Общая характеристика организации «Фитнес-Клуб»

Фитнес-клуб представляет собой современную спортивную организацию, которая активно способствует популяризации здорового образа жизни. Основная деятельность клуба сосредоточена на предоставлении разнообразных услуг в области фитнеса, включая персональные, групповые занятия и дополнительные сервисы, направленные на улучшение физического и эмоционального состояния клиентов.

В клубе имеется современный тренажерный зал, который оборудован всем необходимым для силовых и кардиотренировок. Групповые занятия охватывают широкий спектр направлений, таких как йога, аэробика, функциональные тренировки и танцевальные программы. Персональные тренировки проводятся опытными тренерами, которые разрабатывают индивидуальные программы для каждого клиента. Кроме того, клуб предлагает дополнительные услуги, такие как массаж, физиотерапия и консультации по питанию, что позволяет клиентам получать комплексный и индивидуальный подход к своему здоровью и физической форме.

Целевая аудитория клуба включает людей в возрасте от 16 лет, а также корпоративных клиентов и спортсменов-любителей. Основные запросы клиентов заключаются в доступе к разнообразным тренировочным программам, персонализированном подходе, удобном расписании и возможности онлайн-записи на занятия.

Организационная структура клуба включает несколько ключевых подразделений. Административный отдел отвечает за управление записями, клиентским сервисом и финансами. Тренерский состав состоит из инструкторов групповых программ и персональных тренеров, которые обеспечивают высокое качество тренировок. Техническая поддержка занимается обслуживанием оборудования и *IT*-инфраструктуры, а отдел маркетинга и продаж фокусируется на привлечении клиентов, рекламе и партнерских программах.

Клуб ставит перед собой различные цели, такие как повышение качества услуг через внедрение инновационных методик, увеличение клиентской базы и оптимизация внутренних процессов для снижения различных затрат. Важной задачей является создание лояльного сообщества вокруг бренда, что позволит укрепить позиции клуба на рынке.

Однако, в текущей системе управления существуют определенные проблемы. Ручной ввод данных с использованием *Excel* и бумажных журналов приводит к ошибкам и потере информации. Сложности в аналитике затрудняют быстрое формирование отчетов по посещаемости, финансам и эффективности работы тренеров. Неэффективное расписание создает конфликты в бронировании залов и тренеров, а ограниченный клиентский сервис не позволяет использовать автоматические напоминания и персонализированные предложения.

Внедрение базы данных станет важным шагом для решения этих проблем. Централизованное хранение данных позволит объединить информацию о клиентах, абонементах, расписании и платежах, а автоматизация процессов, таких как онлайн-запись и уведомления, значительно упростит работу клуба. Это также повысит точность аналитики, что позволит генерировать отчеты для принятия управленческих решений. Улучшение клиентского опыта через персонализацию услуг, создание системы лояльности станет важным аспектом в стратегии клуба.

Таким образом, внедрение базы данных не только оптимизирует управление ресурсами, но и повысит качество обслуживания клиентов, создавая прочную основу для масштабирования бизнеса. Реализация данного проекта направлена на решение ключевых проблем организации и достижение стратегических целей, что в конечном итоге приведет к успешному раз витию фитнес-клуба.

Организационно-штатная структура и модель бизнес-процессов представлены соответственно на рисунках 1, и 2,­ 3.

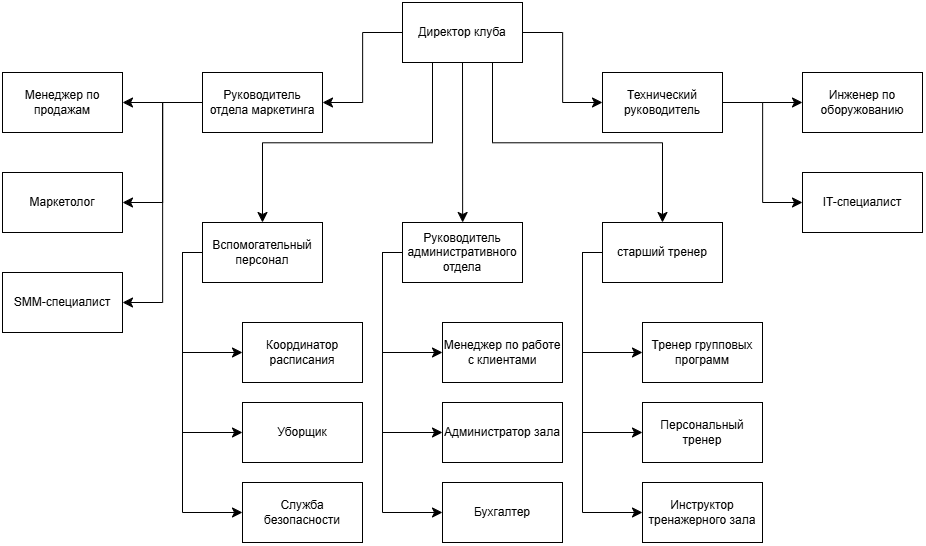


Рисунок 1 – Организационно-штатная структура фитнес-клуба

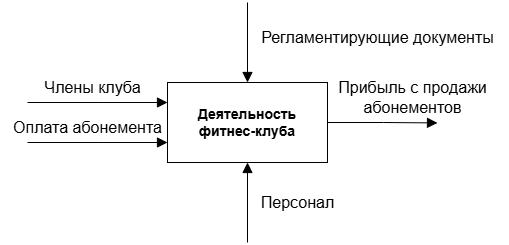


Рисунок 2 – Контекстная диаграмма бизнес-процессов фитнес-клуба

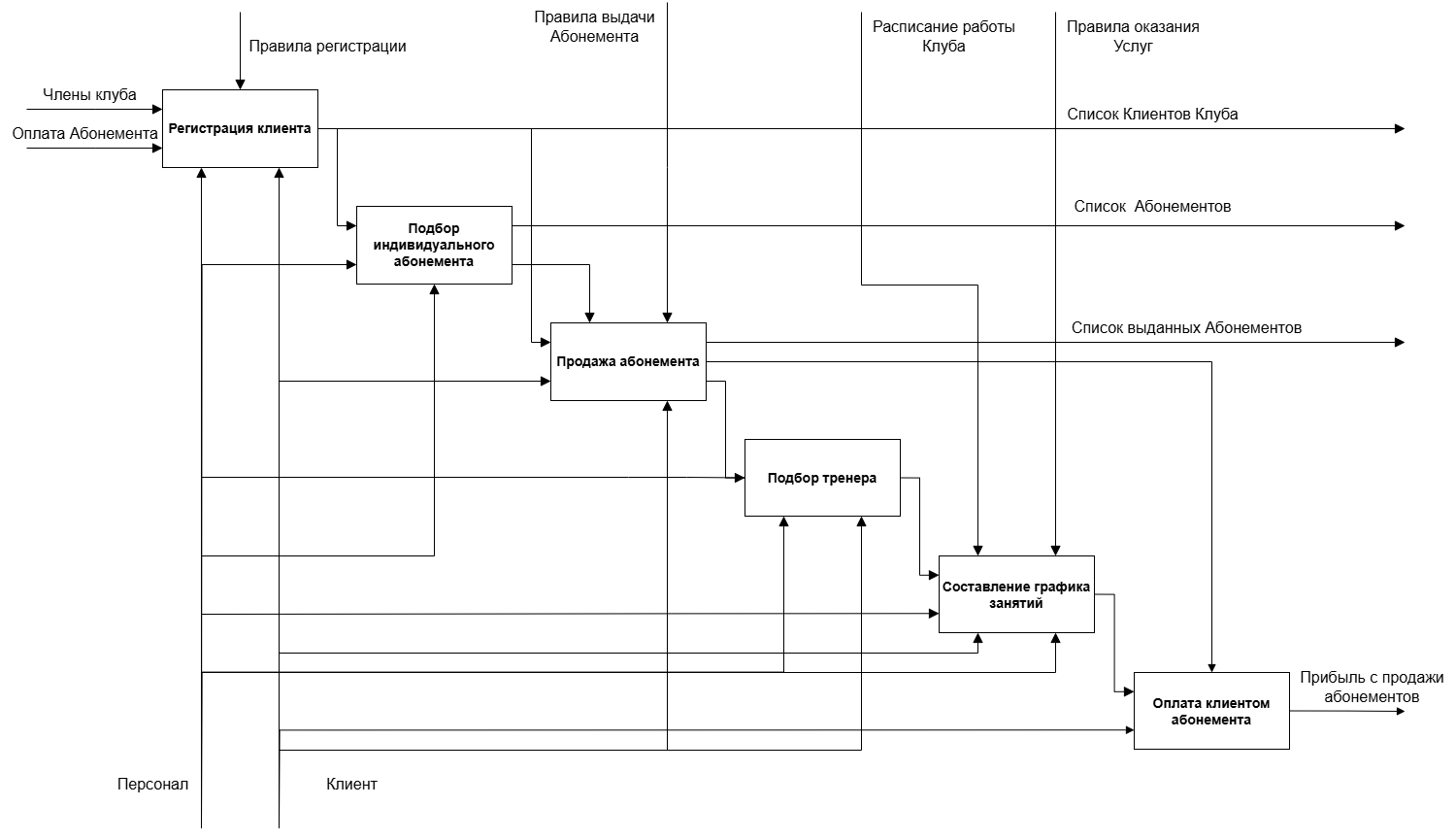


Рисунок 3 – Декомпозиция контекстной диаграммы фитнес-клуба

Следующим этапом разработки стало концептуальное проектирование. На этом этапе главной целью было определиться с сущностями будущей базы данных, их атрибутами, связями между ними, а также построить *ER*-модель, которая наглядно отображает структуру данных.

На основе анализа требований и бизнес-процессов были выделены следующие основные сущности:

1. *Clients* (Клиенты):

–*ClientID* (уникальный идентификатор клиента) – первичный ключ;

– *FirstName* (имя клиента);

– *LastName* (фамилия клиента);

– *DateOfBirth* (дата рождения клиента);

– *Phone* (телефон клиента);

– *Email* (электронная почта клиента);

– *RegistrationDate* (дата регистрации клиента).

1. *Enrollments* (Записи на занятия):

– *EnrollmentID* (уникальный идентификатор записи) – первичный ключ;

– *ClientID* (идентификатор клиента) – внешний ключ, связь с сущностью *Clients*;

– *ScheduledID* (идентификатор расписания) – внешний ключ, связь с сущностью *Schedule*;

– *ClubID* (идентификатор клуба) – внешний ключ.

– *MembershipID* (идентификатор членства) – внешний ключ, связь с сущностью *Memberships*.

– *EnrollmentDate* (дата записи на занятие).

1. *Schedule* (Расписание):

– *ScheduledID* (уникальный идентификатор расписания) – первичный ключ;

– *GroupID* (идентификатор группы) – внешний ключ, связь с сущностью *Groups*;

– *DateTime* (дата и время занятия);

– *MaxParticipants* (максимальное количество участников).

1. *Trainers* (Тренеры):

– *TrainerID* (уникальный идентификатор тренера) – первичный ключ;

– *FirstName* (имя тренера);

– *LastName* (фамилия тренера);

– *Specialization* (специализация тренера);

– *Phone* (телефон тренера);

– *Email* (электронная почта тренера).

1. *Groups* (Группы):

– *GroupID* (уникальный идентификатор группы) – первичный ключ;

– *GroupName* (название группы);

– *Description* (описание группы);

– *Duration* (продолжительность занятий);

– *TrainerID* (идентификатор тренера) – внешний ключ, связь с сущностью *Trainers*.

1. *Memberships* (Членства):

– *MembershipID* (уникальный идентификатор членства) — первичный ключ;

– *MembershipName* (название членства);

– *DurationDays* (продолжительность членства в днях);

– *AccessTime* (время доступа).

На этапе концептуального проектирования были определены следующие связи между сущностями:

*Clients* и *Enrollments*: Один клиент может быть записан на несколько занятий, но каждая запись относится только к одному клиенту. Связь 1:*M*.

*Enrollments* и *Schedule*: Каждая запись на занятие связана с одним расписанием, но одно расписание может иметь несколько записей. Связь 1:*M*.

*Enrollments* и *Memberships*: Каждая запись на занятие связана с одним членством, но одно членство может быть использовано в нескольких записях. Связь 1:*M*.

*Groups* и *Schedule*: Одна группа может иметь несколько расписаний, но каждое расписание относится только к одной группе. Связь 1:*M*.

*Trainers* и *Groups*: Один тренер может вести несколько групп, но каждая группа ведется только одним тренером. Связь 1:*M*.

Сама *ER*-модель была построена на основе выделенных сущностей и связей между ними. Каждая сущность была представлена в виде прямоугольника с указанием атрибутов, а связи между сущностями обозначены линиями с указанием типа связи (1:*M*).*ER*-модель представлена на рисунке 4.

Для каждой сущности были выбраны атрибуты, которые наиболее полно описывали бы её свойства. Первичные ключи были определены для идентификации каждой записи в таблице. В качестве первичных ключей использовались уникальные идентификаторы (например, *ClientID*, *EnrollmentID* и т.д.), которые гарантируют уникальность каждой записи.

Таким образом, на этапе концептуального проектирования мной была разработана структура базы данных, которая учитывает все необходимые сущности, их атрибуты и связи между ними.

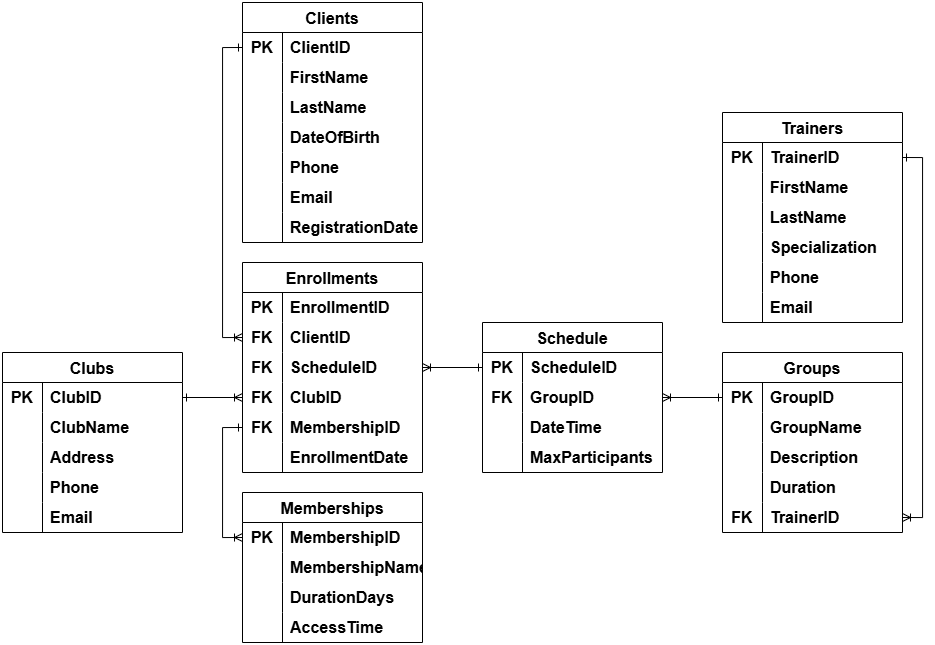


Рисунок 4 – *ER*-модель фитнес-клуба

После концептуального проектирования я перешёл к этапу логического проектирования. На это этапе я преобразовал концептуальную модель в набор таблиц будущей БД, применив правила преобразования *ER*-модели в реляционную. Всего существует 6 правил, но в данном случае понадобилось только 4-ое.

Оно гласит, что «Если связь типа 1:М и класс принадлежности сущности на стороне М является обязательным, то необходимо построить таблицу для каждой сущности…».

Поскольку все таблицы имеют тип связи 1:М, то и применяться к ним будет одно и то же правило.

Так же для более гибкой работы была добавлена новая таблица *Specializations*, которая отвечает за специализации тренеров.

Следующим этапом логического проектирования и преобразований должна была стать нормализация, осуществляемая последовательно начиная с 1 нормальной формы и заканчивая 3 нормальной формой. Для этого я начал внимательно анализировать каждую таблицу и проверять соответствуют ли они требованиям нормализации: и пришел к выводу: что таблицы уже находятся в 3 нормальной форме, так как, транзитивные зависимости отсутствуют. Например:

Таблица *Clients*:

* + все значения в ячейках данной таблицы атомарны, следовательно, таблица находится в 1 нормальной форме;
  + первичный ключ таблицы *Clients* состоит из одного атрибута, следовательно, в данной таблице все неключевые атрибуты полностью зависят от первичного ключа, и поэтому таблица находится во 2 нормальной форме;
  + между неключевыми атрибутами данной таблицы (*FirstName*, *LastName*, *DateOfBirth* и т.д.) отсутствуют транзитивные зависимости, следовательно, данная таблица находится в 3 нормальной форме.

Таблица *Clubs*:

* + все значения в ячейках данной таблицы атомарны, следовательно, таблица находится в 1 нормальной форме;
  + первичный ключ таблицы *Clubs* состоит из одного атрибута, следовательно, в данной таблице все неключевые атрибуты полностью зависят от первичного ключа, и поэтому таблица находится во 2 нормальной форме;
  + между неключевыми атрибутами данной таблицы (*ClubName*, *Adress*, *Phone* и*Email*) отсутствуют транзитивные зависимости, следовательно, данная таблица находится в 3 нормальной форме.

Таблица *Enrollments*:

* + все значения в ячейках данной таблицы атомарны, следовательно, таблица находится в 1 нормальной форме;
  + первичный ключ таблицы *Enrollments* состоит из одного атрибута, следовательно, в данной таблице все неключевые атрибуты полностью зависят от первичного ключа, и поэтому таблица находится во 2 нормальной форме;
  + между неключевыми атрибутами данной таблицы отсутствуют транзитивные зависимости, следовательно, данная таблица находится в 3 нормальной форме.

Таблица *Memberships*:

* + все значения в ячейках данной таблицы атомарны, следовательно, таблица находится в 1 нормальной форме;
  + первичный ключ таблицы *Memberships* состоит из одного атрибута, следовательно, в данной таблице все неключевые атрибуты полностью зависят от первичного ключа, и поэтому таблица находится во 2 нормальной форме;
  + между неключевыми атрибутами данной таблицы отсутствуют транзитивные зависимости, следовательно, данная таблица находится в 3 нормальной форме.

Таблица *Shedules*:

* + все значения в ячейках данной таблицы атомарны, следовательно, таблица находится в 1 нормальной форме;
  + первичный ключ таблицы *Shedules* состоит из одного атрибута, следовательно, в данной таблице все неключевые атрибуты полностью зависят от первичного ключа, и поэтому таблица находится во 2 нормальной форме;
  + между неключевыми атрибутами данной таблицы отсутствуют транзитивные зависимости, следовательно, данная таблица находится в 3 нормальной форме.

Таблица *Groups*:

* + все значения в ячейках данной таблицы атомарны, следовательно, таблица находится в 1 нормальной форме;
  + первичный ключ таблицы *Groups* состоит из одного атрибута, следовательно, в данной таблице все неключевые атрибуты полностью зависят от первичного ключа, и поэтому таблица находится во 2 нормальной форме;
  + между неключевыми атрибутами данной таблицы (*GroupName*, *Description*, *Duration*) отсутствуют транзитивные зависимости, следовательно, данная таблица находится в 3 нормальной форме.

Таблица *Trainers*:

* + все значения в ячейках данной таблицы атомарны, следовательно, таблица находится в 1 нормальной форме;
  + первичный ключ таблицы *Trainers* состоит из одного атрибута, следовательно, в данной таблице все неключевые атрибуты полностью зависят от первичного ключа, и поэтому таблица находится во 2 нормальной форме;
  + между неключевыми атрибутами данной таблицы (*FirstName*, *LastName*, *Phone* и *Email*) отсутствуют транзитивные зависимости, следовательно, данная таблица находится в 3 нормальной форме.

Таблица *Specializations*:

* + все значения в ячейках данной таблицы атомарны, следовательно, таблица находится в 1 нормальной форме;
  + первичный ключ таблицы *Specializations* состоит из одного атрибута, следовательно, в данной таблице все неключевые атрибуты полностью зависят от первичного ключа, и поэтому таблица находится во 2 нормальной форме;
  + между неключевыми атрибутами данной таблицы отсутствуют транзитивные зависимости, следовательно, данная таблица находится в 3 нормальной форме.

Таким образом, каждая таблица содержит только атомарные данные, все неключевые атрибуты зависят от первичного ключа, а транзитивные зависимости устранены за счёт правильного использования внешних ключей и разделения сущностей. Это подтверждает соответствие схемы требованиям третьей нормальной формы. Результат логического проектирования представлен на рисунке 5.

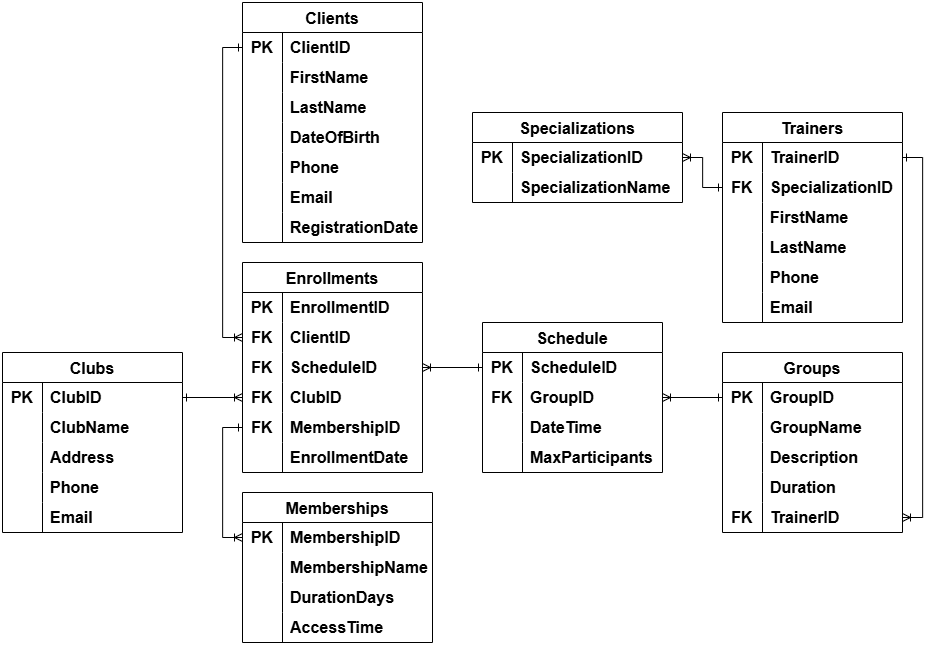


Рисунок 5 – Логическая модель фитнес-клуба

После всех прошедших этапов я приступил к физическому проектированию в среде СУБД *MySQL*базы данных фитнес-клуба. Для этого я создал базу данных и поочередно стал создавать все таблицы и поля для них, опираясь на мою логическую модель.

Создание базы данных выполнялось поэтапно, следующим образом:

1. Создание базы данных. Для создания базы данных и перехода в неё используются команды*CREATEDATABASE*и *USE*, соответственно, с указанием имени базы данных. Например:

*CREATE DATABASE fitness*\_*club*;

*USE fitness*\_*club*;

1. Определение таблиц и полей. Для каждой получившейся сущности создана соответствующая таблица. Поля объявлены с указанием типов данных.
2. Первичные и внешние ключи. Каждая таблица включает первичный ключ (*PRIMARYKEY*), соответствующий атрибуту «*PK»*из логической модели. Внешние ключи (*FOREIGNKEY*), соответствующие атрибуту «*FK»*из логической модели, используются для установления связей между таблицами.
3. Реализация кода с учетом ограничений. На данном этапе я приступил непосредственно к реализации кода. Во время его написания были учтены различные ограничения: ограничения длины строк (например, *VARCHAR*(50)) и ограничения на пустую строку (*NOTNULL*). Данные ограничения были необходимы воизбежание ошибок в будущем.

Итоговый *SQL* код отражает структуру логической модели, включая все сущности, атрибуты и связи, что обеспечивает корректную работу базы данных для фитнес-клуба.

Листинг итогового кода базы данных представлен в Приложении А (см. стр. 15).

Таблица 1 – Таблица сущности *Clients*с описанием атрибутов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название атрибута** | **Тип атрибута** | **Диапазон значений** | **Значение по умолчанию** | ***Null*-значение** | **Является ли атрибут расчетным** | **Бизнес правила** |
| *ClientID\** | *INT* | 1–2147483647 | Нет | Нет | Нет | Отсутствуют |
| *FirstName* | *VARCHAR*(50) | Строка до 50 символов | Нет | Нет | Нет | Отсутствуют |
| *LastName* | *VARCHAR*(50) | Строка до 50 символов | Нет | Нет | Нет | Отсутствуют |
| *DateOfBirth* | *DATE* | 1900-01-01 – текущая дата | Нет | Нет | Нет | Отсутствуют |
| *Phone* | *VARCHAR*(15) | Строка до 15 символов | Нет | Нет | Нет | Отсутствуют |
| *Email* | *VARCHAR*(100) | Строка до 100 символов | Нет | Да | Нет | Отсутствуют |
| *RegistrationDate* | *DATE* | Текущая дата | Нет | Нет | Нет | Отсутствуют |

\* – является внешним первичным ключом.

\*\* – является внешним ключом.

Таблица 2 – Таблица сущности *Memberships*с описанием атрибутов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название атрибута** | **Тип атрибута** | **Диапазон значений** | **Значение по умолчанию** | ***Null*-значение** | **Является ли атрибут расчетным** | **Бизнес правила** |
| *MembershipID\** | *INT* | 1–2147483647 | Нет | Нет | Нет | Отсутствуют |
| *MembershipName* | *VARCHAR*(100) | Строка до 100 символов | Нет | Да | Нет | Отсутствуют |
| *DurationDays* | *INT* | 1–365 | Нет | Да | Нет | Отсутствуют |
| *AccessTime* | *TIME* | 00:00:00 – 23:59:59 | Нет | Да | Нет | Отсутствуют |

\* – является внешним первичным ключом.

\*\* – является внешним ключом.

Таблица 3 – Таблица сущности *Trainers*с описанием атрибутов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название атрибута** | **Тип атрибута** | **Диапазон значений** | **Значение по умолчанию** | ***Null*-значение** | **Является ли атрибут расчетным** | **Бизнес правила** |
| *TrainerID\** | *INT* | 1–2147483647 | Нет | Нет | Нет | Отсутствуют |
| *FirstName* | *VARCHAR*(50) | Строка до 50 символов | Нет | Нет | Нет | Отсутствуют |
| *LastName* | *VARCHAR*(50) | Строка до 50 символов | Нет | Нет | Нет | Отсутствуют |
| *Phone* | *VARCHAR*(15) | Строка до 15 символов | Нет | Нет | Нет | Отсутствуют |
| *Email* | *VARCHAR*(100) | Строка до 100 символов | Нет | Да | Нет | Отсутствуют |
| *SpecializationID\*\** | *INT* | 1–2147483647 | Нет | Нет | Нет | Отсутствуют |

\* – является внешним первичным ключом.

\*\* – является внешним ключом.

*Trainers*Таблица 4 – Таблица сущности *Groups*с описанием атрибутов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название атрибута** | **Тип атрибута** | **Диапазон значений** | **Значение по умолчанию** | ***Null*-значение** | **Является ли атрибут расчетным** | **Бизнес правила** |
| *GroupID\** | *INT* | 1–2147483647 | Нет | Нет | Нет | Отсутствуют |
| *GroupName* | *VARCHAR*(100) | Строка до 100 символов | Нет | Нет | Нет | Отсутствуют |
| *Description* | *TEXT* | Произвольный текст | Нет | Да | Нет | Отсутствуют |
| *Duration* | *INT* | 1–24 (часы) | Нет | Нет | Нет | Отсутствуют |
| *TrainerID\*\** | *INT* | 1–2147483647 | Нет | Нет | Нет | Отсутствуют |

\* – является внешним первичным ключом.

\*\* – является внешним ключом.

Таблица 5 – Таблица сущности *Schedule*с описанием атрибутов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название атрибута** | **Тип атрибута** | **Диапазон значений** | **Значение по умолчанию** | ***Null*-значение** | **Является ли атрибут расчетным** | **Бизнес правила** |
| *ScheduledID\** | *INT* | 1–2147483647 | Нет | Нет | Нет | Отсутствуют |
| *GroupID\*\** | *INT* | 1–2147483647 | Нет | Да | Нет | Отсутствуют |
| *DateTime* | *DATETIME* | Дата и время | Нет | Да | Нет | Отсутствуют |
| *MaxParticipants* | *INT* | 1–100 | Нет | Да | Нет | Отсутствуют |

\* – является внешним первичным ключом.

\*\* – является внешним ключом.

Таблица 6 – Таблица сущности *Specializations*с описанием атрибутов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название атрибута** | **Тип атрибута** | **Диапазон значений** | **Значение по умолчанию** | ***Null*-значение** | **Является ли атрибут расчетным** | **Бизнес правила** |
| *SpecializationID\** | *INT* | 1–2147483647 | Нет | Нет | Нет | Отсутствуют |
| *SpecializationName* | *VARCHAR*(100) | Строка до 100 символов | Нет | Да | Нет | Отсутствуют |

\* – является внешним первичным ключом.

\*\* – является внешним ключом.

Таблица 7 – Таблица сущности *Enrollments*с описанием атрибутов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название атрибута** | **Тип атрибута** | **Диапазон значений** | **Значение по умолчанию** | ***Null*-значение** | **Является ли атрибут расчетным** | **Бизнес правила** |
| *EnrollmentID\** | *INT* | 1–2147483647 | Нет | Нет | Нет | Отсутствуют |
| *ClientID\*\** | *INT* | 1–2147483647 | Нет | Да | Нет | Отсутствуют |
| *ScheduledID\*\** | *INT* | 1–2147483647 | Нет | Да | Нет | Отсутствуют |
| *ClubID\*\** | *INT* | 1–2147483647 | Нет | Да | Нет | Отсутствуют |
| *MembershipID\*\** | *INT* | 1–2147483647 | Нет | Да | Нет | Отсутствуют |
| *EnrollmentDate* | *DATE* | Текущая дата | Нет | Нет | Нет | Отсутствуют |

\* – является внешним первичным ключом.

\*\* – является внешним ключом.

Таблица 8 – Таблица сущности *Clubs*с описанием атрибутов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название атрибута** | **Тип атрибута** | **Диапазон значений** | **Значение по умолчанию** | ***Null*-значение** | **Является ли атрибут расчетным** | **Бизнес правила** |
| *ClubID\** | *INT* | 1–2147483647 | Нет | Нет | Нет | Отсутствуют |
| *ClubName* | *VARCHAR*(100) | Строка до 100 символов | Нет | Нет | Нет | Отсутствуют |
| *Address* | *VARCHAR*(255) | Строка до 255 символов | Нет | Нет | Нет | Отсутствуют |
| *Phone* | *VARCHAR*(15) | Строка до 15 символов | Нет | Нет | Нет | Отсутствуют |
| *Email* | *VARCHAR*(100) | Строка до 100 символов | Нет | Нет | Нет | Отсутствуют |

\* – является внешним первичным ключом.

\*\* – является внешним ключом.

Приложение А

Листинг кода на языке *SQL* по созданию базы данных фитнес-клуба

*CREATE DATABASE fitness\_club;*

*USE fitness\_club;*

*CREATE TABLE Clients (*

*ClientID INT PRIMARY KEY AUTO\_INCREMENT,*

*FirstName VARCHAR(50) NOT NULL,*

*LastName VARCHAR(50) NOT NULL,*

*DateOfBirth DATE NOT NULL,*

*Phone VARCHAR(15) NOT NULL,*

*Email VARCHAR(100),*

*RegistrationDate DATE NOT NULL*

*);*

*CREATE TABLE Memberships (*

*MembershipID INT PRIMARY KEY AUTO\_INCREMENT,*

*MembershipName VARCHAR(100),*

*DurationDays INT,*

*AccessTime TIME*

*);*

*CREATE TABLE Trainers (*

*TrainerID INT PRIMARY KEY AUTO\_INCREMENT,*

*FirstName VARCHAR(50) NOT NULL,*

*LastName VARCHAR(50) NOT NULL,*

*Phone VARCHAR(15) NOT NULL,*

*Email VARCHAR(100),*

*SpecializationID INT NOT NULL,*

*FOREIGN KEY (SpecializationID) REFERENCES Specializations(SpecializationID)*

*);*

*CREATE TABLE Groups (*

*GroupID INT PRIMARY KEY AUTO\_INCREMENT,*

*GroupName VARCHAR(100) NOT NULL,*

*Description TEXT,*

*Duration INT NOT NULL,*

*TrainerID INT NOT NULL,*

*FOREIGN KEY (TrainerID) REFERENCES Trainers(TrainerID)*

*);*

*CREATE TABLE Schedule (*

*ScheduledID INT PRIMARY KEY AUTO\_INCREMENT,*

*GroupID INT,*

*DateTime DATETIME,*

*MaxParticipants INT,*

*FOREIGN KEY (GroupID) REFERENCES Groups(GroupID)*

*);*

*CREATE TABLE Specializations (*

*SpecializationID INT PRIMARY KEY AUTO\_INCREMENT,*

*SpecializationName VARCHAR(100)*

*);*

*CREATE TABLE Enrollments (*

*EnrollmentID INT PRIMARY KEY AUTO\_INCREMENT,*

*ClientID INT,*

*ScheduledID INT,*

*ClubID INT,*

*MembershipID INT,*

*EnrollmentDate DATE NOT NULL,*

*FOREIGN KEY (ClientID) REFERENCES Clients(ClientID),*

*FOREIGN KEY (ScheduledID) REFERENCES Schedule(ScheduledID),*

*FOREIGN KEY (ClubID) REFERENCES Clubs(ClubID),*

*FOREIGN KEY (MembershipID) REFERENCES Memberships(MembershipID)*

*);*

*CREATE TABLE Clubs (*

*ClubID INT PRIMARY KEY AUTO\_INCREMENT,*

*ClubName VARCHAR(100) NOT NULL,*

*Address VARCHAR(255) NOT NULL*

*);*

Приложение Б

Листинг кода на языке *SQL* для заполнения базы данных фитнес-клуба

*USE fitness\_club;*

*INSERT INTO Clubs (ClubName, Address) VALUES*

*('Fitness City', 'ул. Центральная, 1'),*

*('Fitness City Pro', 'ул. Западная, 5'),*

*('Fitness City', 'пр. Восточный, 10'),*

*('Fitness City Pro', 'ул. Северная, 3'),*

*('Fitness City', 'ул. Южная, 7'),*

*('Fitness City Plus', 'ул. Парковая, 15'),*

*('Fitness City Elite', 'пр. Ленина, 20'),*

*('Fitness City Junior', 'ул. Школьная, 30'),*

*('Fitness City Express', 'ул. Вокзальная, 8'),*

*('Fitness City Premium', 'пр. Мира, 45');*

*INSERT INTO Specializations (SpecializationName) VALUES*

*('Йога'),*

*('Пилатес'),*

*('Функциональный тренинг'),*

*('Кроссфит'),*

*('Стретчинг');*

*INSERT INTO Trainers (FirstName, LastName, Phone, Email, SpecializationID) VALUES*

*('Анна', 'Волкова', '79530000001', 'a.volkova@fitpro.ru', 1),*

*('Максим', 'Соколов', '79530000002', 'm.sokolov@fitpro.ru', 3),*

*('Иван', 'Иванов', '79530000003', 'i.ivanov@fitpro.ru', 2),*

*('Артем', 'Крылов', '79530000004', 'a.krylov@fitpro.ru', 4),*

*('Олеся', 'Мельникова', '79530000005', 'o.melnikova@fitpro.ru', 5),*

*('Елена', 'Петрова', '79530000006', 'e.petrova@fitpro.ru', 1),*

*('Дмитрий', 'Федоров', '79530000007', 'd.fedorov@fitpro.ru', 2),*

*('Светлана', 'Игнатова', '79530000008', 's.ignatova@fitpro.ru', 3),*

*('Павел', 'Белов', '79530000009', 'p.belov@fitpro.ru', 4),*

*('Ольга', 'Кузнецова', '79530000010', 'o.kuznetsova@fitpro.ru', 5);*

*INSERT INTO Groups (GroupName, Description, Duration, TrainerID) VALUES*

*('Рассветная йога', 'Утренние практики на рассвете', 60, 1),*

*('Силовые горизонты', 'Работа с собственным весом', 90, 2),*

*('Пилатес-детокс', 'Очищающие комплексы', 75, 3),*

*('Кроссфит 2.0', 'Новые методики', 120, 4),*

*('Гибкое тело', 'Глубокий стретчинг', 60, 5),*

*('Вечерняя йога', 'Расслабляющие практики вечером', 60, 6),*

*('Функциональный микс', 'Смешанные тренировки', 90, 7),*

*('Пилатес для начинающих', 'Базовые упражнения', 75, 8),*

*('Кроссфит интенсив', 'Высокая нагрузка', 120, 9),*

*('Стретчинг утро', 'Утренние упражнения на гибкость', 60, 10);*

*INSERT INTO Schedule (GroupID, DateTime, MaxParticipants) VALUES*

*(1, '2025-01-07 07:30:00', 12),*

*(2, '2025-01-14 19:00:00', 8),*

*(3, '2025-02-02 10:00:00', 10),*

*(4, '2025-02-18 20:00:00', 6),*

*(5, '2025-02-28 18:30:00', 15),*

*(6, '2025-03-01 08:00:00', 10),*

*(7, '2025-03-05 18:00:00', 12),*

*(8, '2025-03-10 09:30:00', 15),*

*(9, '2025-03-15 19:30:00', 8),*

*(10, '2025-03-20 07:00:00', 20);*

*INSERT INTO Memberships (MembershipName, DurationDays, Accesstime) VALUES*

*('Утренний PRO', 30, '06:00:00'),*

*('Полный день', 90, '00:00:00'),*

*('Вечерний PRO', 30, '17:00:00'),*

*('Интенсивы', 14, '08:00:00'),*

*('Годовой PRO', 365, '05:00:00'),*

*('Ночной доступ', 30, '22:00:00'),*

*('Уикенд PRO', 60, '08:00:00'),*

*('Корпоративный', 180, '00:00:00'),*

*('Студенческий', 90, '10:00:00'),*

*('VIP доступ', 365, '00:00:00');*

*INSERT INTO Clients (FirstName, LastName, DateOfBirth, Phone, Email, RegistrationDate) VALUES*

*('Ирина', 'Соколова', '1998-12-05', '79170000001', 'sokol@mail.com', '2025-01-05'),*

*('Глеб', 'Крутов', '2001-04-22', '79170000002', 'krytoi@mail.com', '2025-01-12'),*

*('Полина', 'Ветрова', '1995-08-17', '79170000003', 'windy@mail.com', '2025-01-19'),*

*('Станислав', 'Морозов', '1989-02-28', '79170000004', 'frost@mail.com', '2025-01-26'),*

*('Алиса', 'Медведева', '2003-07-14', '79170000005', 'alice@mail.com', '2025-02-02'),*

*('Андрей', 'Смирнов', '1990-05-10', '79170000006', 'a.smirnov@mail.com', '2025-02-10'),*

*('Мария', 'Ковалева', '1985-09-15', '79170000007', 'm.kovaleva@mail.com', '2025-02-12'),*

*('Денис', 'Павлов', '2000-03-25', '79170000008', 'd.pavlov@mail.com', '2025-02-18'),*

*('Екатерина', 'Никитина', '1993-11-30', '79170000009', 'e.nikitina@mail.com', '2025-02-20'),*

*('Артур', 'Жуков', '1997-07-07', '79170000010', 'a.zhukov@mail.com', '2025-02-25');*

*INSERT INTO Enrollments (ClientID, ScheduledID, ClubID, MembershipID, EnrollmentDate) VALUES*

*(1, 1, 2, 1, '2025-01-01'),*

*(2, 2, 1, 2, '2025-01-10'),*

*(3, 3, 3, 3, '2025-01-20'),*

*(4, 4, 4, 4, '2025-02-05'),*

*(5, 5, 5, 5, '2025-02-15'),*

*(6, 6, 6, 6, '2025-03-01'),*

*(7, 7, 7, 7, '2025-03-05'),*

*(8, 8, 8, 8, '2025-03-10'),*

*(9, 9, 9, 9, '2025-03-15'),*

*(10, 10, 10, 10, '2025-03-20');*

Приложение В

Листинг кода на языке *SQL* с реализацией триггеров и запросов к базе данных фитнес-клуба

*USE fitness*\_*club*;

*SELECT* *FirstName*, *LastName*, *Email*

*FROM* *Clients*;

*SELECT* \*

*FROM* *Clients*

*WHERE*

*YEAR*(*RegistrationDate*) = 2025

*AND* *MONTH*(*RegistrationDate*) = 02;

*SELECT* *FirstName*, *LastName*, *DateOfBirth*

*FROM* *Clients*

*ORDER* *BY* *RegistrationDate* *ASC*;

*SELECT* *GroupName*, *Duration*

*FROM* *Groups*

*ORDER* *BY* *Duration* *DESC*;

*DELIMITER* $$

*CREATE* *TRIGGER* *trg*\_*check*\_*client*\_*email*

*BEFORE* *INSERT* *ON* *Clients*

*FOR* *EACH* *ROW*

*BEGIN*

*IF* *NEW*.*Email* *NOT* *LIKE* '%@%.\_%' *THEN*

*SIGNAL* *SQLSTATE* '45000'

*SET* *MESSAGE*\_*TEXT* = 'Некорректный формат *email*';

*END* *IF*;

*END*$$

*DELIMITER* ;

*DELIMITER* $$

*CREATE* *TRIGGER* *trg*\_*prevent*\_*duplicate*\_*enrollment*

*BEFORE* *INSERT* *ON* *Enrollments*

*FOR* *EACH* *ROW*

*BEGIN*

*IF* *EXISTS* (

*SELECT* 1

*FROM* *Enrollments*

*WHERE*

*ClientID* = *NEW*.*ClientID*

*AND* *ScheduledID* = *NEW*.*ScheduledID*

) *THEN*

*SIGNAL* *SQLSTATE* '45000'

*SET* *MESSAGE*\_*TEXT* = 'Клиент уже записан на это занятие';

*END* *IF*;

*END*$$

*DELIMITER* ;

*DELIMITER* $$

*CREATE* *TRIGGER* *trg*\_*check*\_*enrollment*\_*date*

*BEFORE* *INSERT* *ON* *Enrollments*

*FOR* *EACH* *ROW*

*BEGIN*

*IF* *NEW*.*EnrollmentDate* < *CURDATE*() *THEN*

*SIGNAL* *SQLSTATE* '45000'

*SET* *MESSAGE*\_*TEXT* = 'Дата записи не может быть в прошлом';

*END* *IF*;

*END*$$

*DELIMITER*;

SELECT FirstName, LastName, 'Client' AS Type

FROM Clients

UNION

SELECT FirstName, LastName, 'Trainer'

FROM Trainers

ORDER BY LastName, FirstName;

SELECT

s.DateTime,

g.GroupName,

t.FirstName AS TrainerFirstName,

t.LastName AS TrainerLastName,

sp.SpecializationName

FROM Schedule s

JOIN Groups g ON s.GroupID = g.GroupID

JOIN Trainers t ON g.TrainerID = t.TrainerID

JOIN Specializations sp ON t.SpecializationID = sp.SpecializationID;

SELECT

c.FirstName,

c.LastName,

cl.ClubName,

m.MembershipName,

s.DateTime AS TrainingTime

FROM Enrollments e

JOIN Clients c ON e.ClientID = c.ClientID

JOIN Clubs cl ON e.ClubID = cl.ClubID

JOIN Memberships m ON e.MembershipID = m.MembershipID

LEFT JOIN Schedule s ON e.ScheduledID = s.ScheduledID;

SELECT

cl.ClubName,

COUNT(e.EnrollmentID) AS TotalEnrollments

FROM Clubs cl

LEFT JOIN Enrollments e ON cl.ClubID = e.ClubID

GROUP BY cl.ClubName;