

Даулбай Дамир
Группа: ТИС-23Р

Предметная область: Фитнес Клуб

Атрибут	Домен	Обяз
		.
client_id (PK)	int	Да
full_name	varchar(100)	Да
phone	varchar(20)	Да
email	varchar(100)	Нет
birth_date	date	Нет

Абонемент:

Атрибут	Домен	Обяз
		.
membership_id (PK)	int	Да
name	varchar(50)	Да
price	decimal(10,2)	Да
duration_days	int	Да

Покупка абониента:

Атрибут	Доме н	Обяз
		.
client_membership_id (PK)	int	Да
client_id (FK)	int	Да
membership_id (FK)	int	Да
start_date	date	Да
end_date	date	Да

Тренер:

Атрибут	Домен	Обяз
		.
trainer_id (PK)	int	Да
full_name	varchar(100)	Да
speciality	varchar(100)	Да
phone	varchar(20)	Да

Тренировка:

Атрибут	Домен	Обяз
		.
session_id (PK)	int	Да
trainer_id (FK)	int	Да
session_date	datetime	Да
room	varchar(20)	Да
type	enum('group','personal')	Да

Запись клиента на тренировку:

Атрибут	Домен	Обяз
		.
client_session_id (PK)	int	Да
client_id (FK)	int	Да
session_id (FK)	int	Да
signup_time	datetime	Да

2. Связи между сущностями

1. Client — ClientMembership

- Тип: 1:N
- Обязательность: не обязательно иметь покупку
- Описание: клиент может покупать много абонементов

2. Membership — ClientMembership

- Тип: 1:N
- Описание: один абонемент покупают разные клиенты

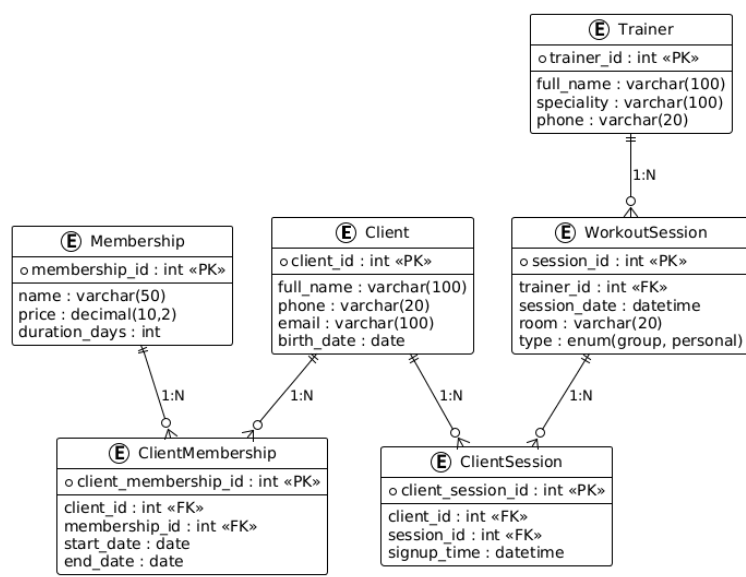
3. Trainer — WorkoutSession

- Тип: 1:N
- Описание: тренер проводит много тренировок

4. Client — ClientSession — WorkoutSession

- Тип: M:N
- Описание: клиент может записываться на разные тренировки, а тренировка имеет нескольких клиентов

3 - ЗАДАНИЕ: ER-ДИАГРАММА



4 - ЗАДАНИЕ

Таблица клиент:

Client (client_id PK, full_name, phone, ...)

Таблица тренировок:

WorkoutSession (session_id PK, trainer_id FK, date, room, type)

Таблица связка:

ClientSession (
 client_session_id PK,
 client_id FK,
 session_id FK,
 signup_time
)

Пересечение ключей:

- client_id — ключ Client
- session_id — ключ WorkoutSession

Client ⋈ ClientSession ⋈ WorkoutSession

Этот JOIN восстановит все данные о том, какой клиент на какую тренировку записан.

Задачи с книги:

id	first_name	last_name
	Марсия	Брэди
1	Джен	Брэли
	Бобби	Брэди
	Синди	Брэди
99	Питер	Брэди

-- 0. Удаляем таблицу, если она уже существует
DROP TABLE IF EXISTS people;

-- 1. Создаем таблицу
**CREATE TABLE people (
 id INT AUTO_INCREMENT PRIMARY KEY,
 first_name VARCHAR(50) NOT NULL,
 last_name VARCHAR(50) NOT NULL
);**

-- 2. Попытки вставки данных

-- Команда №1 (успешно)

```
INSERT INTO people (id, first_name, last_name)
VALUES (NULL, 'Марсия', 'Брэди');
```

```
-- Команда №2 (ОШИБКА: id = 1 уже используется)
-- (оставляем для задания, выполнять не обязательно)
INSERT INTO people (id, first_name, last_name)
VALUES (1, 'Джен', 'Брэли');
```

```
-- Команда №3 (ОШИБКА: " нельзя вставить в INT)
-- (оставляем для задания)
INSERT INTO people
VALUES ("", 'Бобби', 'Брэди');
```

```
-- Команда №4 (успешно: id = 2)
INSERT INTO people (first_name, last_name)
VALUES ('Синди', 'Брэди');
```

```
-- Команда №5 (успешно: id = 99)
INSERT INTO people (id, first_name, last_name)
VALUES (99, 'Питер', 'Брэди');
```

```
-- 3. Проверка содержимого таблицы
SELECT * FROM people;
```

Упражнение:
Таблица Грега:

таблица, где Грег записывает свои заказы пиццы, указывая *номер заказа* и *адрес доставки*.

Атомарна?

Да, атомарна.

Почему:

- В каждой строке — один заказ.
- Каждый столбец содержит одно значение: один номер, один адрес.
- Нет множественных адресов или списков внутри ячеек.

2. Таблица с оценками пончиков (стр. 112)

Эта таблица обычно содержит:

- название пончика
- оценку от разных людей

Пример:

Donut | Ratings

В книге она *до нормализации* содержит несколько оценок в одной ячейке, типа:

"5, 4, 3"

Нет, НЕ атомарна.

Почему:

- В ячейке "оценки" хранится сразу несколько значений.
- Это нарушает 1NF.
- Таблица преобразуется в атомарную, когда оценки выносятся в отдельные строки: donut | person | rating

3. Таблица с клоунами (стр. 155)

Это таблица, где у каждого клоуна может быть несколько навыков или атрибутов.

Обычно выглядит примерно так:

`clown | act`

Но в книге колонка *act* содержит список актов, например:
"juggling, balloon animals, pratfall"

Атомарна?

Не атомарна.

Почему:

- Поле "act" содержит не одно значение, а список.
- Это нарушает принцип 1NF.
- Чтобы сделать атомарной, надо нормализовать: clown | act

4. Таблица с описаниями напитков (стр. 93)

Обычно таблица описывает напиток, ингредиенты и цену.

Но до *нормализации* ингредиенты записаны как:

<code>coffee drinks</code>	<code>ingredients</code>
<code>Mocha</code>	<code>espresso, chocolate, milk</code>
<code>Latte</code>	<code>espresso, milk</code>

Атомарна?

Нет.

Почему:

- Колонка *ingredients* содержит сразу несколько ингредиентов.
- Нужна отдельная таблица Drink–Ingredient.

5. Информация о рыбах (стр. 19)

На ранних страницах дают таблицу вида:

<code>fish_name</code>	<code>color</code>	<code>location</code>
------------------------	--------------------	-----------------------

Возможно, там встречается строка типа:

<code>Salmon</code>	<code>pink</code>	<code>Alaska, Canada</code>
---------------------	-------------------	-----------------------------

Если *location* перечисляет несколько географий → таблица не атомарна.

Атомарна?

Скорее всего — нет.

Почему:


- Если столбец *location* содержит **несколько мест**, это нарушение атомарности.
- Каждая рыба может жить в нескольких регионах → нужно вынести в отдельную таблицу FishLocations:

Итоговое наше краткое сравнение:

Таблица	Атомарна?	Причина
Таблица Грега	Да	одно значение в каждой ячейке
Оценки пончиков	Нет	несколько оценок в одной ячейке
Таблица клоунов	Нет	список актов в одной ячейке
Описания напитков	Нет	список ингредиентов
Информация о рыбах	Скорее нет	несколько локаций в одной ячейке


Задание: Таблица Клоун

Возьми в руку карандаш



Давайте сделаем таблицу более атомарной. Предположим, поиск должен вестись по столбцам *appearance* и *activities*, а также по столбцам *last_seen*. Запишите более правильную структуру столбцов.

Она есть на с. 223



Правильная атомарная структура:

clown		
id	name	last_seen_location_id
<ul style="list-style-type: none">• <code>name</code> — имя клоуна• <code>last_seen_location_id</code> — ссылка на таблицу мест (нормализовано)		

2. Места, где видели клоунов

location

| id | place_name |

Примеры:

"Дом престарелых Черри Хилл", "Цирк BG", "Болмарт" и т. д.

3. Таблица appearance (внешность)

Внешность НЕ должна храниться строкой типа:

"Ж, рыжие волосы, зеленый костюм, огромные ботинки"

Нужно разбить на отдельные признаки.

Лучший способ — таблица "appearance_properties":

appearance_property

| id | property_name |

Примеры свойств:

- "женщина"
- "рыжие волосы"
- "зеленый костюм"
- "огромные ботинки"

- "большой нос"
- "сигара"
- "маленькая шляпа"

И связующая:

clown_appearance

| clown_id | property_id |

Каждый элемент appearance становится отдельной строкой.

4. Таблица activities (виды деятельности)

Тоже не должно быть:

"танцы, шарики"

Создаём:

activity

| id | activity_name |

Примеры:

- шарики
- машинки
- мим
- рожок
- зонтик
- танцы
- скрипка

- эквилибристика
- разъезжает на машинке

Связующая:

clown_activity

| clown_id | activity_id |

Финал: Атомарная таблица

clown (id, name, last_seen_location_id)

location (id, place_name)

appearance_property (id, property_name)

clown_appearance (clown_id, property_id)

activity (id, activity_name)

clown_activity (clown_id, activity_id)

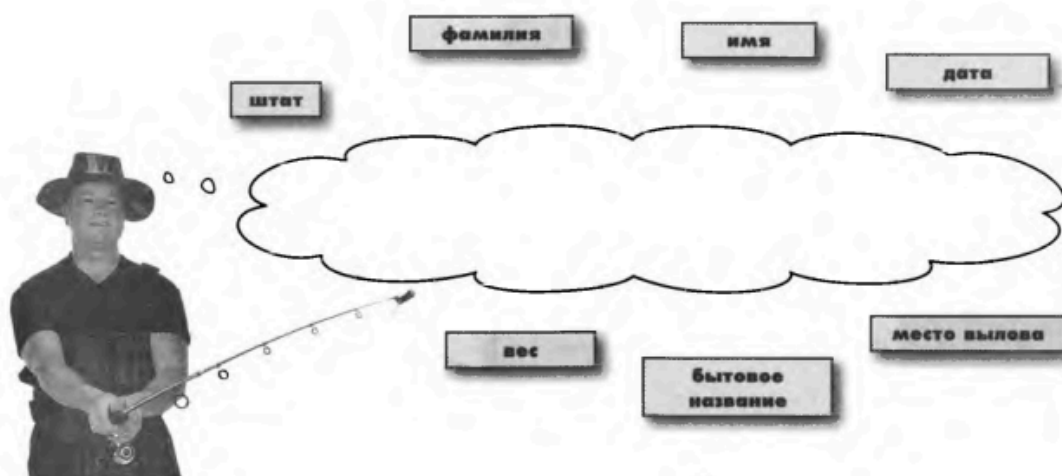
Задание:



Сможете ли вы определить столбцы таблицы по тем словам, которыми ихтиолог Марк описывает выборку данных из таблицы? Запишите имена столбцов в прямоугольниках.



Теперь ваша очередь. Напишите аналогичную фразу для Джека, автора статей по рыбной ловле, который использует таблицу для получения подробной информации для своих статей. Затем проведите стрелки от каждого столбца к его упоминанию в описании.



Часть 1. Определяем столбцы таблицы по словам Марка

В облачке Марк говорит:

«Я провожу поиск по бытовому или научному названию рыбы и хочу узнать вес и место вылова.»

По смыслу он упоминает такие столбцы таблицы:

- Бытовое название
- Научное название
- Вес
- Место вылова

Именно эти названия должны быть вписаны в прямоугольники вокруг его фразы.

Часть 2. Пишем аналогичную фразу для Джека

Джек — автор статей о рыбной ловле. Он использует таблицу, чтобы получать подробные сведения для статей.

Под таблицей изображены доступные столбцы:

- Фамилия
- Имя
- Дата
- Штат
- Вес
- Бытовое название
- Место вылова

Пример подходящей фразы для Джека:

«Я подбираю интересные случаи улова по дате и штату, смотрю, кто именно поймал рыбу, и использую её бытовое название и вес для описания в статье.»

Отметки для стрелок (какие слова относятся к каким столбцам)

В фразе Джека упоминаются:

- Дата → «по дате»
- Штат → «по ... штату»
- Имя / Фамилия → «кто именно поймал рыбу»
(можно направить стрелки к обоим столбцам: «имя» и «фамилия»)
- Бытовое название → «её бытовое название»
- Вес → «и вес»

Столбец «место вылова» в примере можно использовать, но по смыслу фразы он необязателен. Если хотите включить — можно расширить фразу.