**РОССИЙСКАЯ ФЕДЕРАЦИЯ**

**ХАНТЫ-МАНСИЙСКИЙ АВТОНОМНЫЙ ОКРУГ-ЮГРА**



**АВТОНОМНАЯ НЕКОММЕРЧЕСКАЯ ПРОФЕССИОНАЛЬНАЯ ОБРАЗОВАТЕЛЬНАЯ ОРГАНИЗАЦИЯ**

**«СУРГУТСКИЙ ИНСТИТУТ ЭКОНОМИКИ, УПРАВЛЕНИЯ И ПРАВА»**

КУРСОВАЯ РАБОТА

по дисциплине «Технология разработки и защиты баз данных»

на тему «Разработка модели реляционной базы данных, для фитнес зала»

Выполнил: Таращук Максим Павлович

студент(ка) группы ИС 23/11

Проверил: Преподаватель профессиональных дисциплин

Колмыков М.В.\_\_\_\_\_\_\_\_\_\_\_

Сургут, 2024

СОДЕРЖАНИЕ

|  |  |
| --- | --- |
| ВВЕДЕНИЕ | 2 |
| ГЛАВА I ТЕОРЕТИЧЕСКАЯ ЧАСТЬ: | 3 |
| 1.1 Основные положения теории баз данных, хранилищ данных, баз знаний | 3 |
| 1.1.1 Типы моделей данных: иерархическая, сетевая, реляционная и объектно ‒ ориентированная модели данных | 3 |
| 1.1.2 Свойства базы данных | 4 |
| 1.1.3 Основные принципы построения хранилищ данных и баз знаний | 6 |
| 1.2 Анализ предметной области проектирования | 7 |
| 1.2.1 Выбор приложения на чем будет реализована база дынных | 8 |
| 1.2.2 Описание основных сущностей и атрибутов, связей между ними | 10 |
| 1.2.3 Выделение ключевых сущностей, их атрибутов и связей | 11 |
| ГЛАВА II. ПРАКТИЧЕСКАЯ ЧАСТЬ: | 13 |
| 2.1 Определение сущностей логической модели БД | 13 |
| 2.1.1 Построение логической БД | 18 |
|  | 18 |
| 2.2.1 Запросы для БД | 19 |
| 3 ЗАКЛЮЧЕНИЕ | 24 |
| 4 СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ | 25 |

ВВЕДЕНИЕ

В современном мире, где здоровый образ жизни становится все более популярным, фитнес-залы играют ключевую роль в обеспечении населения возможностью вести активный и здоровый образ жизни. Однако, чтобы эффективно управлять такими заведениями, необходимо надежное и гибкое информационное обеспечение, которое позволит хранить, обрабатывать и анализировать данные о клиентах, тренировках, абонементах и многих других аспектах деятельности фитнес-центра.

**Разработка модели реляционной базы данных для фитнес-зала** - это актуальная и многогранная задача, которая требует глубокого понимания принципов реляционных баз данных, а также специфики работы фитнес-центров.

**Целью** данной курсовой работы является разработка модели реляционной базы данных, которая позволит эффективно управлять информацией в фитнес-зале, обеспечивая:

**Точность и актуальность данных:** База данных должна хранить информацию о клиентах, тренерах, абонементах, занятиях и других сущностях в фитнес-зале.

**Гибкость и масштабируемость:** Модель должна позволять легко добавлять новые данные и изменять существующие, а также адаптироваться к растущим потребностям фитнес-центра.

**Безопасность и конфиденциальность:** Важно обеспечить защиту персональных данных клиентов и предотвратить несанкционированный доступ к информации.

ГЛАВА I. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

1.1 Основные положения теории баз данных, хранилищ данных, баз знаний

База данных — это структурированная коллекция данных, хранящаяся в электронном виде, которая организована таким образом, чтобы минимизировать избыточность и обеспечить эффективное использование для одного или нескольких приложений.

Цель создания баз данных — разработка системы, которая не зависит от программного обеспечения, используемых технических средств и физического расположения данных. Проектирование баз данных предполагает их многоцелевое использование.

База данных, в самом простом виде, представляет собой набор двумерных таблиц.

Схема данных — это описание логической структуры данных, выполненное на языке описания данных и обрабатываемое системой управления базами данных (СУБД).

Схема пользователя — это конкретный вариант порядка полей таблицы, зафиксированный для определенного пользователя.

Система управления базами данных (СУБД) — это программное обеспечение, отвечающее за организацию, хранение, целостность, изменение, чтение и безопасность информации в базе данных.

Информационная система — это система, автоматизирующая сбор, обработку и управление данными, включающая технические средства обработки данных, программное обеспечение и обслуживающий персонал.

1.1.1 Типы моделей данных: иерархическая, сетевая, реляционная и объектно-ориентированная модели данных

В зависимости от способа хранения данных и организации связей между ними, базы данных делятся на несколько типов:

Иерархическая: Данные организованы в древовидную структуру, где корневой элемент связан с элементами второго уровня иерархии.

Сетевая: Похожа на иерархическую, но связи могут устанавливаться между любыми элементами графа.

Реляционная: Данные организованы в виде таблиц, где каждая запись — это строка с уникальным ключом.

Объектно-ориентированная: Данные хранятся в виде объектов, обладающих свойствами и методами.

Документо-ориентированная: Данные организованы в виде документов, которые могут быть структурированными или неструктурированными.

Графовая: Данные представлены в виде графа, где узлы — это объекты, а ребра — связи между ними.

Каждая модель имеет свои преимущества и недостатки, и выбор модели зависит от конкретных требований приложения и характера данных.

1.1.2 Свойства базы данных

Основные свойства баз данных:

Структурированность: Данные организованы в определенной структуре, что позволяет эффективно хранить и управлять ими.

Постоянство: Данные сохраняются долгосрочно и остаются доступными после завершения сессии работы с приложением.

Доступность: База данных обеспечивает доступ к данным для пользователей и приложений в соответствии с установленными правами доступа.

Целостность: База данных гарантирует точность и согласованность данных во всей системе, даже при одновременном доступе нескольких пользователей.

Безопасность: База данных обеспечивает защиту данных от несанкционированного доступа, изменений или утраты.

Масштабируемость: База данных должна быть способна обрабатывать как малые, так и большие объемы данных и увеличивать производительность при необходимости.

Производительность: База данных должна обеспечивать быстрый доступ к данным и эффективную обработку запросов.

Распределенность: Современные базы данных могут быть распределенными, что обеспечивает отказоустойчивость и масштабируемость.

Резервирование: База данных должна иметь механизмы резервного копирования и восстановления данных.

1.1.3 Основные принципы построения хранилищ данных и баз знаний

Основные принципы построения баз данных основаны на научных принципах, которые позволяют создавать высококачественные системы, соответствующие современным требованиям.

Из множества принципов создания баз данных, наиболее важными являются интеграция данных и централизация управления данными.

Принцип интеграции данных заключается в объединении отдельных, несвязанных данных в единую базу данных, которая представляет собой единый информационный массив для пользователя и его прикладных программ. Это упрощает поиск и обработку взаимосвязанных данных, уменьшает избыточность данных и упрощает процесс ведения баз данных.

Принцип централизации управления данными обеспечивает единый контроль и администрирование всей базы данных, упрощая управление доступом, резервное копирование и восстановление.

Эти принципы обеспечивают основу для создания высококачественных систем, которые соответствуют современным требованиям к надежности, эффективности и масштабируемости.

1.2 Анализ предметной области проектирования

Анализ предметной области проектирования для парикмахерской включает изучение основных аспектов деятельности, потребностей клиентов и требований к функциональности программного обеспечения.

Ключевые моменты, которые следует учитывать:

Услуги парикмахерской: Анализ предоставляемых услуг, их частоты и стоимости.

Расписание и запись: Анализ расписания работы персонала и возможностей онлайн-записи.

Инвентарь и расходные материалы: Анализ инвентаря и его управления.

Учет услуг и финансов: Система учета доходов, расходов, прибыли и других финансовых показателей.

Клиентская база и управление клиентами: Ведение базы данных клиентов для отслеживания предпочтений, истории посещений и контактных данных.

Управление персоналом: Оптимальное управление персоналом, включая назначение рабочих графиков, контроль за производительностью и мотивацию сотрудников.

Безопасность и гигиена: Соблюдение норм безопасности и гигиены для обеспечения безопасности персонала и клиентов.

Анализ этих аспектов поможет определить требования к программному обеспечению для парикмахерской и разработать систему, которая будет эффективно поддерживать операции бизнеса.

1.2.1 Создание базы данных на SQLite

SQLite — это компактная и самодостаточная реляционная СУБД, которая является частью стандартной библиотеки в большинстве языков программирования. Она отличается от других СУБД тем, что не является серверной базой данных, а вместо этого представляет собой библиотеку, которая непосредственно встраивается в приложение.

Основные особенности SQLite:

Легковесность: Очень компактна и не требует установки отдельного сервера баз данных. Вся база данных хранится в одном файле.

Надежность: Обеспечивает надежное хранение данных с поддержкой транзакций и ACID-свойств.

Кросс-платформенность: Поддерживается на различных операционных системах.

Поддержка большого объема данных: Может управлять базами данных размером до нескольких терабайт.

Преимущества SQLite:

Простота интеграции: Легко интегрируется в приложения благодаря своей компактности и отсутствию необходимости установки отдельного сервера баз данных.

Независимость от платформы: Поддерживается на различных операционных системах.

Эффективность: Обеспечивает высокую производительность и эффективное использование ресурсов.

Низкие требования к ресурсам: Не требует больших объемов оперативной памяти или вычислительных мощностей для работы.

Богатый набор функций: Поддерживает множество функций SQL.

Бесплатность и открытость: Является бесплатным и открытым программным обеспечением.

1.2.2 Описание основных сущностей и атрибутов, связей между ними

Основные сущности, которые могут быть присутствовать в базе данных парикмахерской:

Клиенты: Хранение информации о клиентах, такой как имя, контактные данные, предпочтения в отношении услуг, история посещений.

Услуги: Хранение информации о предоставляемых услугах и их стоимости.

Персонал: Хранение информации о персонале, включая имена, специализацию, контактные данные и график работы.

Расписание: Хранение информации о расписании работы сотрудников и записях клиентов на определенные даты и время.

Операции: Отслеживание финансовых операций, таких как оплата услуг, расходы на инвентарь и другие расходы.

Инвентарь: Хранение информации о инвентаре и расходных материалах.

Связи между сущностями:

Каждый клиент может иметь несколько записей на услуги.

Каждый парикмахер может иметь много записей на услуги, в которых он участвует.

Услуги могут быть назначены на несколько записей клиентов.

Такая база данных позволит эффективно управлять информацией о клиентах, услугах и записях, а также облегчит планирование графика работы парикмахеров.

1.2.3 Выделение ключевых сущностей, их атрибутов и связей

Для выделения ключевых сущностей, их атрибутов и связей в базе данных используется метод семантического моделирования.

Реляционные модели данных подходят для моделирования предметных областей. Однако они имеют ограничения:

Не обеспечивают адекватных средств представления смысла данных.

Не предоставляют средств для разделения сущностей и связей.

Эти ограничения привели к появлению семантических (концептуальных, информационных) моделей данных. Они содержат структурную, операционную и целостную части, аналогичные реляционным моделям. Основная цель семантических моделей — обеспечить возможность представления семантики данных.

Наиболее известным представителем класса семантических моделей является модель сущность-связь (ER-модель). ER-модели обычно представляются в графической форме, с использованием ER-диаграмм или других графических нотаций. Основными понятиями ER-модели являются сущности, отношения и атрибуты.

ER-модель позволяет парикмахерской визуально представить свои данные и отношения между ними. Это помогает обеспечить целостность и точность данных, а также облегчает извлечение ценной информации для принятия решений.

ГЛАВА II. ПРАКТИЧЕСКАЯ ЧАСТЬ

2.1 Определение сущностей логической модели БД

В результате проведённого предметного анализа было выявлено что, данная база данных будет содержать следующие сущности:

− клиенты;

− залы;

− абонемент;

− тренажёрный зал;

− тренировка;

− Срок абонемента.

Для создания логической и физической схемы базы данных использовались инструменты Draw.io и DBdesigner. Данные инструменты позволили визуализировать концептуальную модель данных и преобразовать ее в физическую схему, учитывающую конкретную систему управления базами данных.

Таблица 1 ‒ «Клиент»

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название | Тип | Описание |
| 1 | Id | Primary key, autoincrement, integer | Уникальный идентификатор |
| 2 | full\_name | Varchar | ФИО клиента |
| 3 | hall\_id | Varchar | Идентификатор зала, куда куплен абонемент |
| 4 | begin\_ticket | Date | Начало абонемента |
| 5 | end\_ticket | Date | Конец абонемента |

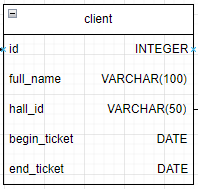


Рисунок 1 ‒ Таблица сущности клиента

Сущность клиент хранит информацию о клиентах фитнес зала, включая их данные, историю посещений, покупку абонемента. Эта информация используется для управления отношениями с клиентами, предоставления персонализированных услуг и повышения общего качества обслуживания.

Таблица 2 ‒ «Зал»

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название | Тип | Описание |
| 1 | id | Primary key, autoincrement, integer | Уникальный идентификатор |
| 2 | gym | Varchar | Зал с тренажёрами |
| 3 | dance | Varchar | Танцевальный зал |
| 4 | box | Varchar | Зал для занятия единоборствами |

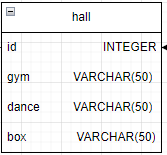


Рисунок 2 ‒ Таблица сущности зала

Сущность зала хранит информацию о наличии залов. Эта информация используется для знания используемых залов.

Таблица 3 ‒ «Фитнес комнат»

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название | Тип | Описание |
| 1 | id | Primary key, autoincrement, integer | Уникальный идентификатор |
| 2 | client\_id | Varchar | Идентификатор клиента, который купил абонемент |
| 3 | hall\_id | Varchar | Идентификатор зала, где было занятия |
| 4 | season\_ticket | Varchar | Абонент |
| 5 | coach | Varchar | Тренер |

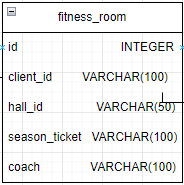


Рисунок 3 ‒ Таблица сущности фитнес комнат

Сущность фитнес комнат хранит информацию о различных услугах, предоставляемых фитнесом, включая их клиентов, нужно ли занятия с тренером. Эта информация отслеживания гостей, планирования встреч и предоставления клиентам информации о доступных вариантах.

Таблица 4 ‒ «Тренировка»

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название | Тип | Описание |
| 1 | id | Primary key, autoincrement, integer | Уникальный идентификатор |
| 4 | client\_id | Varchar | Идентификатор клиента, который занимался |
| 5 | begin\_coaching | Time | Время начала тренировки |
| 6 | end\_coaching | Time | Время конца тренировки |

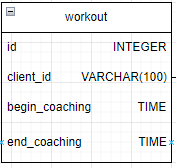


Рисунок 4 ‒ Таблица сущности тренировка

Сущность тренировки хранит информацию о клиенте, который пришёл, во сколько он начал тренировку и во сколько закончил.

Таблица 5 ‒ «Абонемент»

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название | Тип | Описание |
| 1 | id | Primary key, autoincrement, integer | Уникальный идентификатор |
| 2 | time\_ticket\_id | Date | Срок абонемента |
| 3 | hall\_id | Varchar | Уникальный идентификатор зала, на который купили абонемент |

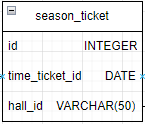
****

Рисунок 5 ‒ Таблица сущности абонемента

Сущность абонемента хранит информацию о том на сколько был куплен абонемент и в какой зал.

Таблица 6 ‒ «Срок абонемента»

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название | Тип | Описание |
| 1 | id | Primary key, autoincrement, integer | Уникальный идентификатор |
| 2 | one\_traning | Time | Одна тренировка |
| 3 | month | Time | Абонемент на месяц |
| 4 | year | Time | Абонемент на год |

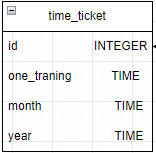


Рисунок 6 ‒ Таблица сущности срока абонемента

Сущность срока абонемента хранит информацию о том какие абонементы присутствуют и что можно приобрести.

2.1.1 Построение логической БД

На основе определения сущностей, можно построить логическую БД, я воспользовался программой Draw.io, и вот что у меня получилось:

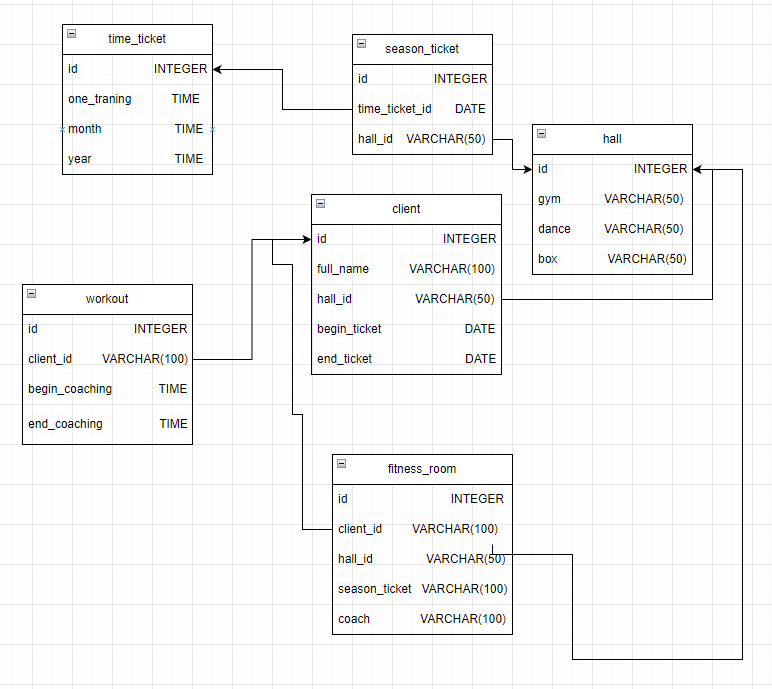


Рисунок 7 ‒ База данных

2.2 Построение физической модели БД

После создания логической БД, можно создать физическую БД, и реализовать связь между таблицами «Рисунок 7». Теперь же я воспользуюсь программой DBdesigner:

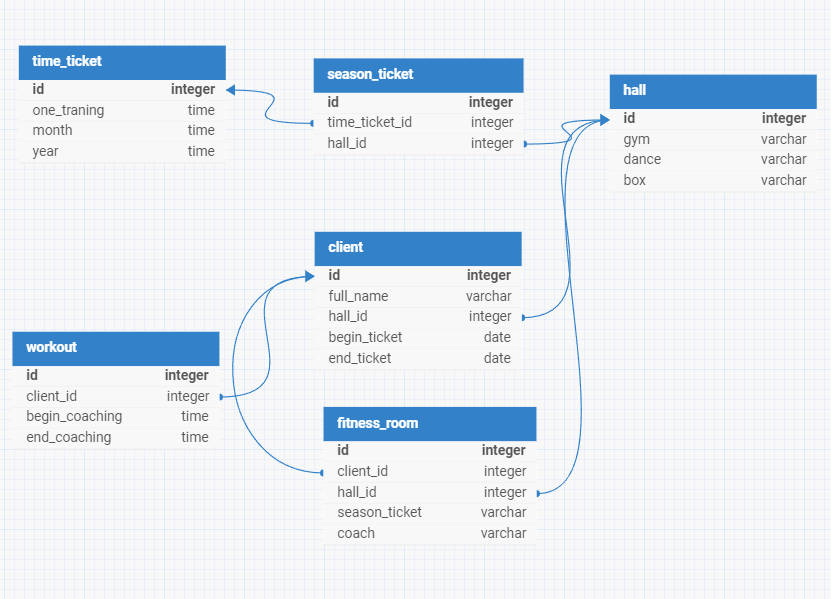


Рисунок 8 ‒ База данных на DBdesigner

2.2.1 Запросы для БД

Одним из основных типов запросов, используемых в базах данных, является запрос на выборку данных. Запросы на выборку позволяют извлекать конкретные данные из таблиц базы данных на основе определенных условий.

Запросы на выборку могут использоваться для различных целей, таких как:

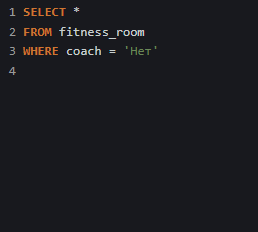
‒ получение списка клиентов;

‒начало и конец тренировки;

‒ вывод списка сотрудников, работающих в отделе парикмахерских услуг;

‒Выбор различных залов.

Используя запросы на выборку, администраторы могут спокойно пробивать информацию о клиентах и знать в какое время они приходят и уходят.



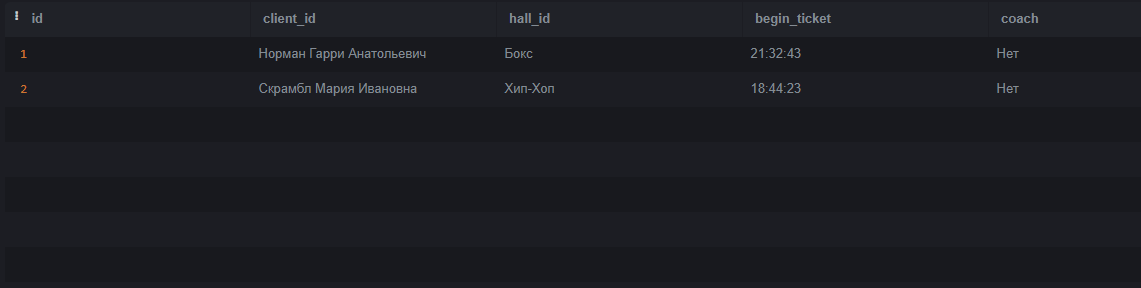


Рисунок 9,10 ‒ Результат запроса о получении имени пользователя

Вывод информации о клиенте в таблице «fitness\_room».

Запрос:

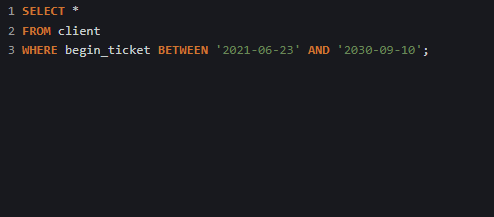
− select;

− from fitness\_room;

− where coach = ‘Нет’.

Этот запрос извлекает все строки из таблицы «fitness\_room», где значение столбца «coach» равно “Нет”. Он вернет всю информацию, хранящуюся в таблице для клиентов, которые отказались от тренера.

Условие WHERE фильтрует строки по столбцам, гарантируя, что будут возвращены только данные о клиентах, которые отказались от тренера.



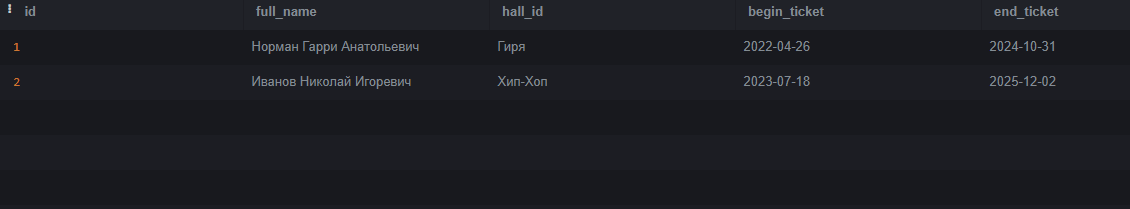


Рисунок 11,12 ‒ Результат запроса о выводе цены на товары

Вывод в таблице «client» цену закупок.

Запрос:

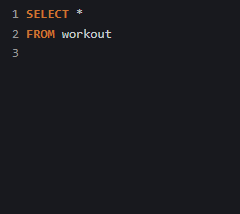
− select;

− from client;

−where begin\_ticket BETWEEN ‘2021-06-23’ AND ‘2030-09-10’.

Этот запрос извлекает все строки из таблицы «client», где значения столбца «begin\_ticket» в диапазоне от 2021-06-23 и до 2030-09-10. Он вернёт лишь те даты, которые находятся в этом диапазоне.

Условие WHERE фильтрует строки гарантируя, что будут возвращены только данные о тех датах, которые находятся в указанном диапазоне.



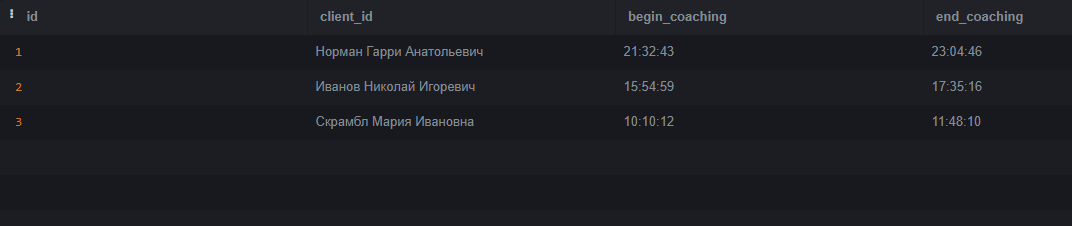


Рисунок 13,14 ‒ Результат запроса о выводе времени на запись

Вывод в таблице «workout» время и запись.

Запрос:

− select;

− from workout.

Этот запрос извлекает все строки из таблицы «workout». Он вернет всю информацию, хранящуюся в таблице, включая время и запись для каждого элемента расписания.

В данном случае запрос не использует условие WHERE, что означает, что он извлечет все строки из таблицы, не применяя никаких фильтров, и будут возвращены только данные тренировках.

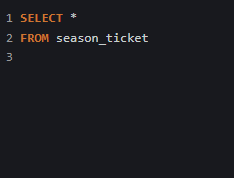




Рисунок 15,16 ‒ Результат запроса о выводе информации сотрудниках

Вывод информации о сотрудниках в таблице «season\_ticket»:

Запрос:

− select;

− from season\_ticket.

Этот запрос извлекает все строки из таблицы season\_ticket. Он вернет всю информацию, хранящуюся в таблице, для всех сотрудников.

В данном случае запрос не использует условие WHERE, что означает, что он извлечет все строки из таблицы, не применяя никаких фильтров, и будут возвращены только данные о абонементе.

ЗАКЛЮЧЕНИЕ

В процессе создания модели реляционной базы данных для парикмахерской были выявлены ключевые сущности и связи между ними. Это позволило разработать эффективный механизм хранения и управления данными, обеспечивающий быстрый доступ к информации о клиентах, услугах, персонале и других важных аспектах работы салона.

Созданная модель данных послужит основой для дальнейшего развития информационной системы парикмахерской. Она позволит автоматизировать процессы работы, повысить качество обслуживания клиентов и оптимизировать работу персонала.

Оптимизация работы персонала: Автоматизация рутинных задач позволит сотрудникам сосредоточиться на предоставлении качественных услуг клиентам, повышая общую эффективность работы.

Повышение удовлетворенности клиентов: Улучшение системы управления данными позволит парикмахерской предоставлять клиентам персонализированные услуги, учитывая их предпочтения и историю посещений.

Упрощение управления салоном: Централизация данных и автоматизированные процессы упростят управление парикмахерской, высвобождая время руководителей для стратегического планирования и развития бизнеса.

Внедрение базы данных, разработанной на основе выявленных сущностей и связей, позволит парикмахерской повысить свою эффективность, улучшить качество обслуживания и заложить основу для дальнейшего роста и развития, трансформируются в высокоэффективные и ориентированные на клиента предприятия. Они не просто улучшают свои операции, но и повышают планку для всей индустрии, устанавливая новые стандарты в области красоты и обслуживания.

Список использованных источников

1. А. И. Гайдов. Проектирование информационных систем. – М.: Издательский дом «Альпина Паблишер», 2017. – 416 с.
2. Архангельский А.Я. Программирование в Delphi 7. – М.: ООО "Бином Пресс", 2005. – 627 с.
3. Баженова И.Ю. «Основы проектирования приложений баз данных»,2009.
4. Голубев Д.А. "Реляционные базы данных: теория и практика". – М.: БХВ–Петербург, 2019.
5. К.Д. Кондратьев, С.И. Сидоренко, В.В. Карпов. Базы данных: учебное пособие. – М.: Инфра–М, 2019. – 432 с.
6. Карпов А.В. "Основы проектирования баз данных". – СПб.: Питер, 2018.
7. Кириллов В.В., Громов Г.Ю. «Введение в реляционные базы данных», 2009.
8. Кирин М. Н.”Учебник для продвинутых по Delphi 7 “. – М.: ООО "Бином–Пресс", 2003. – 496 с..
9. Климова Л.М. Delphi 7. Основы программирования. Решение типовых задач. Самоучитель. Издание третье. – М.: КУДИЙ–ОБРАЗ,2006. – 480 с.
10. Корняков В.Н. Программирование документов и приложений MS Office в Delphi. – 2005.
11. Кузнецов С.О. "Базы данных: учебное пособие". – М.: Издательский центр "Академия", 2019.
12. Л.С. Васильев. Основы проектирования баз данных. – СПб.: БХВ–Петербург, 2018. – 384 с.
13. Лукин А.В. "Реляционные базы данных: учебное пособие". – М.: Издательский центр "Академия", 2016.
14. Осипов Д.Л. Базы данных и Delphi. Теория и практика.: БХВ–Петербург, 2011.– 752 с.