

Экзаменационный проект по
дисциплине Базы данных
студента гр. **М3435**
Григорьева Галина Алексеевна
по теме
Сеть фитнес-клубов

Построение отношений

В результате предварительного проектирования были выделены следующие отношения:

- **Club** - один из клубов сети
- **Address** - адрес клуба. Вероятно, при развитии БД для данной таблицы также найдутся другие применения.
- **Client** - клиент клуба (если у него есть абонемент) или потенциальный клиент (если у него еще нет абонемента, но его данные есть в базе, например, для рекламной рассылки).
- **Membership** - абонемент. В том числе храним абонементы, срок которых подошел к концу. Клиент может иметь несколько абонементов в разные клубы сети.
- **Trainer** - тренер, работающий в одном из клубов сети. Проводит индивидуальные платные тренировки. Цена тренировки у разных тренеров различается по субъективному решению руководства клуба (известность, спортивный разряд и т.д.). Также проводит групповые занятия в клубе.
- **Training** - индивидуальная тренировка с тренером. Храним цену тренировки для отчетности, так как с течением времени цены тренеров могут меняться.
- **Room** - комната для групповых занятий.
- **GroupWorkout** - групповое занятие (йога, пилатес и т.д.). Проходит в определенный день недели в определенное время. Занятия в разное время недели не связаны между собой, могут иметь одинаковые названия, их могут проводить те же или другие тренеры. Вход свободный без записи.
- **Equipment** - оборудование, принадлежащее клубу.
- **Pool** - бассейн клуба. Также в клубе может не быть бассейна или их может быть несколько.

- Отношение **Club**

Атрибуты:

- ClubId
- ClubName
- AddressId

Функциональные зависимости:

- ClubId -> ClubName, AddressId
- ClubName -> ClubId

Ключ {ClubId}, {ClubName}

Других ключей быть не может, так как расширение данных ключей не имеет смысла, а атрибут AddressId не встречается в левых частях фз.

1. Все атрибуты атомарны, нет повторяющихся групп, есть ключ, значит отношение находится в 1НФ.
2. Отношение находится в 1НФ и нет фз, зависящих от части ключа, поэтому отношение находится в 2НФ.
3. Неключевые атрибуты непосредственно зависят от ключей (потому что все левые части являются ключами) и отношение находится в 2НФ, значит, и в 3НФ.
4. Отношение находится в 3НФ и нет перекрывающихся ключей, значит, находится в НФБК.
5. Отношение находится в НФБК и имеет простой ключ {ClubId}, значит, находится в 4НФ по второй теореме Дейта-Фейгина.
6. Отношение находится в 4НФ и все ключи простые, значит, находится в 5НФ по первой теореме Дейта-Фейгина.

- Отношение **Address**

Атрибуты:

- AddressId
- Country
- Region
- City
- Street
- Building

Функциональные зависимости:

- AddressId -> Country, Region, City, Street, Building, Postcode

Ключ {AddressId}. Других ключей быть не может так как существует единственная ФЗ с этим ключом в левой части, при расширении это множество перестанет быть минимальным надключом.

1-6. Аналогично отношению Club.

Отношение **Client**

Атрибуты:

- ClientId
- ClientName
- Birthday
- Email

Функциональные зависимости:

- ClientId -> ClientName, Birthday, Email

Ключ {ClientId}. Нет ключей, не включающих ClientId, т.к. ClientId - единственный атрибут, стоящий в левой части. Любое другое множество атрибутов, включающее ClientId не будет минимальным надключом.

1-6. Аналогично отношению Club

- Отношение **Membership**

Атрибуты:

- MembershipId
- Until
- NumberOfVisits
- ClubId
- ClientId

Функциональные зависимости:

MembershipId -> Until, FrozenUntil, NumberOfVisits, ClubId, ClientId

Ключ {MembershipId}. Единственность аналогично отношению Client.

1-6. Аналогично отношению Club.

- Отношение **Trainer**

Атрибуты:

- TrainerId
- TrainerName
- SportsCategory
- CurrentPrice
- Salary
- ClubId

Функциональные зависимости:

TrainerId -> TrainerName, SportsCategory, CurrentPrice, Salary, ClubId

Ключ {TrainerId}. Единственность аналогично отношению Client.

1-6. Аналогично отношению Club.

- Отношение **Training**

Атрибуты:

- TrainingDate
- StartTime
- Price
- ClientId
- TrainerId

Функциональные зависимости:

TrainingDate, StartTime, TrainerId -> Price

TrainingDate, StartTime, TrainerId -> ClientId

TrainingDate, StartTime, ClientId -> Price
TrainingDate, StartTime, ClientId -> TrainerId

Ключи {TrainingDate, StartTime, TrainerId}, {TrainingDate, StartTime, ClientId}.
При уменьшении числа атрибутов в этих множествах из них становится невозможно получить полный список атрибутов (т.к. в левых частях ФЗ стоят только эти множества), при расширении этих множеств они перестают быть минимальными по включению надключами. Другие атрибуты в ключах быть не могут, поскольку не присутствуют в левых частях.

1-4. Аналогично отношению Club.
5.

- Отношение **Room**

Атрибуты:

- RoomId
- Area
- RoomName
- ClubId

Функциональные зависимости:

RoomId -> Area, RoomName, ClubId

Ключ {RoomId}. Единственность аналогично отношению Client.

1-6. Аналогично отношению Club.

- Отношение **GroupWorkout**

Атрибуты:

- Day
- Starts
- Ends
- RoomId
- TrainerId
- Workout_name

Функциональные зависимости:

Day, Starts, RoomId -> Ends, TrainerId, Workout_name

Day, Starts, TrainerId -> Ends, RoomId, Workout_name

Ключи {Day, Starts, RoomId}, {Day, Starts, TrainerId}. Единственность аналогично отношению Training.

1-6. Аналогично отношению Training

- Отношение **Equipment**

Атрибуты:

- EquipmentId
- EquipmentName
- ClubId

Функциональные зависимости:

EquipmentId -> EquipmentName, ClubId

Ключ {EquipmentId}. Единственность аналогично отношению Client.

1-6. Аналогично отношению Club.

- Отношение **Pool**

Атрибуты:

- PoolId
- LaneCount
- LaneLength
- ClubId

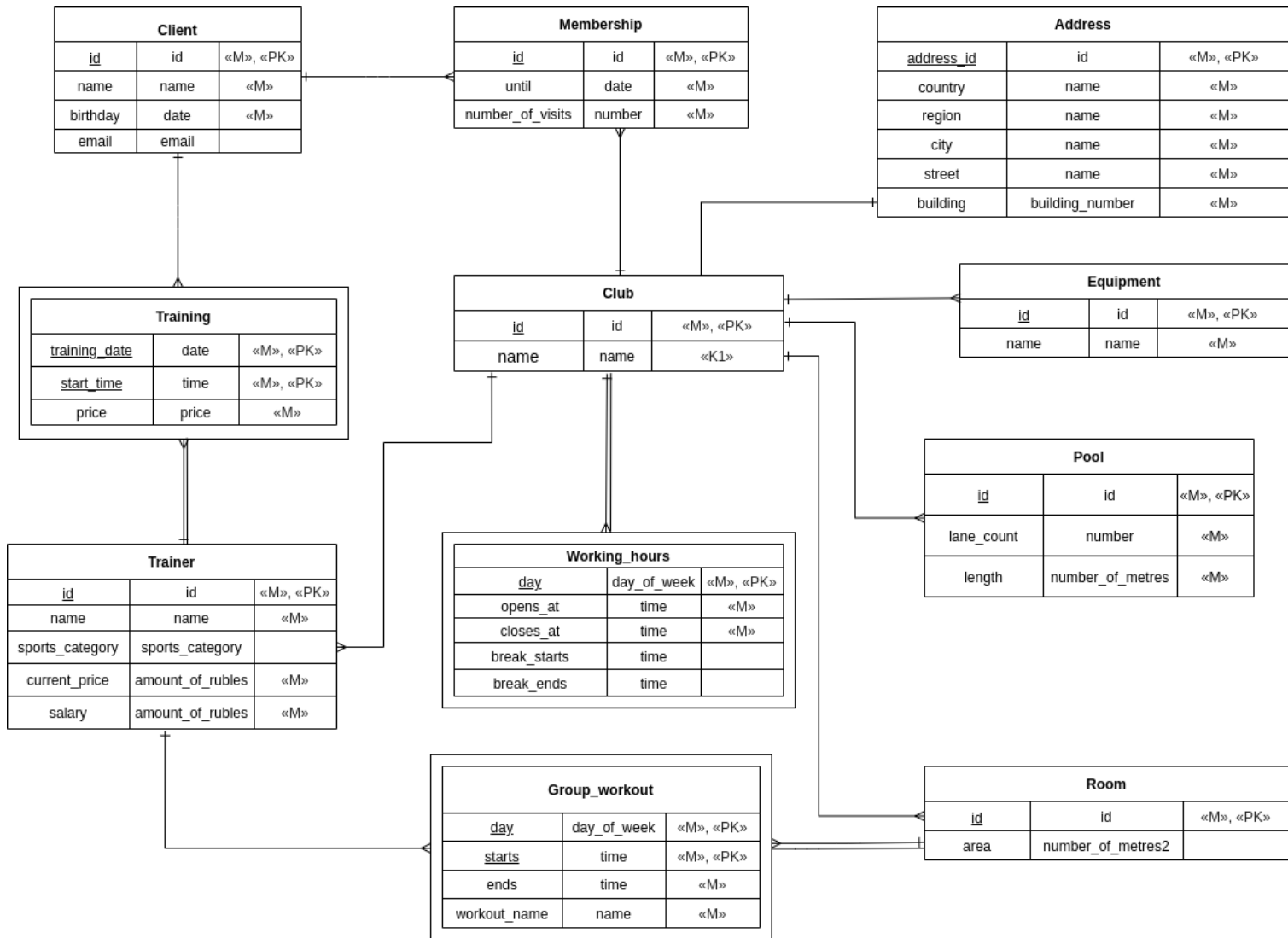
Функциональные зависимости:

PoolId -> LaneCount, LaneLength, ClubId

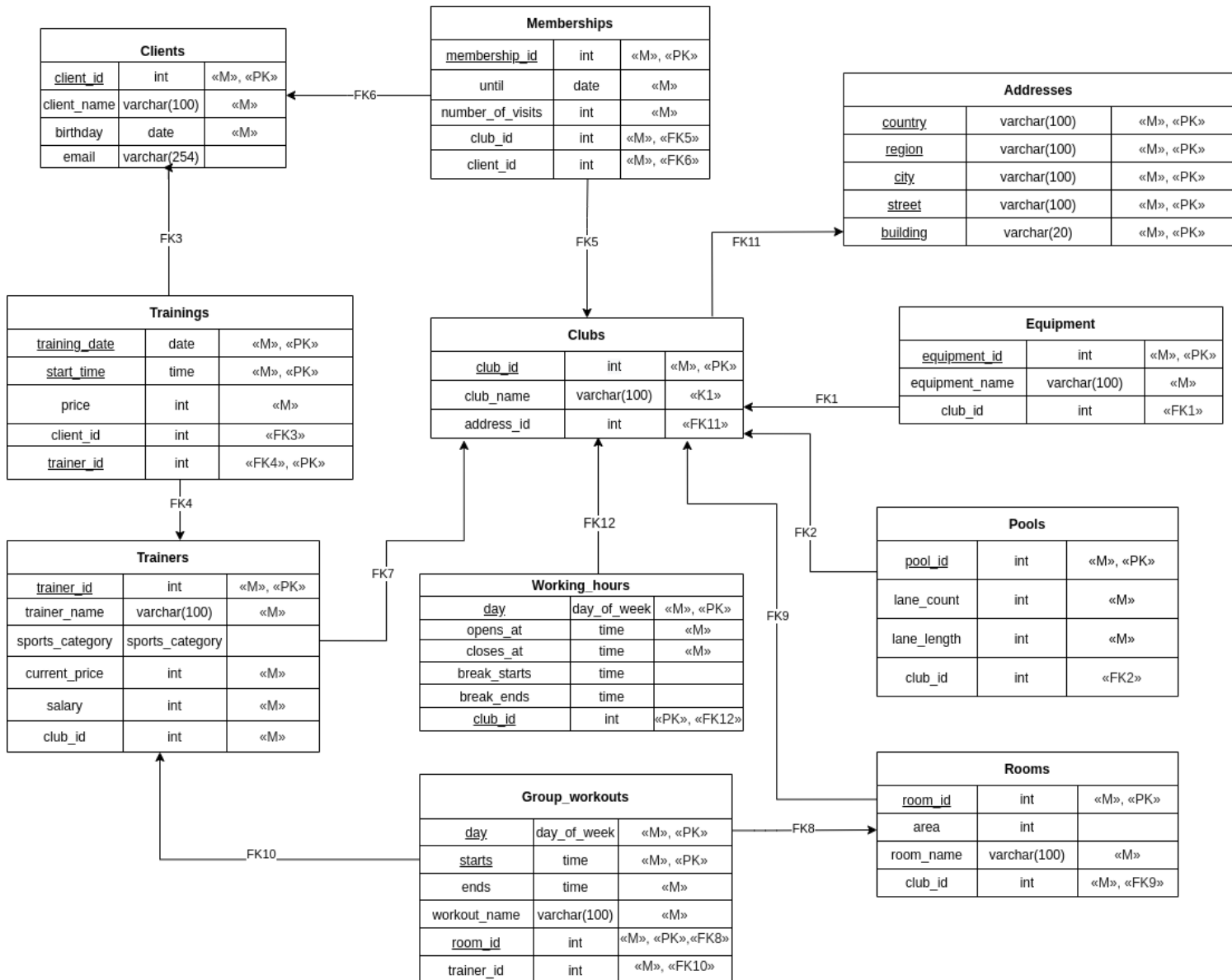
Ключ {PoolId}. Единственность аналогично отношению Client.

1-6. Аналогично отношению Club.

Модель сущность-связь



Физическая модель



При построении физической модели использовалось следующее отображение доменов в типы:

Домен	Тип
id	int
day_of_week	day_of_week
name	varchar(100)
time	time
date	date

building_number	varchar(20)
email	varchar(254)
price	int
number	int
number_of_metres	int
number_of_metres2	int
sports_category	varchar(100)
amount_of_rubles	int

Определения таблиц

Для реализации проекта использовалась СУБД PostgreSQL 11.7.
Определения таблиц и их индексов приведено в файле `ddl.sql`.

Тестовые данные

Скрипт для добавления тестовых данных приведен в файле `data.sql`.

Запросы на получение данных

В рамках проекта были реализованы следующие запросы:

- `SelectActive1` — клиенты с активным абонементом в первом клубе
- `SelectMasters2` — тренера со званием “мастер спорта” во втором клубе
- `Shedule1` — расписание групповых занятий в первом клубе
- `ClubsWithPools` — клубы с бассейном и информацией о бассейнах
- `Trainings1` — данные о тренировках в первом клубе
- `WorkingHours1` — часы работы первого клуба
- `TrainingShedule1` — расписание тренировок тренера 1 2020-03-03
- `Equipment2` — снаряжение второго клуба
- `Characteristics` — список клубов с их адресами и характеристиками - количество снаряжения, есть ли бассейн

Название запроса может быть как на русском, так и на английском языке.
В файле указывается в комментарии перед запросом.

Запросы на получение данных и вспомогательные представления приведены в файле `selects.sql`.

Запросы на изменение данных

В рамках проекта были реализованы следующие запросы:

- updatePrices1 — увеличение стоимости тренировок в клубе на 10%.
- addClient — добавление клиента
- addTrainer — добавление тренера
- addMembership — добавление абонемента
- addTraining - добавление тренировки
- addGroupWorkout - добавление группового занятия

Запросы на изменение данных, хранимые процедуры и триггеры приведены в файле `updates.sql`.