

Задание

Автоматизация и учет абонентов и их договоров с фирмами-провайдерами предоставляющих услуг связи Internet.

Абонент (ФИО или название предприятия, тип (частное лицо, школа, ВУЗ, предприятие,...), адрес, телефон, договор/договора (номер договора, информация о фирме-провайдере (название, тип собственности (государственный, частный, ЗАО, ОАО,...), адрес, телефон), тарифный план, дополнительные услуги (IPTV (кол-во каналов), тип IP-адреса (статический/динамический), IP-адрес, скидки (бизнес-время- 100% оплата, вечерний тариф- скидка 20%, ночной тариф- скидка 50%,...) ...), скорость интернета, стоимость подключения, дата подключения)). Примечание: один абонент может заключать несколько договоров.

Пример Структуры

```
{
  "name": "Petrov Ivan Petrovich",
  "type_abonent": "частное лицо",
  "phone": "+38071234567890",
  "address": "Artema 34",
  "dogovor": [
    {
      "firma": {
        "name": "Matrix",
        "phone": "+38071234567890",
        "address": "Vatutina 36",
        "type_sob": "частный"
      },
      "price": 294,
      "speed": 100,
      "tarif": "Медиа (100 Мбит + 120 IPTV)",
      "date_in": "2019-06-25 11:12:52.01897+03",
      "dop_usluga": {
        "name": "IPTV",
        "chanel": 120,
        "ip_address": "192.168.137.1",
        "type_ip_address": "static"
      },
      "number_dogovora": 32766
    },
    {
      "firma": {
        "name": "Trinity",
        "phone": "+38071239387890",
        "address": "Popovicha 102",
        "type_sob": "государственный"
      },
      "price": 500,
      "speed": 120,
      "tarif": "120Mb",
      "type_address": "dynamic",
      "number_dogovora": 301666
    }
  ]
}
```

Выборки из БД

Простые запросы:

```
SELECT data_jsonb ->> 'name',
       data_jsonb ->> 'phone',
       data_jsonb ->> 'address',
       data_jsonb ->> 'dogovor'
FROM json_table
WHERE data_jsonb ->> 'type_abonent' = 'предприятие';
```

name	phone	address	dogovor
Будур Захар Олегович	+38164802742762	ул. Ленина, 27	[{"data": "2020-10-16", "firma": {"name": "DIPT", "phone": "+38178117307161", "ad
Букалов Зураб Олесьевна	+38106771878894	ул. Ленина, 51	[{"data": "2020-08-28", "firma": {"name": "ORION", "phone": "+38131277632763", "
Бурзак Ирина Радионович	+38119338185172	ул. Куйбышева, д. 252	[{"data": "2020-06-24", "firma": {"name": null, "phone": "+38122954100448", "adre
Бурсов Карим Раимовна	+38145809468738	пр-т Ильича, д. 91	[{"data": "2020-12-14", "firma": {"name": "ORION", "phone": "+38106916910956", "
Бухало Лев Романович	+38100453076756	ул. Куйбышева, д. 252	[{"data": "2020-07-29", "firma": {"name": "Феникс", "phone": "+38157317472568", "
Буцик Лиля Руслановна	+38158933286627	ул. Мушкетовская, д. 24	[{"data": "2020-11-18", "firma": {"name": "ORION", "phone": "+38200050117753", "
Валуев Мишель Тимурович	+38186471412967	ул. Ленина, д.15	[{"data": "2020-07-09", "firma": {"name": "ORION", "phone": "+38162397800895", "
Вальковец Моисей Турсунович	+38197218889829	ул. Ленина, 27	[{"data": "2020-07-18", "firma": {"name": "DIPT", "phone": "+38184846919340", "ad

```
SELECT data_jsonb ->> 'name' as "name",
       data_jsonb #>> '{dogovor,1,firma,name}' as name_company,
       data_jsonb #>> '{dogovor,1,tarif}' as tarif,
       data_jsonb #>> '{dogovor,1,price}' as price
FROM json_table
WHERE data_jsonb->> 'type_abonent'='юридическое лицо' AND
       jsonb_array_length(data_jsonb->'dogovor')>1
```

name	name_company	tarif	price
Черняк Рене Артуровна	DIDAN	Минимальный	947
Шевела Владена Ивановна	DIPT	Минимальный	867
Чудновский Алла Витальевич	EAST.NET	КОМФОРТ	316
Шерекин Дамир Максимовна	Trinity	Базовый	247
Чирка Алевтина Васильевич	ДонАпекс	Социальный	136
Шевцов Владислав Игоревна	DIPT	Минимальный	867
Шелудяков Георгий Леонтьевич	DTK	Коттедж-100	646

```
SELECT (jsonb_array_elements(data_jsonb->'dogovor'))->'firma' ->> 'name' as "name",
       (jsonb_array_elements(data_jsonb->'dogovor'))->'number_dogovora' as num,
       (jsonb_array_elements(data_jsonb->'dogovor'))->'tarif' as tarif,
       (jsonb_array_elements(data_jsonb->'dogovor'))->'data' as "data",
       (jsonb_array_elements(data_jsonb->'dogovor'))->'dop_usluga' as dop
FROM json_table
WHERE data_jsonb ->> 'name'='Козаченко Зинаида Арифович';
```

name	num	tarif	data	dop
"DTK"	1506582051	"ПРЕМИУМ HD"	"2020-06-16"	(Null)
"Trinity"	1936926562	"Минимальный"	"2020-12-13"	(Null)
"NEW-TON"	1275614	"Расширенный HD"	"2020-12-01"	(Null)
"ДонАпекс"	324027997	"Mega HD (100 Мбит + 190 IPTV)"	"2020-08-26"	(Null)
"ДонАпекс"	2139548554	"Социальный"	"2021-02-25"	(Null)

Выборки с использованием jsonpath:

```
SELECT data_jsonb->>'name',
       data_jsonb->>'address',
       data_jsonb->>'phone'
FROM json_table
WHERE data_jsonb @> '$.dogovor[*].firma.name == "Trinity"';
```

name	address	phone
Тяпкин Николь Валентинович	ул. Мушкетовская, д. 24	+38192275809060
Уваров Нина Валентиновна	ул. Ленина, д.15	+38149610129790
Удовиченко Олеся Валерьевич	пл. Победы, д. 27	+38180442053116
Утка Рене Владимировна	ул. Кавченко, 7	+38146565342218
Фероян Анжелика Жасурович	ул. Мушкетовская, д. 24	+38204288216225
Фидинчик Арина Ивановна	ул. Уральская, д. 17	+38128256385919

```
SELECT
  DISTINCT jsonb_path_query(data_jsonb, '$.dogovor[*].firma.name') as speed
FROM json_table
WHERE data_jsonb @> 'exists($.dogovor[*] ? (@.speed < 100 && @.speed > 75)
                    ? (@.price >= 999))';
```

speed
null
"DIDAN"
"EAST.NET"
"ORION"
"RAITCOM"
"TerraLine"
"ДонАпекс"
"Комтел"
"Феникс"

```
SELECT data_jsonb ->>'name',
       jsonb_path_query(data_jsonb, '$.dogovor.size()')
FROM json_table
WHERE jsonb_path_exists(data_jsonb, '$.dogovor[*] ? (@.firma.name=="DTK")
                        ? (@.tarif=="ПРЕМИУМ HD")');
```

?column?	jsonb_path_query
Береза Александр Аркадиевна	5
Даллакян Генрик Ромазанович	5
Даньшин Дамир Рушанович	5
Закутний Георгий Степанович	5
Иванников Людмила Богдановна	5
Иванченко Маргарита Бюльбюльевич	5
Иващенко Мирослава Валерьевна	5
Сухобоков Амина Егоровна	5
Тесленко Данила Ринатович	5

Выборки с использованием функций обработки JSON (не менее 3 штук)

```
SELECT *
FROM jsonb_populate_recordset(null::record,
    (SELECT jsonb_path_query_array(data_jsonb, '$.dogovor[*].firma')
      FROM json_table
      WHERE data_jsonb ->> 'name' = 'Козаченко Зинаида Арифович')
    ) AS ("name" varchar, "phone" varchar, "address" text, "type_sob" text);
```

name	phone	address	type_sob
DTK	+38148795354823	ул. Гаврилова, д. 3	государственный
Trinity	+38157928386836	пл. Победы, д. 27	ЗАО
NEW-TON	+38204577709313	ул. Мушкетовская, д. 24	ЗАО
ДонАпекс	+38177082384096	ул. Мушкетовская, д. 24	ОАО
ДонАпекс	+38172665897113	ул. Шевченко, 9	ЗАО

```
SELECT *
FROM jsonb_populate_record(null::record,
    (SELECT jsonb_path_query(data_jsonb, '$.dogovor[*].firma?(@.name=="ДонАпекс")')
      FROM json_table
      WHERE data_jsonb ->> 'name' = 'Фахрутдинов Светлана Геннадиевна')
    ) AS ("name" varchar, "phone" varchar, "address" text, "type_sob" text);
```

name	phone	address	type_sob
ДонАпекс	+38167854797636	ул. Ленина, д.15	государственный

```
SELECT
    jsonb_each_text(jsonb_path_query(data_jsonb, '$.dogovor[*].firma?(@.name=="DTK")'))
FROM json_table
WHERE jsonb_path_match(data_jsonb, 'exists($.dogovor[*].firma.name ? (@ == "DTK"))')
AND data_jsonb ->> 'name' = 'Козаченко Зинаида Арифович'
```

jsonb_each_text
(name,DTK)
(phone,+38148795354823)
(address,"ул. Гаврилова, д. 3")
(type_sob,государственный)

Обновление/добавление данных:

```
UPDATE json_table
SET data_jsonb = jsonb_set(data_jsonb, '{dogovor,2,dop_usluga}', '{"name": "IPTV",
    "chanel": 120,
    "ip_address": "102.168.183.1",
    "type_ip_address": "static"}')
WHERE data_jsonb ->> 'name' = 'Ришняк Дмитрий Максимовна' RETURNING *;
```

data_jsonb
{"name": "Ришняк Дмитрий Максимовна", "phone": "+38128236631130", "address": "ул. Ленина, 51", "dogovor": [{"data": "2021-01-

```
UPDATE json_table
SET data_jsonb = jsonb_insert(data_jsonb,
                              '{dogovor,2,dop_usluga,ip_address}','"192.168.137.1"')
WHERE jsonb_path_match(data_jsonb, 'exists($.dogovor[*].dop_usluga)')
AND data_jsonb ->> 'type_abonent'='юридическое лицо' RETURNING *;
```

id	data_jsonb
1096	{"name": "Мурас Василиса Григорьевна", "phone": "+38164055191292", "address": "ул. Советская, 79", "dogovor": [
1347	{"name": "Репин Дария Леонтьевич", "phone": "+38108011731582", "address": "просп. Семашко, 27а", "dogovor": [
27	{"name": "Алпеева Вильёта Васильевич", "phone": "+38107683233940", "address": "пр-т Ильича, д. 91", "dogovor":
3	{"name": "Абдуев Алексей Алексадрович", "phone": "+38181634913029", "address": "ул. Гаврилова, д. 3", "dogovor":
9	{"name": "Авраменко Анатолий Андреевич", "phone": "+38188919195372", "address": "бул. Ильича, 8", "dogovor":
10	{"name": "Агаев Ангелина Андреевна", "phone": "+38129122781886", "address": "ул. Орджоникидзе, 155", "dogovo
17	{"name": "Акимов Артур Артемовна", "phone": "+38177464559285", "address": "пр. Металлургов, 35а", "dogovor":

```
UPDATE json_table
SET data_jsonb = (data_jsonb #- '{dogovor,2,dop_usluga,ip_address}')
WHERE data_jsonb ->> 'type_abonent' = 'юридическое лицо' RETURNING data_jsonb;
```

```
data_jsonb
{"name": "Абдуев Алексей Алексадрович", "phone": "+38181634913029", "address": "ул. Гаврилова, д. 3", "dogovor": [{"data": "2020-01-01", "dop_usluga": "ремонт", "ip_address": "192.168.137.1"}]}
{"name": "Авраменко Анатолий Андреевич", "phone": "+38188919195372", "address": "бул. Ильича, 8", "dogovor": [{"data": "2020-01-01", "dop_usluga": "ремонт", "ip_address": "192.168.137.1"}]}
{"name": "Агаев Ангелина Андреевна", "phone": "+38129122781886", "address": "ул. Орджоникидзе, 155", "dogovor": [{"data": "2020-01-01", "dop_usluga": "ремонт", "ip_address": "192.168.137.1"}]}
{"name": "Акимов Артур Артемовна", "phone": "+38177464559285", "address": "пр. Металлургов, 35а", "dogovor": [{"data": "2020-01-01", "dop_usluga": "ремонт", "ip_address": "192.168.137.1"}]}
{"name": "Аксенов Борис Бахтиярович", "phone": "+38106210469782", "address": "ул. Куйбышева, д. 252 ", "dogovor": [{"data": "2020-01-01", "dop_usluga": "ремонт", "ip_address": "192.168.137.1"}]}
{"name": "Алексеев Валерий Бюльбюльевич", "phone": "+38180148922039", "address": "бул. Ильича, 8", "dogovor": [{"data": "2020-01-01", "dop_usluga": "ремонт", "ip_address": "192.168.137.1"}]}
{"name": "Алехнович Вера Валериевич", "phone": "+38139985229870", "address": "ул. Гаврилова, д. 3", "dogovor": [{"data": "2020-01-01", "dop_usluga": "ремонт", "ip_address": "192.168.137.1"}]}
```

```
UPDATE json_table
SET data_jsonb = data_jsonb - 'dogovor'
WHERE data_jsonb ->> 'name'='Рамазан Геннадий Ибрагимович' RETURNING data_jsonb;
```

```
data_jsonb
{"name": "Рамазан Геннадий Ибрагимович", "phone": "+38148704782371", "address": null, "type_abonent": "предприятие"}
```

```
UPDATE json_table
SET data_jsonb = data_jsonb || '{"address": "Новослободская 496"}'
WHERE data_jsonb ->> 'name'='Рамазан Геннадий Ибрагимович' RETURNING data_jsonb;
```

```
data_jsonb
{"name": "Рамазан Геннадий Ибрагимович", "phone": "+38148704782371", "address": "Новослободская 496", "type_abonent": "пр
```

Удаление:

```
DELETE FROM json_table WHERE data_jsonb ->> 'name'='Цыбенко Каролина Усама';
```

```
DELETE FROM json_table WHERE data_jsonb ->> 'name'='Цыбенко Каролина Усама'
> Affected rows: 1
> Time: 0,017s
```

```
DELETE
FROM json_table
WHERE
  jsonb_path_match(data_jsonb, 'exists($.dogovor[*]?(@.type_addres=="dynamic"))');
```

```
DELETE FROM json_table
  WHERE jsonb_path_match(data_jsonb, 'exists($.dogovor[*]? (@.type_adres == "dynamic"))')
> Affected rows: 1
> Time: 0,019s
```

Создание индексов:

```
CREATE INDEX json_table_data_jsonb_idx ON json_table
  USING GIN (data_jsonb);
CREATE INDEX json_table_data_jsonb_dogovor_idx ON json_table
  USING GIN ((data_jsonb-> 'dogovor'));
CREATE INDEX json_table_data_jsonb_name_idx ON json_table
  USING GIN ((data_jsonb-> 'name'));
```

Анализ и оптимизация запросов:

Исходный запрос:

```
EXPLAIN (ANALYZE)
  SELECT data_jsonb ->> 'name' as "name",
         data_jsonb ->> 'phone' as phone,
         data_jsonb ->> 'address' as address,
         data_jsonb ->> 'dogovor' as dogovor
  FROM json_table
  WHERE data_jsonb ->> 'type_abonent' = 'предприятие' ORDER BY "name", address;
```

QUERY PLAN

```
Sort (cost=18841.57..18842.87 rows=518 width=128) (actual time=4199.246..4390.099 rows=20964 loops=1)
  Sort Key: ((data_jsonb ->> 'name'::text)), ((data_jsonb ->> 'address'::text))
  Sort Method: external merge  Disk: 39912kB
-> Seq Scan on json_table (cost=0.00..18818.22 rows=518 width=128) (actual time=0.458..3686.145 rows=20964 loops=1)
  Filter: ((data_jsonb ->> 'type_abonent'::text) = 'предприятие'::text)
  Rows Removed by Filter: 82572
Planning Time: 0.307 ms
Execution Time: 4401.948 ms
```

Оптимизированный запрос:

```
EXPLAIN (ANALYZE)
  SELECT data_jsonb ->> 'name' as "name",
         data_jsonb ->> 'phone' as phone,
         data_jsonb ->> 'address' as address,
         data_jsonb ->> 'dogovor' as dogovor
  FROM json_table
  WHERE data_jsonb @> '{"type_abonent": "предприятие"}' ORDER BY "name", address;
```

QUERY PLAN

```
Sort (cost=418.41..418.67 rows=104 width=128) (actual time=2802.121..3006.726 rows=20964 loops=1)
  Sort Key: ((data_jsonb ->> 'name'::text)), ((data_jsonb ->> 'address'::text))
  Sort Method: external merge  Disk: 39928kB
-> Bitmap Heap Scan on json_table (cost=20.80..414.92 rows=104 width=128) (actual time=11.559..2303.995 rows=20964 loop
  Recheck Cond: (data_jsonb @> '{"type_abonent": "предприятие"}'::jsonb)
  Heap Blocks: exact=12569
-> Bitmap Index Scan on json_table_data_jsonb_idx (cost=0.00..20.78 rows=104 width=0) (actual time=8.070..8.071 rows=;
  Index Cond: (data_jsonb @> '{"type_abonent": "предприятие"}'::jsonb)
Planning Time: 0.879 ms
Execution Time: 3018.863 ms
```


По результатам видим, что:

- оценка затратности операции (cost) уменьшилась значительно, было – 18842.87, стало – 418.67;
- по затраченному времени (Execution Time) на запрос показатель уменьшился, было – 4401.948 ms, стало – 3018.863 ms;
- последовательное сканирование (Seq Scan) сменилось на Bitmap Index Scan (используется индекс json_table_data_jsonb_idx для определения нужных нам записей) а затем PostgreSQL лезет в саму таблицу: (Bitmap Heap Scan), чтобы убедиться, что эти записи на самом деле существуют.)

Исходный запрос:

EXPLAIN (ANALYZE)

```
SELECT data_jsonb ->> 'name' as "name",
       data_jsonb #>> '{dogovor,1,firma,name}' as name_company,
       data_jsonb #>> '{dogovor,1,tarif}' as tarif,
       data_jsonb #>> '{dogovor,1,price}' as price
FROM json_table
WHERE data_jsonb->> 'type_abonent'='юридическое лицо'
      AND jsonb_array_length(data_jsonb->'dogovor')>1
ORDER BY price DESC, name_company, tarif;
```

QUERY PLAN

```
Gather Merge (cost=19233.61..19250.42 rows=144 width=128) (actual time=1467.874..1531.255 rows=20614 loops=1)
  Workers Planned: 2
  Workers Launched: 2
  -> Sort (cost=18233.59..18233.77 rows=72 width=128) (actual time=1373.797..1375.098 rows=6871 loops=3)
    Sort Key: ((data_jsonb #>> '{dogovor,1,price}':text[])) DESC, ((data_jsonb #>> '{dogovor,1,firma,name}':text[])), ((data_jsonb #>> '{dogovor,1,tarif}':text[]))
    Sort Method: quicksort Memory: 1281kB
    Worker 0: Sort Method: quicksort Memory: 1130kB
    Worker 1: Sort Method: quicksort Memory: 1111kB
  -> Parallel Seq Scan on json_table (cost=0.00..18231.37 rows=72 width=128) (actual time=1.528..1299.614 rows=6871 loops=3)
    Filter: (((data_jsonb ->> 'type_abonent':text) = 'юридическое лицо':text) AND (jsonb_array_length((data_jsonb -> 'dogovor':text)) > 1))
    Rows Removed by Filter: 27641
Planning Time: 0.288 ms
Execution Time: 1532.764 ms
```

Оптимизированный запрос:

```
EXPLAIN (ANALYZE) SELECT data_jsonb ->> 'name' as "name",
       data_jsonb #>> '{dogovor,1,firma,name}' as name_company,
       data_jsonb #>> '{dogovor,1,tarif}' as tarif,
       data_jsonb #>> '{dogovor,1,price}' as price
FROM json_table
WHERE data_jsonb @@ '$.type_abonent == "юридическое лицо"'
      AND jsonb_array_length(data_jsonb->'dogovor')>1
ORDER BY price DESC, name_company, tarif;
```

QUERY PLAN

```
Sort (cost=415.89..415.98 rows=35 width=128) (actual time=2449.833..2454.290 rows=20614 loops=1)
  Sort Key: ((data_jsonb #>> '{dogovor,1,price}':text[])) DESC, ((data_jsonb #>> '{dogovor,1,firma,name}':text[])), ((data_jsonb #>> '{dogovor,1,tarif}':text[]))
  Sort Method: quicksort Memory: 3521kB
  -> Bitmap Heap Scan on json_table (cost=20.79..415.00 rows=35 width=128) (actual time=18.021..2076.376 rows=20614 loops=1)
    Recheck Cond: (data_jsonb @@ '$.type_abonent' == "юридическое лицо")::jsonpath
    Filter: (jsonb_array_length((data_jsonb -> 'dogovor':text)) > 1)
    Heap Blocks: exact=12206
  -> Bitmap Index Scan on json_table_data_jsonb_idx (cost=0.00..20.78 rows=104 width=0) (actual time=12.577..12.577 rows=20614 loops=1)
    Index Cond: (data_jsonb @@ '$.type_abonent' == "юридическое лицо")::jsonpath
Planning Time: 0.345 ms
Execution Time: 2457.088 ms
```

По результатам видим, что:

- оценка затратности операции (cost) уменьшилась значительно, было – 19250.42, стало – 415.98;
- по затраченному времени (Execution Time) на запрос показатель увеличился, было – 1532.764 ms, стало – 2457.088 ms;
- последовательное сканирование (Seq Scan) сменилось на Bitmap Index Scan (используется индекс json_table_data_jsonb_idx для определения нужных нам записей).

Исходный запрос:

EXPLAIN (ANALYZE)

```
SELECT (jsonb_array_elements(data_jsonb->'dogovor'))->'firma'->>'name' as "name",  
       (jsonb_array_elements(data_jsonb->'dogovor'))->>'number_dogovora' as num,  
       (jsonb_array_elements(data_jsonb->'dogovor'))->>'tarif' as tarif,  
       (jsonb_array_elements(data_jsonb->'dogovor'))->>'data' as "data",  
       (jsonb_array_elements(data_jsonb->'dogovor'))->>'dop_usluga' as dop  
FROM json_table  
WHERE data_jsonb ->>'name'='Козаченко Зинаида Арифович';
```

QUERY PLAN
Gather (cost=1000.00..20149.06 rows=518 width=160) (actual time=37.996..832.738 rows=165 loops=1)
Workers Planned: 2
Workers Launched: 2
-> Result (cost=0.00..19097.26 rows=21600 width=160) (actual time=16.949..725.722 rows=55 loops=3)
-> ProjectSet (cost=0.00..18017.26 rows=21600 width=32) (actual time=16.926..725.422 rows=55 loops=3)
-> Parallel Seq Scan on json_table (cost=0.00..17907.10 rows=216 width=307) (actual time=16.804..724.971 rows=10 loops=3)
Filter: ((data_jsonb ->> 'name'::text) = 'Козаченко Зинаида Арифович'::text)
Rows Removed by Filter: 34502
Planning Time: 0.167 ms
Execution Time: 832.790 ms

Оптимизированный запрос:

EXPLAIN (ANALYZE)

```
SELECT (jsonb_array_elements(data_jsonb->'dogovor'))->'firma'->>'name' as "name",  
       (jsonb_array_elements(data_jsonb->'dogovor'))->>'number_dogovora' as num,  
       (jsonb_array_elements(data_jsonb->'dogovor'))->>'tarif' as tarif,  
       (jsonb_array_elements(data_jsonb->'dogovor'))->>'data' as "data",  
       (jsonb_array_elements(data_jsonb->'dogovor'))->>'dop_usluga' as dop  
FROM json_table  
WHERE data_jsonb @> '{"name": "Козаченко Зинаида Арифович"}';
```

QUERY PLAN
Result (cost=20.80..986.92 rows=10400 width=160) (actual time=0.226..3.551 rows=165 loops=1)
-> ProjectSet (cost=20.80..466.92 rows=10400 width=32) (actual time=0.217..2.526 rows=165 loops=1)
-> Bitmap Heap Scan on json_table (cost=20.80..413.88 rows=104 width=307) (actual time=0.174..1.477 rows=29 loops=1)
Recheck Cond: (data_jsonb @> '{"name": "Козаченко Зинаида Арифович"}'::jsonb)
Heap Blocks: exact=29
-> Bitmap Index Scan on json_table_data_jsonb_idx (cost=0.00..20.78 rows=104 width=0) (actual time=0.106..0.106 rows=29 loops=1)
Index Cond: (data_jsonb @> '{"name": "Козаченко Зинаида Арифович"}'::jsonb)
Planning Time: 0.357 ms
Execution Time: 3.675 ms

По результатам видим что:

- оценка затратности операции (cost) уменьшилась значительно, было – 20149.06, стало – 986.92;

- по затраченному времени (Execution Time) на запрос показатель уменьшился значительно, было – 832.790 ms, стало – 3.675 ms;
- последовательное сканирование (Seq Scan) сменилось на Bitmap Index Scan (используется индекс json_table_data_jsonb_idx для определения нужных нам записей).

Исходный запрос:

```
EXPLAIN (ANALYZE) SELECT data_jsonb->>'name',
                        data_jsonb->>'address',
                        data_jsonb->>'phone'
FROM json_table
WHERE data_jsonb @@ '$.dogovor[*].firma.name == "Trinity";
```

QUERY PLAN
Bitmap Heap Scan on json_table (cost=20.80..414.66 rows=104 width=96) (actual time=8.208..2204.111 rows=34060 loops=1)
Recheck Cond: (data_jsonb @@ '\$.dogovor[*].firma.name' == "Trinity")::jsonpath)
Heap Blocks: exact=13208
-> Bitmap Index Scan on json_table_data_jsonb_idx (cost=0.00..20.78 rows=104 width=0) (actual time=6.189..6.189 rows=34060 loops=1)
Index Cond: (data_jsonb @@ '\$.dogovor[*].firma.name' == "Trinity")::jsonpath)
Planning Time: 0.553 ms
Execution Time: 2207.697 ms

Данный запрос можно оставить в прежнем виде т.к. по результатам видим что используется сканирование по индексу (Bitmap Index Scan on json_table_data_jsonb_idx), оценка затратности операции (cost) составляет 414.66, что является небольшим по сравнению с предыдущими запросами. По затраченному времени (Execution Time) на запрос показатель равен 2207.697 ms, что также является не большим.

Исходный запрос:

```
EXPLAIN (ANALYZE) SELECT data_jsonb ->>'name',
                        jsonb_path_query(data_jsonb, '$.dogovor.size()')
FROM json_table
WHERE jsonb_path_exists(data_jsonb, '$.dogovor[*] ? (@.firma.name=="DTK")
                        ? (@.tarif=="ПРЕМИУМ HD")');
```

QUERY PLAN
Gather (cost=1000.00..94294.25 rows=34512 width=64) (actual time=1.184..1080.607 rows=2636 loops=1)
Workers Planned: 2
Workers Launched: 2
-> ProjectSet (cost=0.00..89843.05 rows=14380000 width=64) (actual time=0.429..978.150 rows=879 loops=3)
-> Parallel Seq Scan on json_table (cost=0.00..17799.25 rows=14380 width=307) (actual time=0.393..929.569 rows=879 loops=3)
Filter: jsonb_path_exists(data_jsonb, '\$.dogovor[*]?(@.firma.name=="DTK")?(@.tarif=="ПРЕМИУМ HD")::jsonpath, '{}'::jsonb, false)
Rows Removed by Filter: 33633
Planning Time: 0.107 ms
Execution Time: 1081.004 ms

Оптимизированный запрос:

```
EXPLAIN (ANALYZE) SELECT data_jsonb ->>'name',
                        jsonb_path_query(data_jsonb, '$.dogovor.size()')
FROM json_table
WHERE data_jsonb @@ 'exists($.dogovor[*] ? (@.firma.name=="DTK")
                        ? (@.tarif=="ПРЕМИУМ HD"))';
```

QUERY PLAN
ProjectSet (cost=36.80..950.92 rows=104000 width=64) (actual time=7.307..438.330 rows=2636 loops=1)
-> Bitmap Heap Scan on json_table (cost=36.80..429.88 rows=104 width=307) (actual time=7.293..332.010 rows=2636 loops=1)
Recheck Cond: (data_jsonb @@ 'exists (\$.dogovor["*"]?(@."firma"."name" == "ДТК")?(@."тариф" == "ПРЕМИУМ HD"))::jsonpath)
Rows Removed by Index Recheck: 8524
Heap Blocks: exact=6586
-> Bitmap Index Scan on json_table_data_jsonb_idx (cost=0.00..36.78 rows=104 width=0) (actual time=6.011..6.011 rows=1160 loops=1)
Index Cond: (data_jsonb @@ 'exists (\$.dogovor["*"]?(@."firma"."name" == "ДТК")?(@."тариф" == "ПРЕМИУМ HD"))::jsonpath)
Planning Time: 0.115 ms
Execution Time: 438.933 ms

По результатам видим что:

- оценка затратности операции (cost) уменьшилась значительно, было – 94294.25, стало – 950.92;
- по затраченному времени (Execution Time) на запрос показатель уменьшился значительно, было – 1081.004 ms, стало – 438.933 ms;
- последовательное сканирование (Seq Scan) сменилось на Bitmap Index Scan (используется индекс json_table_data_jsonb_idx для определения нужных нам записей).

Исходный запрос:

```
EXPLAIN (ANALYZE)
SELECT
  jsonb_each_text(jsonb_path_query(data_jsonb, '$.dogovor[*].firma?(@.name=="ДТК")'))
FROM json_table
WHERE jsonb_path_match(data_jsonb, 'exists ($.dogovor[*].firma.name?(@ == "ДТК"))')
      AND data_jsonb ->> 'name'='Козаченко Зинаида Арифович';
```

QUERY PLAN
Gather (cost=1000.00..56112.79 rows=173 width=32) (actual time=399.799..1258.639 rows=36 loops=1)
Workers Planned: 2
Workers Launched: 2
-> ProjectSet (cost=0.00..55095.49 rows=7200000 width=32) (actual time=417.783..1131.448 rows=12 loops=3)
-> ProjectSet (cost=0.00..18375.49 rows=72000 width=32) (actual time=417.734..1131.344 rows=3 loops=3)
-> Parallel Seq Scan on json_table (cost=0.00..18014.95 rows=72 width=307) (actual time=417.693..1131.231 rows=3 loops=3)
Filter: (jsonb_path_match(data_jsonb, 'exists (\$.dogovor["*"].firma.name?(@ == "ДТК"))::jsonpath, '{}'::jsonb, false) AND ((data_jsonb ->> 'name)::text) = 'Козаченко Зинаида Арифович::text))
Rows Removed by Filter: 34509
Planning Time: 0.313 ms
Execution Time: 1258.702 ms

Оптимизированный запрос:

```
EXPLAIN (ANALYZE)
SELECT
  jsonb_each_text(jsonb_path_query(data_jsonb, '$.dogovor[*].firma?(@.name=="ДТК")'))
FROM json_table
WHERE data_jsonb @@ 'exists ($.dogovor[*].firma.name ? (@ == "ДТК"))' AND
      data_jsonb @> '{"name": "Козаченко Зинаида Арифович"}';
```

QUERY PLAN
ProjectSet (cost=36.00..555.02 rows=100000 width=32) (actual time=0.616..1.089 rows=36 loops=1)
-> ProjectSet (cost=36.00..45.02 rows=1000 width=32) (actual time=0.604..0.989 rows=9 loops=1)
-> Bitmap Heap Scan on json_table (cost=36.00..40.02 rows=1 width=307) (actual time=0.591..0.850 rows=8 loops=1)
Recheck Cond: (((data_jsonb @@ 'exists (\$.dogovor["*"].firma.name?(@ == "ДТК"))::jsonpath) AND (data_jsonb @> '{"name": "Козаченко Зинаида Арифович"}::jsonb))
Heap Blocks: exact=8
-> Bitmap Index Scan on json_table_data_jsonb_idx (cost=0.00..36.00 rows=1 width=0) (actual time=0.553..0.553 rows=8 loops=1)
Index Cond: (((data_jsonb @@ 'exists (\$.dogovor["*"].firma.name?(@ == "ДТК"))::jsonpath) AND (data_jsonb @> '{"name": "Козаченко Зинаида Арифович"}::jsonb))
Planning Time: 0.152 ms
Execution Time: 1.160 ms

По результатам видим что:

- оценка затратности операции (cost) уменьшилась значительно, было – 56112.79, стало – 555.02;
- по затраченному времени (Execution Time) на запрос показатель уменьшился значительно, было – 1258.702 ms, стало – 1.160 ms;
- последовательное сканирование (Seq Scan) сменилось на Bitmap Index Scan (используется индекс json_table_data_jsonb_idx для определения нужных нам записей).