



OTUS

ОНЛАЙН-ОБРАЗОВАНИЕ

Онлайн-образование



Меня хорошо видно && слышно?

Ставьте ☐, если все хорошо
Напишите в чат, если есть проблемы





PostgreSQL. Секционирование

Курочкин Константин
«Medindex»

Правила вебинара



Активно участвуем



Задаем вопрос в чат

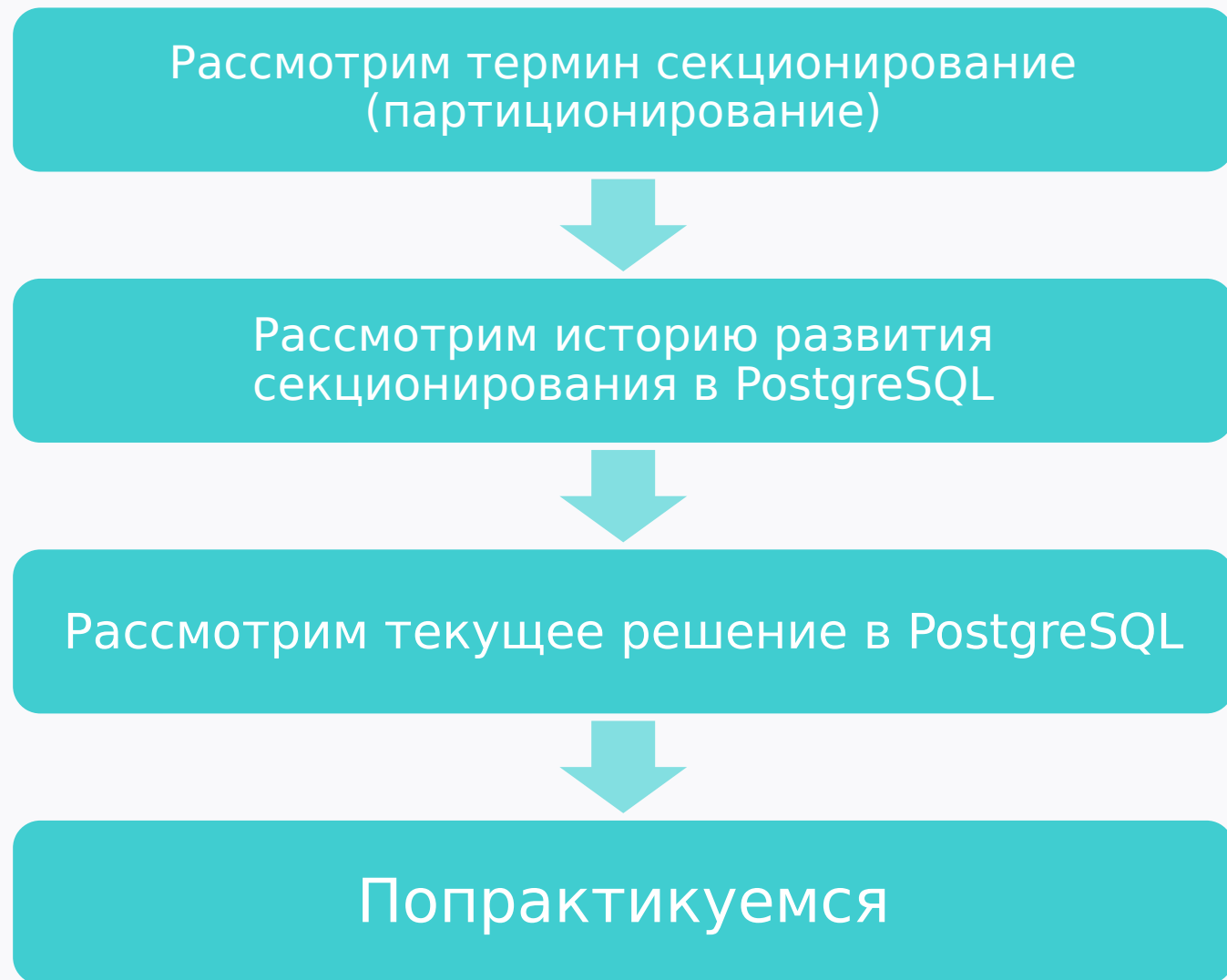


Off-topic обсуждаем в telegram



Вопросы вижу в чате, могу ответить не сразу

Маршрут вебинара



Цели вебинара | После занятия вы сможете

1

Создавать секционированные таблицы

2

Обслуживать секционированные таблицы

3

Увеличить производительность запросов

Смысл | Зачем вам это уметь

1

Для лучшего хранения большого количества данных

2

Для более быстрого доступа к данным

3

Для ускорения запросов

Слайд с заданием

Есть таблица на десять миллионов строк:

```
create table table1 (  
  id bigserial,  
  name text,  
  create_date date,  
  some_sum numeric  
);
```

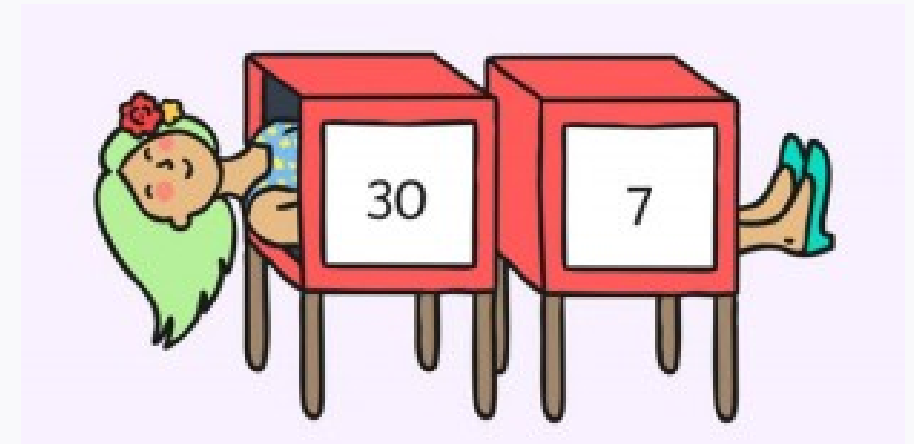
По таблице часто необходимо выполнять запросы вида:

```
select name, sum(some_sum)  
from table1  
where create_date = 'some_date'  
group by name;
```

Каким образом мы можем увеличить скорость выполнения подобных запросов?

Про секционирование

- Секционирование = партиционирование
- «Разделяй и властвуй»
- Таблицы в таблице
- У секций могут быть свои секции



- Общий синтаксис для секционирования таблицы:

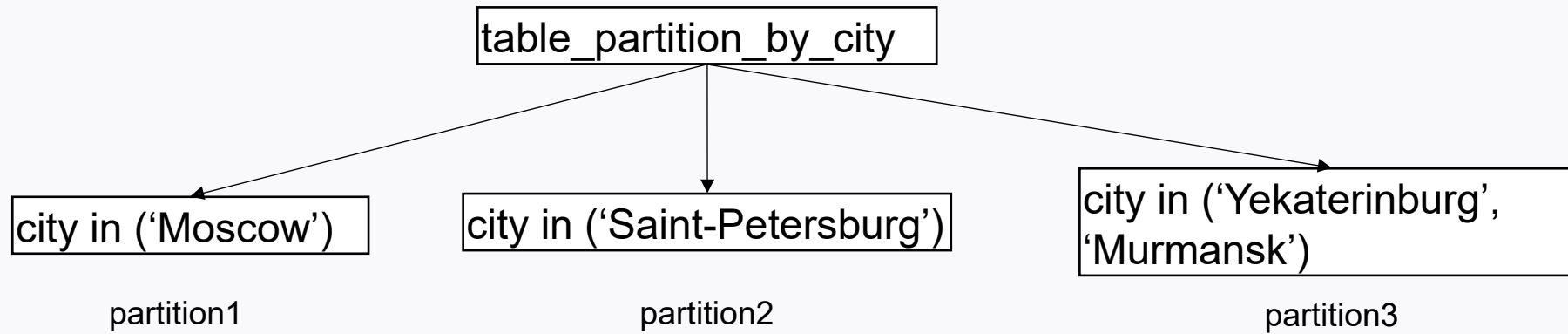
**PARTITION BY { RANGE | LIST | HASH } ({ имя_столбца | (выражение) } [COLLATE
правило_сортировки])**

CREATE TABLE section_name PARTITION OF table_name FOR VALUES FROM ('2006-02-01') TO ('2006-03-01');

Удаление партиции – DROP TABLE section_name

Виды секционирования

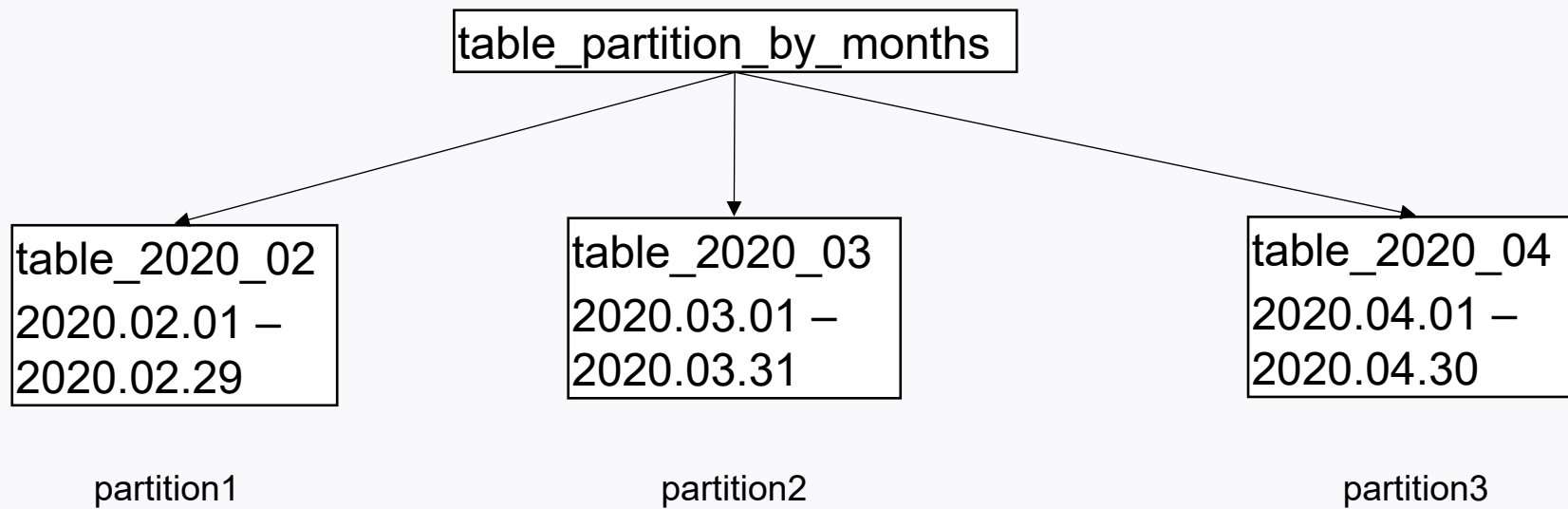
Секционирование по списку значений



```
CREATE TABLE city_moscow PARTITION OF city FOR VALUES IN  
( 'Moscow' );
```


Виды секционирования

Секционирование по диапазону значений



```
CREATE TABLE table_2020_02 PARTITION OF table FOR VALUES FROM ('2020-02-01') TO  
('2020-03-01')
```

Виды секционирования

Секционирование по хэшу

- Когда необходимо равномерное распределение
- Нет явного ключа, по которому можно разбить таблицу
- Для равномерного распределения необходимо уникальное или почти уникальное поле



```
CREATE TABLE table_p1 PARTITION OF table FOR VALUES WITH (MODULUS 5, REMAINDER 0);
```


Как мы можем секционировать таблицу

С помощью наследования (старый способ)

- Создаётся таблица и от неё наследуются секции с помощью команды:

```
CREATE TABLE table_s1 () INHERITS (table);
```

- В дочерних таблицах создаются ограничения, которые не пересекаются между секциями

```
CHECK (date between...)
```

- Создаётся триггер, который распределяет строки

```
CREATE TRIGGER trigger_name
```

```
BEFORE INSERT ON table_name
```

```
FOR EACH ROW EXECUTE PROCEDURE func_name();
```

Как мы можем секционировать таблицу

Декларативный способ

- Относительно новый способ (с версии 10)
- Проще переносить (т.к. синтаксис теперь похож на другие СУБД)
- Проще реализовать (без триггеров)
- Не забыть включить `enable_partition_pruning` для оптимизации
- Можно создавать подсекции

Декларативный способ

Можно создавать дефолтную секцию:

create table part_name partition of main_table default;

Можно использовать минимальное, максимальное значение для range partition

create table part_name partition of main_table for values from (MINVALUE) to (MAXVALUE);

Можно отсоединять секции:

alter table main_table detach partition part_name;

Можно добавлять секции:

alter table main_table attach partition part_name for values from ('2020-01-01') to ('2020-02-01');

Декларативный способ

Проблемки:

- В range partition не получится использовать в ключе секционирования null значения
- Не умеет создавать секции самостоятельно
- Не получится создать уникальное ограничение на часть

Как мы можем секционировать таблицу

Pg_pathman

- Доступно разбиение по хэшу и диапазонам
- Автоматическое создание секций при секционировании по диапазонам
- Можно соединять или разделять секции
- Не факт, что будут развивать в 14 версии (хотя тоже самое говорили про 13)

<https://postgrespro.ru/docs/postgrespro/13/pg-pathman>

https://github.com/postgrespro/pg_pathman

<https://www.programmersought.com/article/25792165890/>

<https://forms.gle/E9NL3R7v7fNsMywN7>

TECT



Слайд с заданием

Есть таблица на десять миллионов строк:

```
create table table1 (  
  id bigserial,  
  name text,  
  create_date date,  
  some_sum numeric  
);
```

По таблице часто необходимо выполнять запросы вида:

```
select name, sum(some_sum)  
from table1  
where create_date = 'some_date'  
group by name;
```

Каким образом мы можем увеличить скорость выполнения подобных запросов?



Практика!



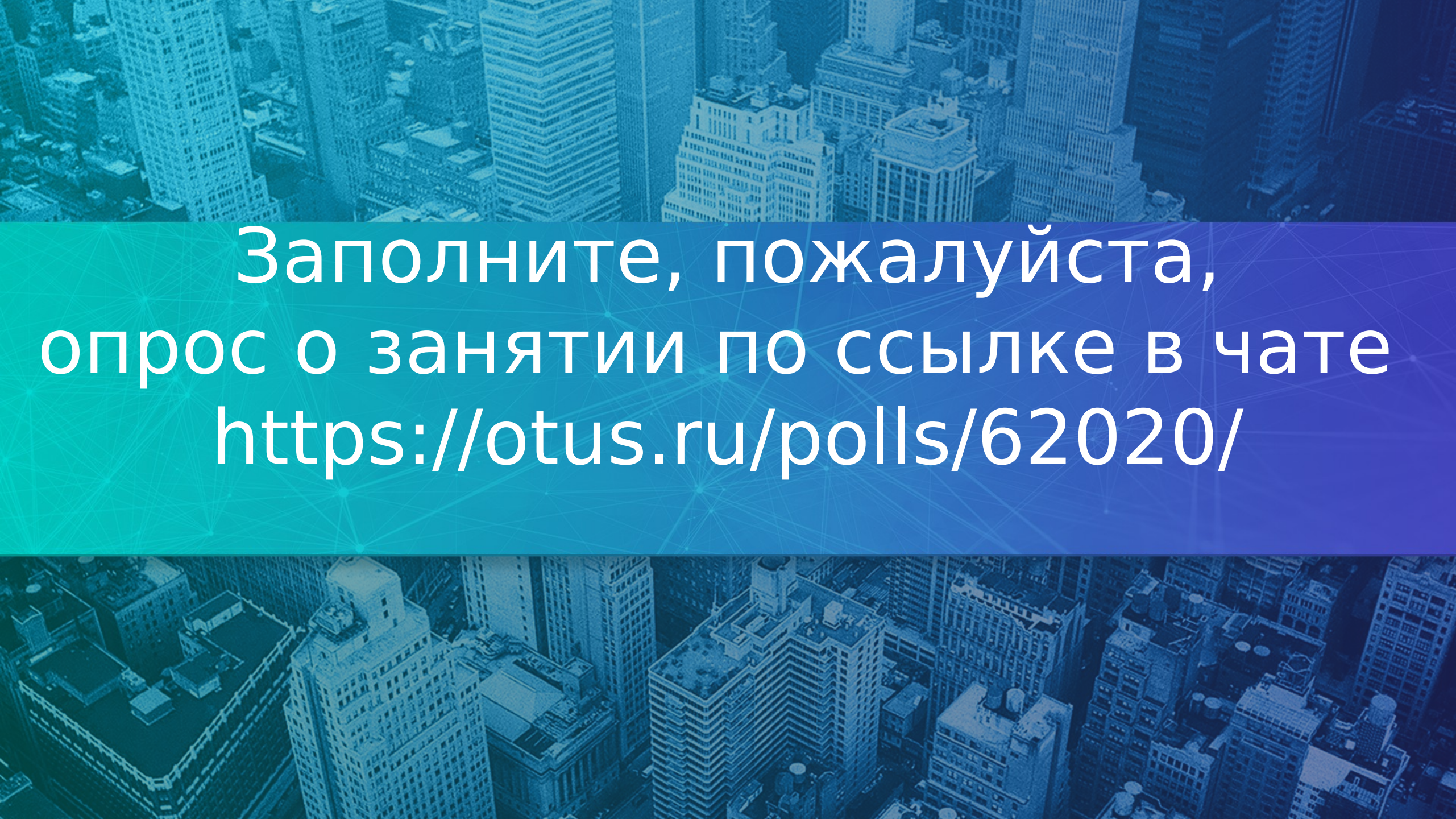
Рефлексия




Отметьте самый не раскрытый, по вашему мнению пункт



Есть ли у вас таблицы, которые стоит секционировать? Как вы будете это делать?



Заполните, пожалуйста,
опрос о занятии по ссылке в чате
<https://otus.ru/polls/62020/>



До новых встреч!
Приходите на следующие занятия

Куручкин Константин
«Medindex»