлетворенности клиентов.

Целью данной курсовой работы является проектирование и реализация базы данных для салона красоты, которая будет способствовать автоматизации его бизнес-процессов и улучшению качества обслуживания клиентов. В условиях современного рынка, где конкуренция становится все более жесткой, эффективное управление данными и оптимизация процессов являются ключевыми факторами успеха. Разработка базы данных позволит не только упростить внутренние операции, но и создать более удобные условия для клиентов, что в конечном итоге приведет к повышению их удовлетворенности и лояльности.

Для достижения этой цели необходимо решить следующие задачи.Провести анализ существующих бизнес-процессов в салоне красотыю.Определить основные сущности и их атрибуты, которые будут включены в базу данных.Определить атрибуты для каждой сущности, которые будут необходимы для эффективного управления данными и выполнения операций.Разработать ER-модель базы данных, отражающую связи между сущностями.Реализовать базу данных в выбранной системе управления базами данных (СУБД).Создать и протестировать SQL-запросы для работы с данными.Оценить целостность и безопасность базы данных.

# **1. Теоретическая часть**

## **1.1 Объект и предмет исследования**

Объектом исследования данной курсовой работы является бизнес-процесс салона красоты, который охватывает широкий спектр взаимодействий между клиентами, сотрудниками и администраторами. В рамках этого бизнес-процесса происходит множество операций, включая запись клиентов на услуги, управление расписанием сотрудников, обработку платежей, а также сбор и анализ отзывов клиентов. Важным аспектом является создание комфортной и эффективной среды для клиентов, что требует от сотрудников высокого уровня профессионализма и организации работы.

Салон красоты представляет собой сложную систему, в которой взаимодействуют различные участники. Клиенты ожидают качественного обслуживания и удобства при записи на процедуры, сотрудники стремятся оптимизировать свое время и повысить производительность, а администраторы должны обеспечивать бесперебойную работу всех процессов. В связи с этим, эффективное управление данными становится ключевым фактором для успешного функционирования салона.

Предметом исследования является разработка базы данных, которая будет поддерживать все вышеупомянутые процессы, обеспечивая эффективное управление данными и автоматизацию операций. База данных должна включать в себя различные сущности, такие как клиенты, услуги, сотрудники, записи на процедуры и платежи.

Она должна быть спроектирована таким образом, чтобы обеспечить легкий доступ к информации, возможность быстрого поиска и фильтрации данных, а также поддержку аналитических функций для оценки качества обслуживания и выявления тенденций.

Разработка базы данных для салона красоты позволит не только оптимизировать внутренние процессы, но и улучшить взаимодействие с клиентами, что в конечном итоге приведет к повышению уровня удовлетворенности клиентов и увеличению прибыли. В ходе работы будет проведен анализ существующих бизнес-процессов, определены ключевые требования к базе данных и разработаны соответствующие модели данных, что позволит создать эффективный инструмент для управления салоном красоты.

## **1.2 Описание бизнес-процесса**

Анализ текущих процессов в салоне красоты.Салон красоты представляет собой комплексное предприятие, предоставляющее разнообразные услуги в области ухода за внешностью. В современных условиях конкуренции и высоких ожиданий клиентов, эффективное управление бизнес-процессами становится критически важным. Основные процессы, протекающие в салоне красоты, включают:

Запись клиентов: Процесс записи клиентов на услуги может осуществляться как в режиме оффлайн (по телефону или при личном визите), так и в режиме онлайн через веб-сайт или мобильное приложение. Важно, чтобы система записи была удобной и доступной, что позволяет избежать очередей и повысить уровень удовлетворенности клиентов.

Предоставление услуг: Сотрудники салона (парикмахеры, косметологи, маникюристы и т.д.) выполняют услуги, основываясь на предварительной записи. Качество предоставляемых услуг напрямую влияет на репутацию салона и уровень лояльности клиентов.

Управление запасами: Салон красоты использует различные материалы и продукты (косметику, инструменты и т.д.), которые необходимо контролировать. Эффективное управление запасами позволяет избежать дефицита необходимых материалов и оптимизировать затраты.

Обработка платежей: После предоставления услуги необходимо произвести расчет с клиентом. Это может включать различные способы оплаты (наличные, банковские карты, электронные платежи). Важно обеспечить безопасность и удобство процесса оплаты.

Обработка отзывов: Сбор и анализ отзывов клиентов о качестве услуг и уровне обслуживания позволяет выявить слабые места в работе салона и улучшить клиентский опыт.В бизнес-процессе салона красоты участвуют несколько ключевых участников, каждый из которых выполняет определенные функции.

Клиенты: Основные пользователи услуг салона. Их потребности и ожидания определяют качество предоставляемых услуг. Клиенты могут записываться на услуги, оставлять отзывы и участвовать в акциях.

Сотрудники: Это специалисты, предоставляющие услуги (парикмахеры, косметологи, маникюристы и т.д.). Они отвечают за качество выполнения услуг и взаимодействие с клиентами. Сотрудники также могут участвовать в процессе управления запасами.

Администраторы: Лица, ответственные за организацию работы салона. Они управляют расписанием сотрудников, контролируют процесс записи клиентов, обрабатывают платежи и собирают отзывы. Администраторы играют ключевую роль в обеспечении эффективного функционирования салона.

Менеджеры: В некоторых салонах могут быть менеджеры, которые занимаются стратегическим управлением, маркетингом и развитием бизнеса. Они анализируют финансовые показатели и разрабатывают новые услуги и акции.

Описание основных операций

Запись клиентов:

Клиенты могут записываться на услуги через телефон, сайт или мобильное приложение.

Администраторы фиксируют записи в системе, указывая дату, время, услугу и специалиста.

Предоставление услуг:

Сотрудники выполняют услуги в соответствии с записями.

Важно поддерживать высокий уровень качества и профессионализма.

Управление запасами:Сотрудники и администраторы контролируют наличие необходимых материалов и продуктов.При необходимости осуществляется заказ новых запасов.Обработка платежей:После завершения услуги клиент производит оплату.Администраторы обрабатывают платежи и выдают чеки.Обработка отзывов:Клиенты могут оставлять отзывы о качестве услуг.Администраторы анализируют отзывы и принимают меры для улучшения сервиса.

## **1.3 Основные сущности и атрибуты базы данных**

Сущность: Payments (Платежи).   
Атрибуты:amount numeric(10,2) (Сумма платежа), paymentmethod character varying(30) (Метод оплаты), paymentdate date (Дата платежа).

Сущность: Inventory (Запасы).   
Атрибуты:productid integer (Идентификатор продукта),arrivaldate date (Дата прихода),

quantity integer (Количество), responsibleemployeeid integer (Идентификатор ответственного сотрудника).

Сущность: Products (Продукты).   
Атрибуты:productname character varying(100) (Название продукта),description text (Описание), stockquantity integer (Количество на складе).

Сущность: Promotions (Акции).   
Атрибуты:promotionname character varying(100) (Название акции), description text (Описание),discountpercentage numeric(5,2) (Процент скидки), startdate date (Дата начала), enddate date (Дата окончания).

Сущность: Employees (Сотрудники).   
Атрибуты: firstname character varying(50) (Имя), lastname character varying(50) (Фамилия), position character varying(50) (Должность), phone character varying(10) (Телефон), email character varying(100) (Электронная почта),hiredate date (Дата найма).

Сущность: Clients (Клиенты).   
Атрибуты:firstname character varying(50) (Имя),lastname character varying(50) (Фамилия).

Сущность: Services (Услуги).   
Атрибуты:servicename character varying(100) (Название услуги),description text (Описание),price numeric(10,2) (Цена).

Сущность: Reviews (Отзывы) .  
Атрибуты: clientid integer (Идентификатор клиента), rating integer (Оценка), comment text (Комментарий), reviewdate date (Дата отзыва).

Сущность: Schedule (Расписание).   
Атрибуты:employeeid integer (Идентификатор сотрудника),date date (Дата расписания),starttime time without time zone (Время начала),endtime time without time zone (Время окончания).

Взаимоотношения между сущностями:Сущности имеют связи между собой через идентификаторы, такие как productid, employeeid, и clientid, что позволяет создавать взаимосвязи и обеспечивать целостность данных.

# **2. Практическая часть**

## **2.1 Разработка ER-модели и диаграммы базы данных**

ER-диаграмма представляет собой структуру базы данных для управления различными аспектами бизнеса, включающую данные о клиентах, продуктах, услугах, платежах и т. д. Её можно увидеть на рисунке 1.

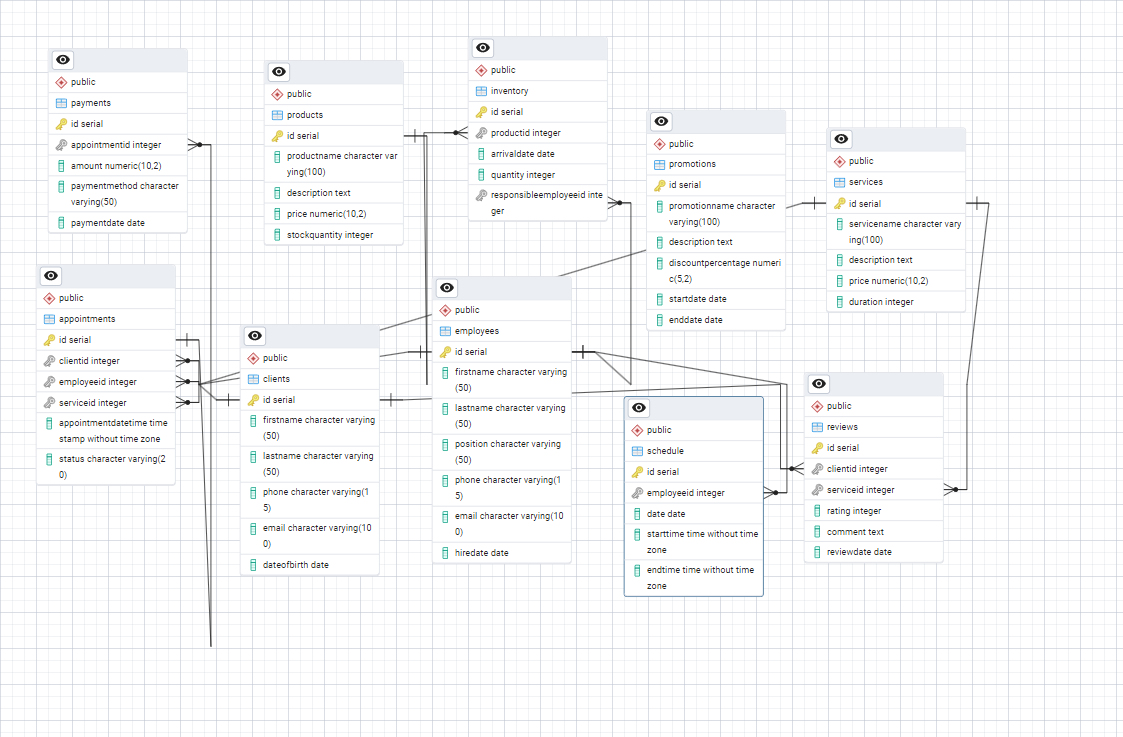


Рисунок 1. ER-модель базы данных.

Диаграмма представляет собой схему взаимодействия различных участников бизнеса, таких как сотрудники и клиенты, в рамках определенных процессов. Центральные элементы базы наглядно демонстрируют связи между различными функциями и действиями. Её можно увидеть на рисунке 2.

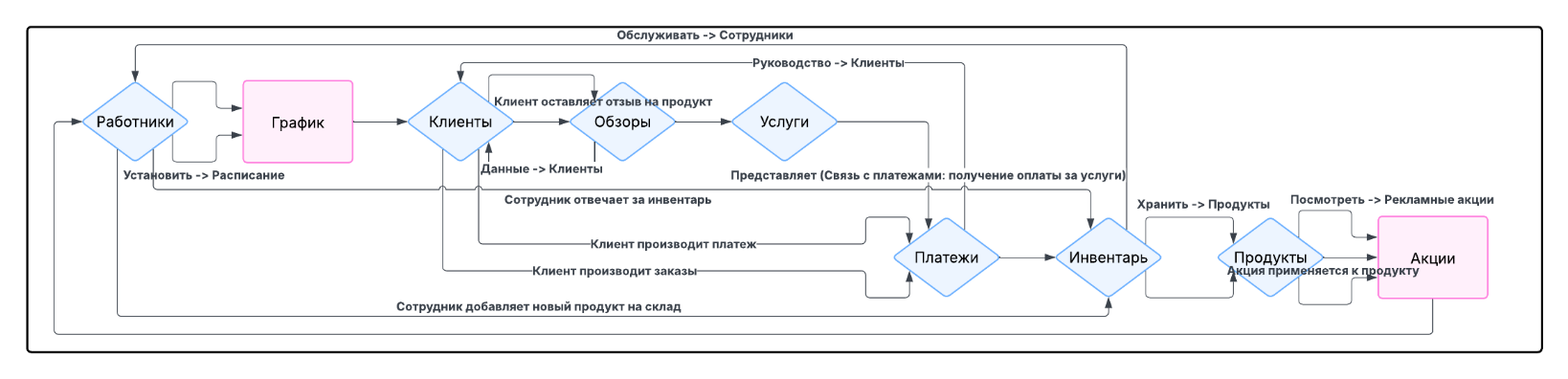
****

Рисунок 2. Диаграмма базы данных.

## **2.2 Целостность базы данных**

Целостность базы данных — это свойство, которое гарантирует, что данные в базе данных остаются точными и последовательными. Одним из ключевых аспектов обеспечения целостности данных является правильное использование первичных и внешних ключей, а также установление связей между таблицами.

1. Определение первичных и внешних ключей

Первичный ключ (Primary Key): Это уникальный идентификатор для каждой записи в таблице. Он не может содержать NULL-значения и должен быть уникальным для каждой строки. Первичный ключ обеспечивает уникальность данных и позволяет быстро находить записи.

Примеры первичных ключей:

В таблице Clients первичный ключ — ID.

В таблице Employees первичный ключ — ID.

В таблице Services первичный ключ — ID.

В таблице Appointments первичный ключ — ID.

В таблице Payments первичный ключ — ID.

В таблице Reviews первичный ключ — ID.

В таблице Schedule первичный ключ — ID.

В таблице Promotions первичный ключ — ID.

В таблице Products первичный ключ — ID.

В таблице Inventory первичный ключ — ID.

Внешний ключ (Foreign Key): Это поле (или набор полей) в одной таблице, которое ссылается на первичный ключ другой таблицы. Внешние ключи обеспечивают связь между таблицами и помогают поддерживать целостность данных, предотвращая создание "сиротских" записей.

Примеры внешних ключей:

В таблице Appointments:

ClientID ссылается на ID в таблице Clients.

EmployeeID ссылается на ID в таблице Employees.

ServiceID ссылается на ID в таблице Services.

В таблице Payments:

AppointmentID ссылается на ID в таблице Appointments.

В таблице Reviews:

ClientID ссылается на ID в таблице Clients.

ServiceID ссылается на ID в таблице Services.

В таблице Schedule:

EmployeeID ссылается на ID в таблице Employees.

В таблице Inventory:

ProductID ссылается на ID в таблице Products.

ResponsibleEmployeeID ссылается на ID в таблице Employees.

2. Описание связей между таблицами.Связи между таблицами могут быть следующих типов:Один к одному (1:1): Каждая запись в одной таблице соответствует одной записи в другой таблице. Например, если у вас есть таблица Clients и таблица ClientDetails, где каждая запись в Clients имеет только одну запись в ClientDetails.

Один ко многим (1:N): Одна запись в первой таблице может соответствовать нескольким записям во второй таблице. Например, один клиент может иметь несколько записей в таблице Appointments, но каждая запись в Appointments относится только к одному клиенту.Многие ко многим (M:N): Записи в одной таблице могут соответствовать нескольким записям в другой таблице и наоборот. Для реализации такой связи обычно создается промежуточная таблица. Например, если у вас есть таблицы Clients и Services, и один клиент может использовать несколько услуг, а одна услуга может быть предоставлена нескольким клиентам, то можно создать таблицу ClientServices, которая будет содержать ClientID и ServiceID.

Поддержка целостности данных: Использование первичных и внешних ключей помогает поддерживать целостность данных, предотвращая создание записей, которые ссылаются на несуществующие записи в других таблицах.

Например, если вы попытаетесь добавить запись в таблицу Appointments с ClientID, который не существует в таблице Clients, СУБД выдаст ошибку.

Обеспечение согласованности: Связи между таблицами помогают обеспечить согласованность данных. Например, если клиент удаляется из таблицы Clients, все связанные записи в таблице Appointments могут быть автоматически удалены (если настроены каскадные удаления).

**2.3 Реализация базы данных в Pgadmin4**

В рамках проекта приступаем к реализации базы данных, используя систему управления базами данных PostgreSQL через интерфейс PgAdmin4. Эта база данных будет поддерживать следующие ключевые сущности и связи для обеспечения надежного и эффективного управления данными. На рисунке 3 можно увидеть запрос для создание бызы данных. Эффективная организация данных позволит повысить качество обслуживания клиентов и оптимизировать внутренние процессы компании.

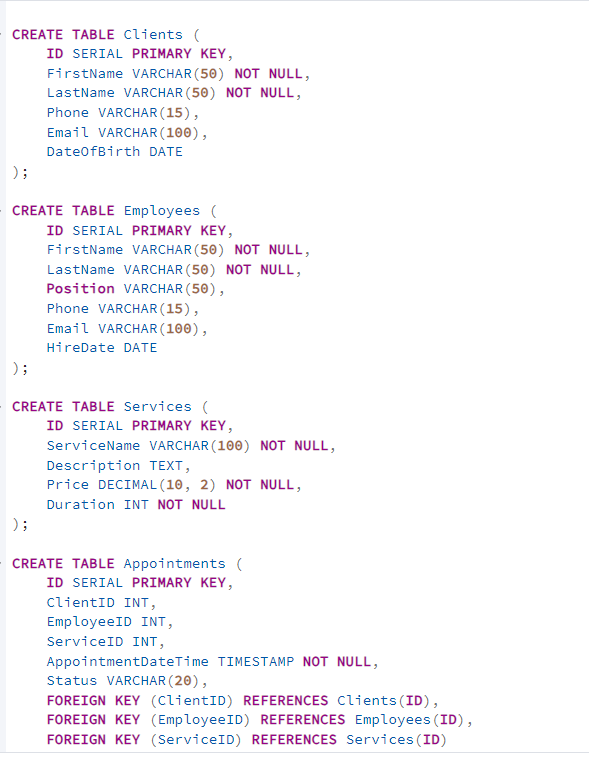


Рисунок 3. Запрос для создания таблиц.

## **2.4 Запросы разных типов**

Добавление данных в таблицу "Клиенты" (Clients). На рисунке 4 представлен SQL-запрос INSERT INTO, который добавляет данные в таблицу "Clients". На рисунке 4 можно увидеть запрос для добовления данных в таблицу.

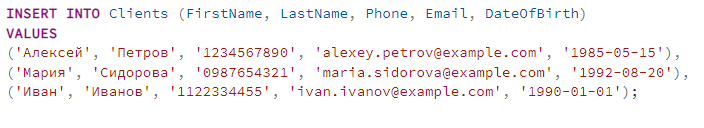


Рисунок 4. Запрос для добовления данных в таблицу.

На Рисунке 5 отображается результат выполнения SQL-запроса SELECT \* FROM Clients, который выбирает все записи из таблицы "Clients".

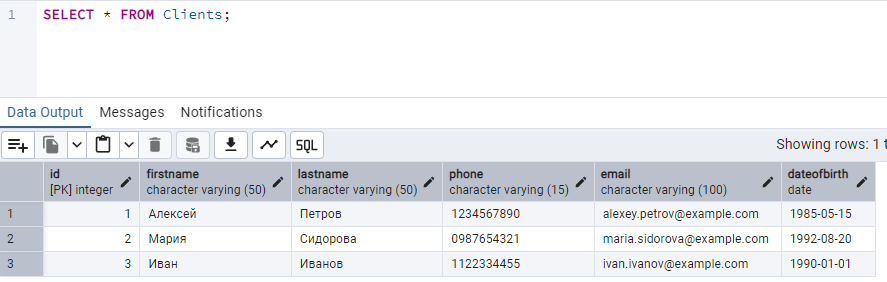


Рисунок 5. Запрос на выборку всех клиентов.

Запрос на обновление информации оклиенте. На рисунке 6 представлен SQL-запрос UPDATE Clients, который обновляет данные уже существующего клиента.

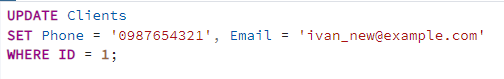


Рисунок 6. Запрос на обновление информации о клиенте.

На рисунке 7 представлено использование SQL-запроса DELETE FROM Clients, предназначенного для удаления клиента из таблицы.

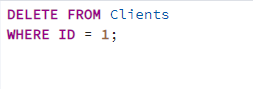


Рисунок **7**. Запрос на обновление информации о клиенте**.**

## **2.5 Создание представлений**

Представление для активных клиентов.Это представление будет содержать информацию о клиентах, которые записаны на услуги в текущем месяце. Оно включает имя клиента, телефон и дату последней записи. На рисунке 8 показано представление для активных клиентов.



Рисунок 8. Представление для активных клиентов.

Представление для услуг с высокой оценкой.

Это представление будет содержать информацию о услугах, которые получили оценку 4 и выше от клиентов. Оно включает название услуги, среднюю оценку и количество отзывов. На рисунке 9 показано представление для услуг с высокой оценкой.

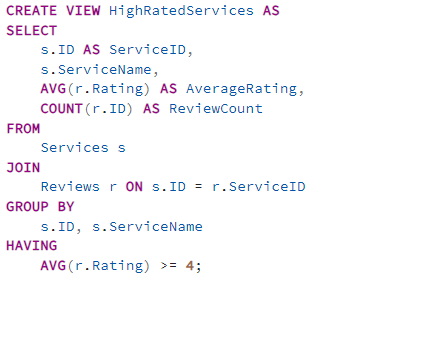


Рисунок 9. Представление для услуг с высокой оценкой.

## **2.6 Создание хранимых процедур**

Хранимые процедуры — это наборы SQL-запросов, которые хранятся в базе данных и могут быть выполнены по запросу. Хранимые процедуры позволяют обеспечивать безопасность данных, так как доступ к ним можно ограничить, а также упрощают управление изменениями в бизнес-логике. Они позволяют инкапсулировать бизнес-логику, улучшать производительность и обеспечивать.

Хранимая процедура для добавления нового клиента

Эта процедура принимает параметры для добавления нового клиента в таблицу Clients. Ее можно увидеть на рисунке 10.

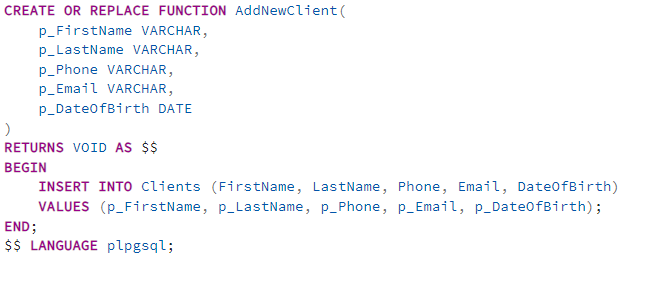


Рисунок 10.Процедура для добавления нового клиента в таблицу Clients.

На рисунке 11 можно увидеть как вызывается функция AddNewClient.

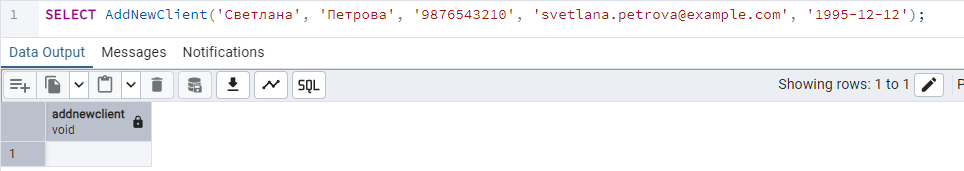


Рисунок 11. Вызов функции.

Хранимая функция для обновления информации о клиенте.Эта функция обновляет информацию о клиенте по его ID. Ее можно увидеть на рисунке 12.  Поскольку процедуры компилируются и хранятся в базе данных, они могут выполняться быстрее, чем обычные SQL-запросы, которые компилируются каждый раз при выполнении.

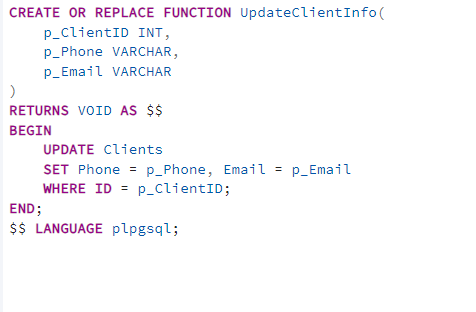


Рисунок 12. Процедура для обновления информации о клиенте..

На рисунке 13 можно увидеть как вызывается функция UpdateClientInfo.

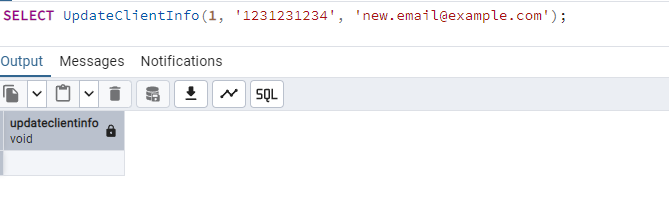


Рисунок 13. Вызов функции.

**2.7 Реализация работы триггеров**

Триггер AFTER INSERT  это специальный объект базы данных, который автоматически выполняет определенные действия после успешного добавления новой записи в таблицу..Этот триггер срабатывает после вставки новой записи в таблицу. Например, он может использоваться для автоматического создания записи в журнале изменений, когда новый клиент добавляется в таблицу Clients. Срабатывает после того, как новая запись была успешно добавлена в таблицу Clients.

Записывает информацию о новом клиенте в таблицу ClientLog, включая ID клиента, действие и дату. На рисунке 14 можно увидеть запрос для создание тригера AFTER INSERT. Таким образом, триггер позволяет автоматически вести учет изменений в базе данных, что может быть полезно для аудита и анализа данных.

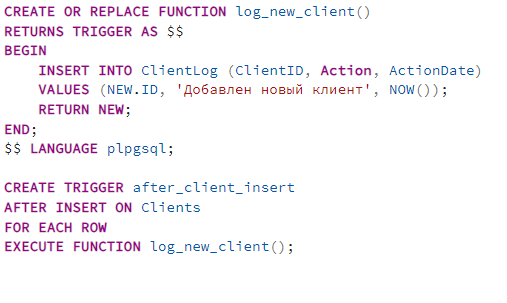


Рисунок 14. Тригер AFTER INSERT.

Триггер AFTER UPDATE. Этот триггер срабатывает после обновления существующей записи в таблице. Например, он может использоваться для записи изменений в информацию о клиенте в журнале. Срабатывает после того, как новая запись была успешно добавлена в таблицу Clients.

Записывает информацию о новом клиенте в таблицу ClientLog, включая ID клиента, действие и дату. На рисунке 15 можно увидеть запрос для создание тригера AFTER UPDATE.

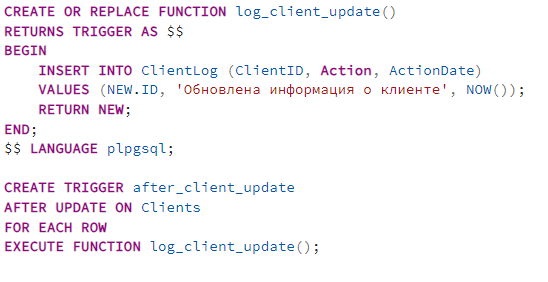


Рисунок 15. Тригер AFTER UPDATE.

Триггер AFTER DELETE. Этот триггер срабатывает после удаления записи из таблицы. Например, он может использоваться для записи информации о том, что клиент был удален. Срабатывает после того, как запись была удалена из таблицы Clients.Записывает информацию о том, что клиент был удален, в таблицу ClientLog. На рисунке 16 можно увидеть запрос для создание тригера AFTER DELETE .

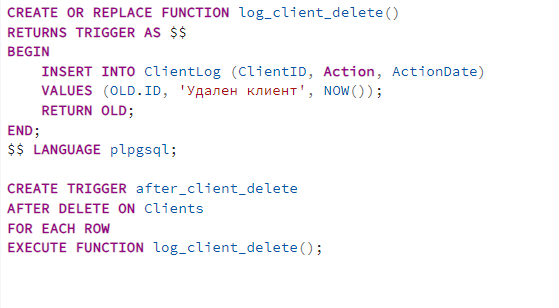


Рисунок 16. Тригер AFTER DELETE.

# **Заключение**

В ходе выполнения курсовой работы была разработана база данных для салона красоты, которая позволяет автоматизировать и оптимизировать ключевые бизнес-процессы. Основные этапы работы включали анализ текущих процессов, проектирование структуры базы данных, создание ER-модели, реализацию базы данных в выбранной системе управления базами данных (СУБД), а также разработку различных запросов, представлений и хранимых процедур.

Основные выводы по курсовой работе

Автоматизация бизнес-процессов: Создание базы данных позволяет значительно упростить процессы записи клиентов, управления услугами, обработки платежей и сбора отзывов. Это, в свою очередь, повышает уровень обслуживания клиентов и снижает вероятность ошибок.

Целостность данных: Использование первичных и внешних ключей, а также триггеров, обеспечивает целостность и согласованность данных в базе. Это позволяет избежать создания "сиротских" записей и гарантирует, что все данные остаются актуальными и связанными.

Гибкость и масштабируемость: Разработанная база данных обладает гибкостью, что позволяет легко добавлять новые услуги, клиентов и сотрудников. Это также открывает возможности для дальнейшего расширения функциональности, например, добавления новых отчетов или интеграции с другими системами.

Создание представлений и хранимых процедур позволяет эффективно анализировать данные, что может помочь в принятии управленческих решений и улучшении бизнес-процессов.

# **Список использованной литературы**

1.**Дейт, С. Д**. Введение в системы баз данных. — Москва: Вильяме, 2005. — 1316с.

2.**Кодд, Э. Ф.** Реляционная база данных: Практическая основа для вычислений. — Москва: Мир, 1982. — 300с.

3.**Корф, Х. Ф.,** Силбершатц, А. Концепции систем баз данных. — Москва: Вильямс, 1999. — 500с.

4.**Маннино, М. В.** Проектирование баз данных, разработка приложений и администрирование. — Киев: ДиаСофт, 2011. — 650с.

5.**Роб, П., Коронель, С.** Системы баз данных: Проектирование, реализация и управление. — Москва: Диалектика, 2016. — 800с.

6.**Силбершатц, А., Корф, Х. Ф., Сударшан, С.** Концепции систем баз данных. — Москва: Вильямс, 2011. — 600с.

**7.Эльмасри, Р., Навас, С. Б**. Основы систем баз данных. — Москва: Вильямс, 2015. — 750с.

8.**Эрнандес, М. Дж**. Проектирование баз данных для простых смертных: Практическое руководство по проектированию реляционных баз данных. — Санкт-Петербург: Питер, 2013. — 460с.