

Основы проектирования баз данных

Нормализация данных

contact_id: 341
last_name: Мур
first_name: Найджел
phone: 5552311111
email: nigelmoore@ranchersrule.com
gender: M
birthday: 1975-28-08
profession: Фермер
city: Остин
state: TX
status: Не женат
interests: животные, лошади, кино
seeking: Незамужняя женщина

Найджел



Задание

```
SELECT * FROM my_contacts
WHERE gender = 'Ж'
AND status = 'Не замужем'
AND state='TX'
AND seeking LIKE '%Мужчина%'
AND birthday > '1970-28-08'
AND birthday < '1980-28-08'
AND interests LIKE .....
AND .....
AND .....
```

```
contact_id: 1854
last_name: Фиоре
first_name: Карла
phone: 5557894855
email: cfioire@fioreanimalclinic.com
gender: Ж
birthday: 1974-01-07 ← подходит
profession: Ветеринар ← хорошая профессия
city: Раунд-Рок
state: TX ← и даже живет неподалеку
status: Не замужем
interests: лошади, кино, животные, детективы,
туризм
seeking: single M ← интересы совпадают!
```

Карла
и Триггер

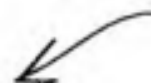


Возьми в руку карандаш

Используйте функцию SUBSTRING_INDEX для выделения первого увлечения из столбца interests.

```
SELECT * FROM my_contacts
WHERE gender = 'Ж'
AND status = 'Не замужем'
AND state='MA'
AND seeking LIKE '%Мужчина%'
AND birthday > '1950-28-08'
AND birthday < '1960-28-08'
AND SUBSTRING_INDEX(interests,',',1) = 'животные';
```

В запросах будут отображаться только женщины, у которых в списке увлечений на первом месте стоят 'животные'.



Задание

contact_id: 459
last_name: Фергюсон
first_name: Алексис
phone: 5550983476
email: alexangel@yahoo.com

gender: Ж

birthday: 1956-19-09 ← подходит

profession: Художник

city: Пфлюгервиль

state: МА ← живет близко

status: Не замужем

interests: животные ← подходящие увлечения

seeking: Мужчина



Реджи →

Напишите для Реджи запрос, не использующий столбец interests.

Грег пишет запрос, который должен вернуть Реджи подходящую пару. Он начинает с простых столбцов — gender, status, state, seeking и birthday — и только потом берется за столбцы interest.

Запишите его запрос.

```
SELECT * FROM my_contacts
```

```
WHERE gender = 'Ж'
AND status = 'Не замужем'
AND state='MA'
AND seeking LIKE '%Мужчина%'
AND birthday > '1950-20-03'
AND birthday < '1960-20-03'
AND
```

```
(
interest1 = 'животные'
OR interest2 = 'животные'
OR interest3 = 'животные'
OR interest4 = 'животные'
)
```

```
AND
```

```
(
interest1 = 'коллекционные карточки'
OR interest2 = 'коллекционные карточки'
OR interest3 = 'коллекционные карточки'
OR interest4 = 'коллекционные карточки'
)
```

```
AND
```

```
(
interest1 = 'геопоиск'
OR interest2 = 'геопоиск'
OR interest3 = 'геопоиск'
OR interest4 = 'геопоиск'
);
```

Реджи ищет незамужнюю женщину, родившуюся в заданный период времени, которая живет в Массачусетсе и хочет встречаться с незамужним мужчиной.

Чтобы найти совпадения с увлечениями Реджи, Грег вынужден проверить все четыре новых столбца «interest», потому что в каждом из них может найтись совпадение.

Столбец «interest4» у Реджи содержит NULL, поэтому проверяются только три увлечения вместо четырех.

должен вернуть Реджи
: простых столбцов —
irthday — И ТОЛЬКО

aisaflake.com

ые карточки

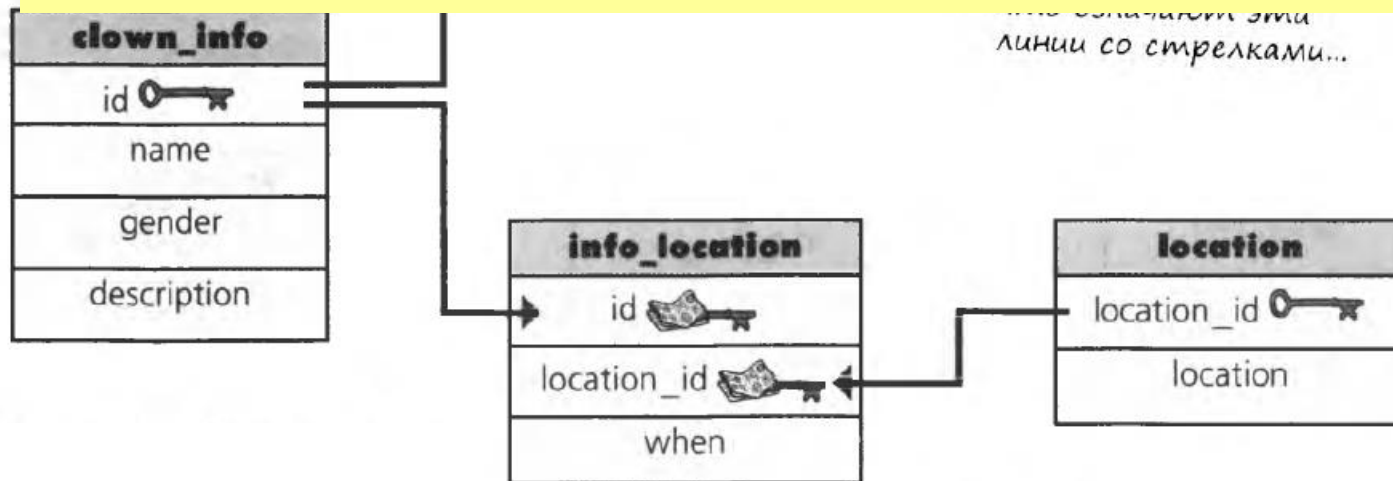
**Неатомарные столбцы из существующей
таблицы следует переместить в новые
таблицы.**

Таблица о Клоунах

clown_tracking

clown_info	name	last seen	activities
Элси	Дом престарелых Черри Хилл	Ж, рыжие волосы, зеленый костюм, огромные ботинки	шарики, машинки
Пиклз	Вечеринка Джека Грина	оранжевые волосы, синий костюм, огромные ботинки	мим
Снагглз	Болмарт	желтая рубашка, красные штаны	рождок, зонтик

Описание данных (столбцов и таблиц) вашей базы данных, включая все взаимосвязанные объекты и связи между ними, называется **СХЕМОЙ**.



Структура таблицы

2

Добавляем столбцы, по которым можно будет узнать, какие увлечения принадлежат тому или иному человеку из таблицы `my_contacts`.

Мы вынесли увлечения из таблицы `my_contacts`, но как определить, кому какие увлечения принадлежат. Необходимо использовать информацию из таблицы `my_contacts` и разместить ее в таблице `interests` так, чтобы эти две таблицы были связаны между собой.

Например, для этого можно включить столбцы `first_name` и `last_name` в таблицу `interests`.

my_contacts
contact_id
last_name
first_name
phone
email
gender
birthday
profession
city
state
status
seeking

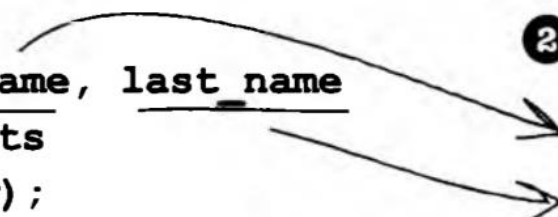
interests
int_id
interest
first_name
last_name

По этим двум столбцам можно узнать, кто какими увлечениями обладает.

Если эти значения совпадают, значит, человек обладает данным увлечением. Таблица `interests` содержит несколько записей с совпадением этих значений, что позволяет связать с одним человеком сразу несколько увлечений.

1 **SELECT first_name, last_name**
FROM my_contacts
WHERE (условия);

2 **SELECT interest FROM interests**
WHERE first_name = 'Имя'
AND last_name = 'Фамилия';



Вам кажется, что эта запись неэффективна?
И правильно. Она всего лишь показывает, как использовать данные одной таблицы для извлечения данных из другой таблицы. (Вскоре мы покажем, как сделать то же самое более эффективно.)

Задание

Какие еще таблицы стоит добавить в базу данных Vasya_List для хранения информации о нескольких увлечениях?

Не старайтесь нарисовать аккуратную схему; сейчас время собирать идеи. Одна идея уже изображена на рисунке, но у нее есть недостаток.



Таблицы связываются по совпадениям в столбцах «first_name» и «last_name». Совпадения определяют, кто какими увлечениями обладает.

Внешний ключ

my_contacts	
contact_id	🔑
last_name	
first_name	
phone	
email	
gender	
birthday	
profession	
city	
state	
status	
seeking	

Чтобы новая таблица соответствовала правилам первой нормальной формы, каждой записи назначается уникальное значение первичного ключа.

interests	
int_id	🔑
interest	
contact_id	🔗

ВНЕШНИЙ КЛЮЧ
определяет, какие увлечения принадлежат тому или иному человеку из таблицы my_contacts.

Внешний ключ

Имя внешнего ключа может отличаться от имени первичного ключа, с которым он связывается.

Внешний ключ может использоваться для установления соответствия между записями двух таблиц.

Внешний ключ может содержать значения NULL, хотя в первичном ключе они запрещены.

Значения внешнего ключа не обязаны быть уникальными — более того, чаще они уникальными не являются.

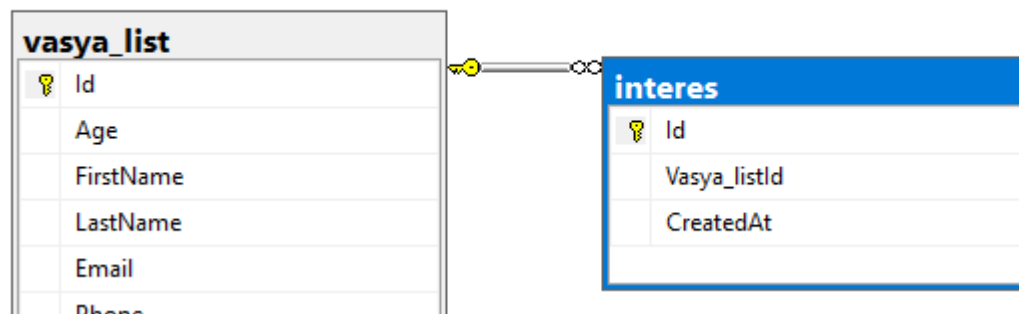
Ограничения внешнего ключа

Создание ВНЕШНЕГО КЛЮЧА как ограничения таблицы дает определенные преимущества.

При попытке нарушения правила вы получите сообщение об ошибке; таким образом предотвращаются случайные нарушения связей между таблицами.

Создание таблицы с внешним ключом

```
CREATE TABLE interests (  
  intid INT NOT NULL PRIMARY KEY IDENTITY,  
  interest VARCHAR(50) NOT NULL,  
  contactid INT NOT NULL,  
  FOREIGN KEY (contactid) REFERENCES vasya_list(Id)  
)
```



Конец фильма
