

Москва, 2022

## Цель

Сформировать понимание особенностей создания аналитических баз данных и умение их настраивать и поддерживать

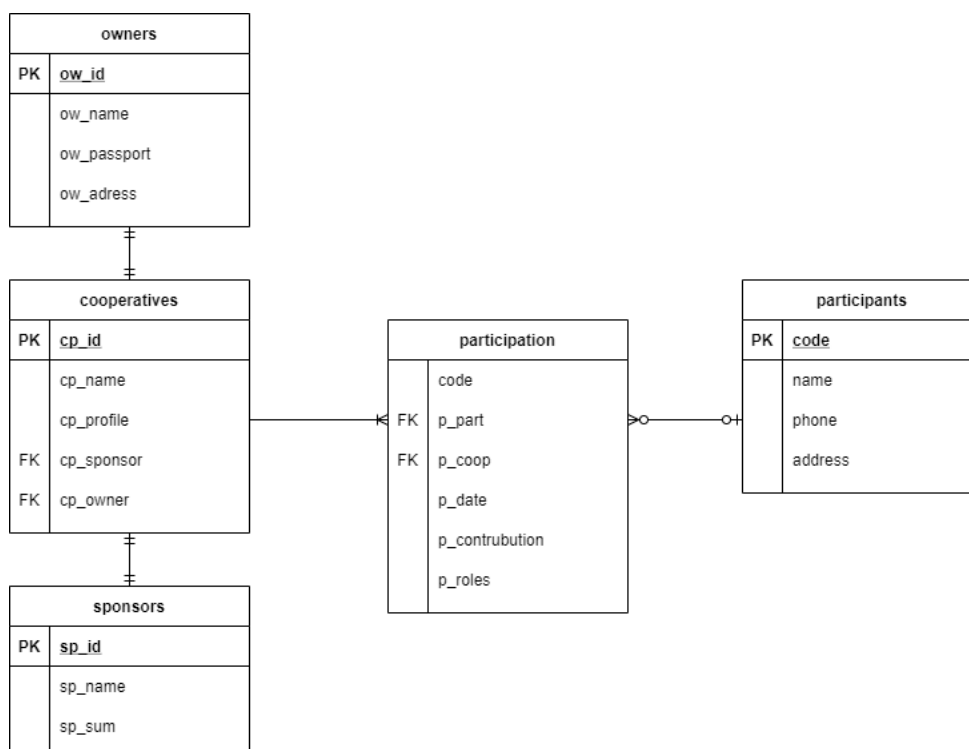
## Задачи

- Получить теоретические знания денормализации
- Узнать о основных методах денормализации
- Ознакомиться с массивами
- Научится (изменять\добавлять\удалять) данные в массиве с помощью встроенных операций
- Более подробно узнать о типе данных JSON
- Узнать о ролях и пользователях
- Научиться пользоваться командами для того, чтобы (определять\отзывать) доступ к данным, GRANT и REVOKE. • Научится упрощать запросы к БД с помощью представлений.

## Практическая часть

### Часть 1

База данных была перепроектирована, были добавлены столбцы формата массив и jsonb. Итоговая ER-диаграмма выглядит следующим образом:



Чтобы сгенерировать данные для таблиц мы воспользовались генератором данных на python, предварительно подключив библиотеки mimesis, faker и pandas.

Мы написали функцию для каждой из сущности, которую меняли - каждая функция генерирует заданное количество строк и возвращает результат в виде структуры данных - словарь, а потом преобразовали его в таблицу pandas. У библиотеки pandas есть метод to\_csv благодаря которому мы преобразовали наши данные в csv формат.

```
bd3.py > participants
1  from mimesis import Person
2  from mimesis import Text
3  from faker import Faker
4  import pandas as pd
5  import random
6  import json
7
8
9  person = Person('ru')
10 rand_text = Text()
11 faker = Faker()
12
13
14 def participants(start, number_of_rows):
15     return [{
16         'code': x + 1,
17         'name': person.first_name() + ' ' + person.last_name(),
18         'phone': person.telephone(),
19         'address': json.dumps({
20             'country': faker.country(),
21             'city': faker.city(),
22             'street': faker.street_name(),
23             'postcode': faker.postcode()
24         })
25     } for x in range(start, start + number_of_rows, 1)]
26
27
28
29 pd.DataFrame(participants(100000001, 50000)).to_csv('participants.csv', index=False, mode='a', header=False)
```

```

bd3.py > ...
1  from mimesis import Person
2  from mimesis import Text
3  from faker import Faker
4  import pandas as pd
5  import random
6  import json
7
8
9  person = Person('ru')
10 rand_text = Text()
11 faker = Faker()
12
13
14 roles = ['accountant', 'participant', 'executor', 'chief', 'manager', 'speaker']
15
16 def participation(start, number_of_rows):
17     return [{
18         'code': x + 1,
19         'p_part': random.randint(1, 100050000),
20         'p_coop': random.randint(1,10),
21         'p_date': faker.date(),
22         'p_contribution': random.randint(10000, 100000),
23         'p_roles': [random.choices(roles)[0] for i in range(random.randint(1,3))]
24     } for x in range(start, start + number_of_rows, 1)]
25
26
27
28 pd.DataFrame(participation(0, 1050000)).to_csv('participation.csv', index=False)

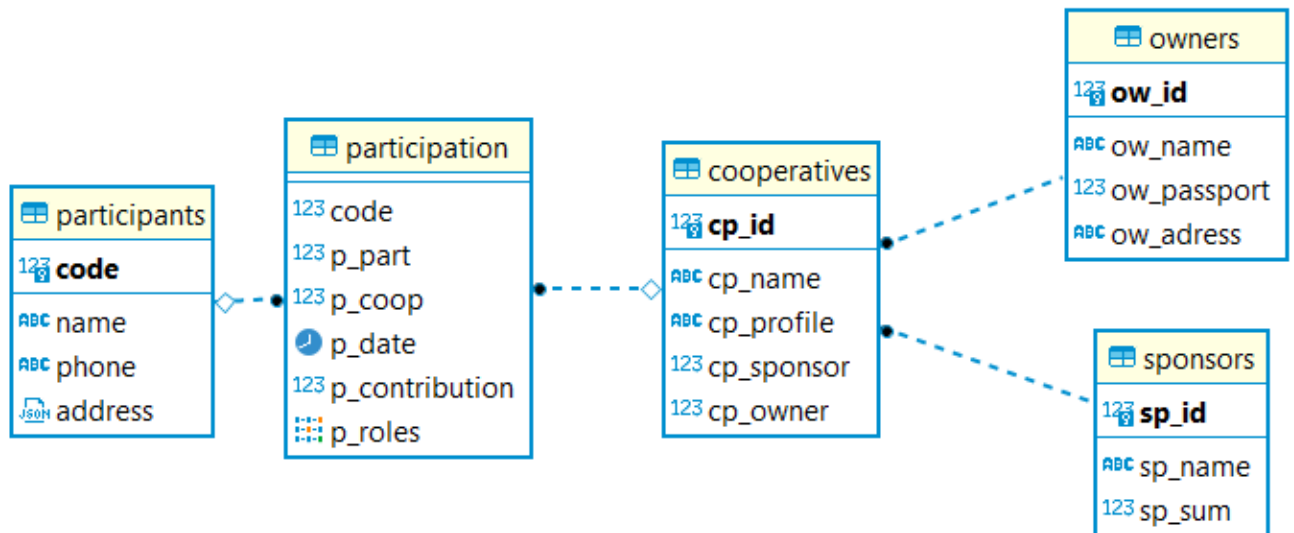
```

Лабораторная работа выполнялась с использованием среды DBeaver. Созданные таблицы формата CSV были импортированы в нашу базу данных с помощью встроенной функции.

Проверим, что наши таблицы participants и participation были импортированы:

<pre>select count(*) from participants p</pre>		<pre>select count(*) from participation p</pre>	
Результат 1 ×		Результат 1 ×	
<pre>select count(*) from participants</pre>		<pre>select count(*) from participation</pre>	
таблица	123 count	123 count	
1	100 050 000	1 050 000	

ER-диаграмма, созданная средой DBeaver:



## Часть 2

**Создать пользователя test и выдать ему доступ к базе данных.**

Создаем новую роль test:

```
CREATE ROLE test LOGIN;
```

Переключаемся на эту роль:

```
SET ROLE test;
```

Назначаем пользователя базы данных:

```
SET SESSION AUTHORIZATION test;
```

**Составить и выполнить скрипты присвоения новому пользователю прав доступа к таблицам, созданным в практическом задании 1.**

Права доступа SELECT, INSERT, UPDATE для таблицы **participants**:

```
GRANT SELECT, INSERT, UPDATE ON TABLE public.participants TO test;
```

Права доступа SELECT, UPDATE для таблицы **participation**:

```
GRANT SELECT, UPDATE ON TABLE public.participation TO test;
```

Права доступа SELECT для таблицы **cooperatives**:

```
GRANT SELECT ON TABLE public.cooperatives TO test;
```

**Составить SQL-скрипты для создания нескольких представлений:**

Общее кол-во участия в кооперативах из каждой страны:

```
create view number_of_participations_for_every_country as
select p2.address->'country' as country, count(p.code) from participation p,
participants p2
where p.p_part = p2.code
group by p2.address->'country'
order by 2 desc;
```

Кол-во участников в каждом кооперативе:

```
create view number_of_each_cooperative as
select c.cp_name, count(*) as number_of_participants from
cooperatives c, participation p
where p.p_coop = c.cp_id
group by c.cp_name
order by 2 desc;
```

Создать стандартную роль уровня базы данных, присвоить ей право доступа (UPDATE на некоторые столбцы) к представлению, созданному в практическом задании №3.3, назначить новому пользователю созданную роль.

```
create role standard LOGIN;
```

```
SET ROLE standard;
```

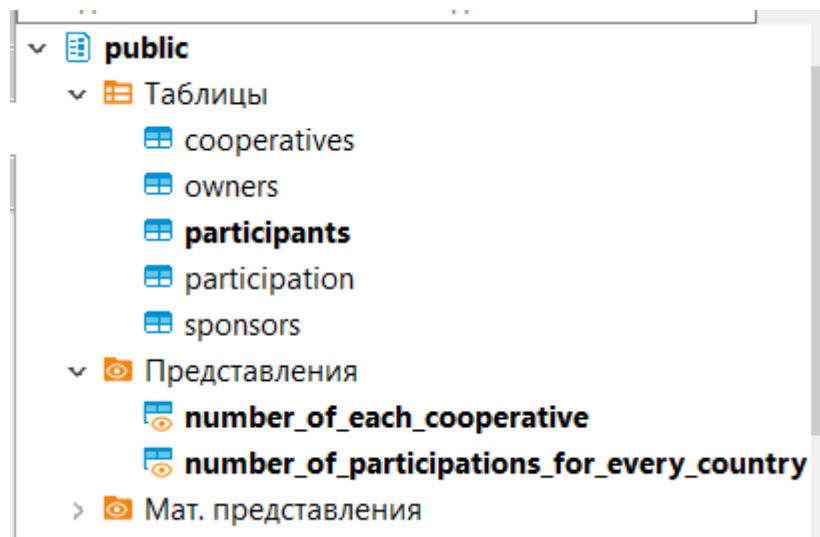
Присвоили роли право доступа на некоторые представления(добавили право доступа SELECT к одному из представлений)

```
GRANT UPDATE ON TABLE public.number_of_each_cooperative TO standard;
GRANT UPDATE ON TABLE public.number_of_participations_for_every_country TO standard;
GRANT SELECT ON TABLE public.number_of_each_cooperative TO standard;
GRANT UPDATE ON TABLE public.participants TO standard;
```

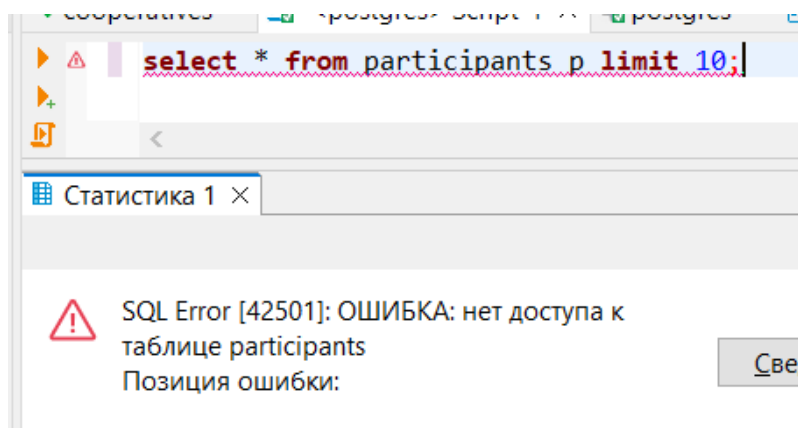
Авторизовались под новым пользователем

```
set session authorization standard;
```

Право доступа роли standard:



При попытке выполнить SELECT-запрос, права доступа на которую у пользователя отсутствуют возникает следующая ошибка:



Так как мы предоставили права доступа на SELECT к представлению number\_of\_each\_cooperative, то результат select-запросов будет вычислен

	cp_name	number_of_participants	
1	Empire Build	105 348	
2	Only Comfort	105 302	
3	All Clean	105 280	
4	OneProduct	105 125	
5	TTM	104 937	
6	Clerc	104 903	
7	PPP	104 880	
8	Eating God	104 871	
9	Gallery	104 846	
10	PenPen	104 508	

Попробуем изменить значения столбцов в таблицу participants

Name	Value
Updated Rows	1
Query	update participants set code = 7777, name = 'Амир Гарифуллин', phone = '+7-(979)-608-65-84' where code = 7777
Finish time	Sun Dec 11 18:26:23 MSK 2022

Так как пользователь с ролью standard обладает правами доступа на изменение, то изменение завершилось успешно. При попытке изменить другие таблицы возникает ошибка прав доступа.

## **Вывод**

В ходе выполнения данной лабораторной работы была доработана база данных из лабораторной работы №2, в частности были добавлены столбцы содержащие json-формат, формат типа массив, формат для полнотекстового поиска(tsvector). Были продемонстрированы навыки заполнения базы данных большим количеством осмысленных данных путем генерации с помощью сторонних библиотек языка Python. Были продемонстрированы навыки импортирования базы данных из формата CSV. Были изучены и продемонстрированы навыки работы с ролями и пользователями. Были изучены и добавлены в базу данных представления. Была проверена правильности контроля доступа при работе с базой данных в созданной роли.