

**Министерство науки и высшего образования Российской Федерации**  
федеральное государственное автономное образовательное учреждение высшего  
образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**Отчет**  
**По Лабораторной работе 2**  
**ЗАПРОСЫ НА ВЫБОРКУ И МОДИФИКАЦИЮ ДАННЫХ, ПРЕДСТАВЛЕНИЯ И ИНДЕКСЫ В**  
**POSTGRESQL**  
**по дисциплине «Базы данных»**

Автор: Филипп. К.Е.

Факультет: ИКТ

Группа: K32402

Преподаватель: Говорова М. М.

Дата: 22.03.2023

**Цель работы:** овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL, использования подзапросов при модификации данных и индексов.

**Программное обеспечение:** СУБД PostgreSQL 1X, pgAdmin 4.

**Практическое задание:**

1. Создать запросы и представления на выборку данных к базе данных PostgreSQL (согласно индивидуальному заданию, часть 2 и 3).
2. Составить 3 запроса на модификацию данных (INSERT, UPDATE, DELETE) с использованием подзапросов.
3. Изучить графическое представление запросов и просмотреть историю запросов.
4. Создать простой и составной индексы для двух произвольных запросов и сравнить время выполнения запросов без индексов и с индексами. Для получения плана запроса использовать команду EXPLAIN.

**Индивидуальное практическое задание:**

Предметная область «**Банк**».

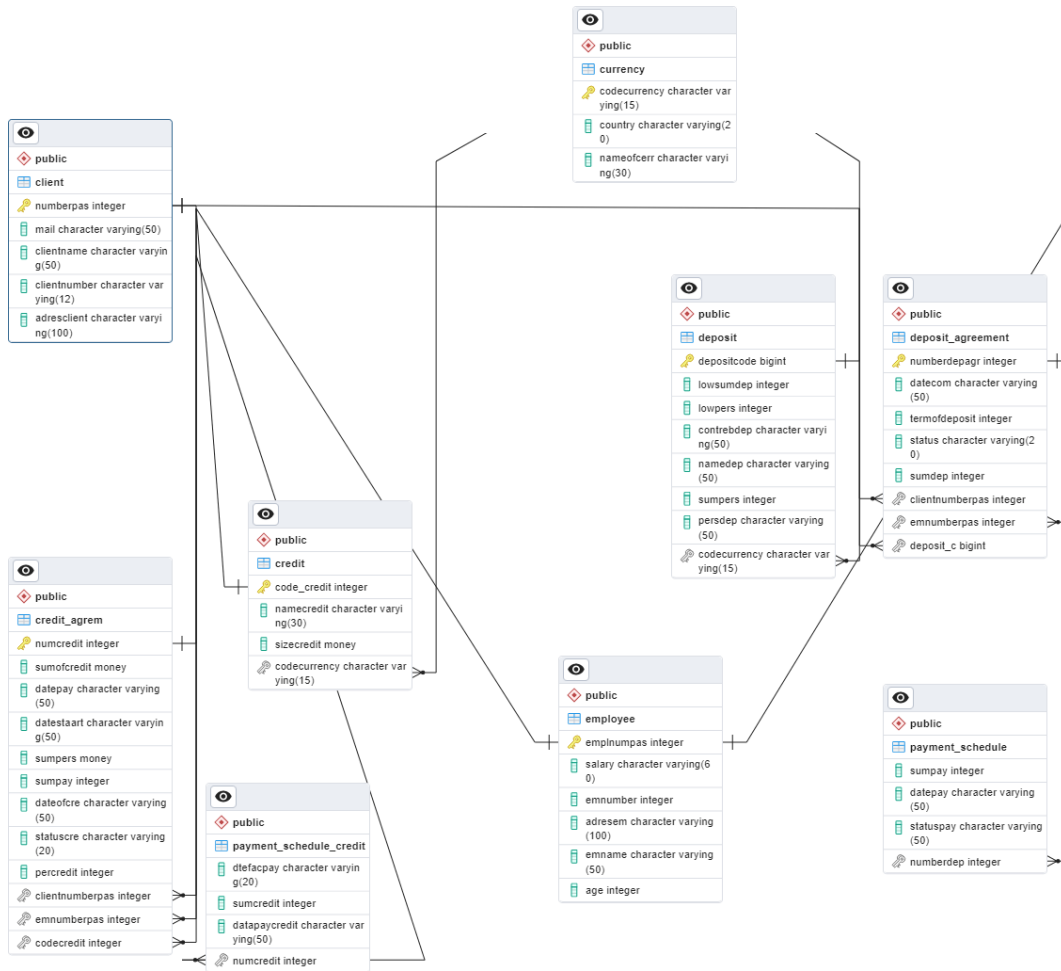
Составьте запросы на выборку:

- Найти вкладчика, имеющего на текущий день несколько вкладов.
- Найти вкладчика, имеющего вклады во всех видах валюты на текущий день.
- Вывести данные вкладчика, имеющего максимальный вклад в английский фунтах.
- Какой из вкладов пользовался наибольшей популярностью за истекший год.
- Кто из сотрудников заключил максимальное число договоров по кредитам за последний месяц.
- Вывести список вкладчиков, у которых срок вклада истекает завтра и суммы начислений, которые могут быть ими востребованы.
- Вывести список сотрудников, заключивших договоры по вкладам на максимальную сумму за последний месяц.

Создать представление:

- содержащее сведения обо всех сотрудниках банка и заключенных ими договорах по кредитам за прошедший месяц;
- найти клиентов банка, имеющих задолженности по кредитам.

## II. схема логической модели базы данных, сгенерированная в Generate ERD



## SELECT

1 Найти вкладчика, имеющего на текущий день несколько вкладов

```

1 SELECT client_number_pas
2 FROM deposit_agreement
3 WHERE date_com <= CURRENT_DATE
4 GROUP BY client_number_pas
5 HAVING COUNT(*) > 1;

```

Data Output   Сообщения   Notifications



	client_number_pas character varying (10) 🔒
1	154974
2	421147964
3	953145

2 Найти вкладчика, имеющего вклады во всех видах валюты на текущий день.

```

1 SELECT client_number_pas
2 FROM deposit_agreement
3 WHERE date_com <= CURRENT_DATE
4 GROUP BY client_number_pas
5 HAVING COUNT(DISTINCT deposit_c) = (SELECT COUNT(DISTINCT code_currency) FROM deposit);

```

	client_number_pas character varying (10) 🔒
1	154974

3 Вывести данные вкладчика, имеющего максимальный вклад в английский фунтах

```
1 SELECT c.*, d.deposit_c
2 from client c
3 inner join deposit_agreement d on c.number_pas = d.client_number_pas
4 where date_com > current_date - interval '1 year'
5 group by deposit_c;
```

	number_pas character varying (10)	mail character varying (50)	cl_name character varying (50)	cl_number character varying (11)	adres_cl character varying (100)	deposit_c bigint
1	154974	mail@mail.com	Семен	896532147	str	2

4 Какой из вкладов пользовался наибольшей популярностью за истекший год

Запрос

История запросов

1

SELECT deposit\_c, COUNT (\*) AS col

2

FROM deposit\_agreement

3

WHERE date\_com >CURRENT\_DATE - INTERVAL '1 year'

4

GROUP BY deposit\_c

5

ORDER BY col;

Data Output

Сообщения

Notifications

deposit\_c

bigint

col

bigint

1

2

1

2

1

6

5 Кто из сотрудников заключил максимальное число договоров по кредитам за последний месяц

```
1 SELECT s.em_name, COUNT (d.em_number_pas) AS Col_agre
2 FROM employee s
3 JOIN deposit_agreement d on s.empl_num_pas = d.em_number_pas
4 WHERE d.date_com > CURRENT_DATE - INTERVAL '1 month'
5 GROUP BY s.em_name
6 ORDER BY Col_agre DESC;
```

Data Output

Сообщения

Notifications

em\_name

character varying (50)

col\_agre

bigint

1	Виктория	1
---	----------	---

6 Вывести список вкладчиков, у которых срок вклада истекает завтра и суммы начислений, которые могут быть ими востребованы.

```

1 SELECT v.cl_name, w.term_of_deposit
2 from client v
3 join deposit_agreement w ON v.number_pas = w.client_number_pas
4 WHERE w.date_com > CURRENT_DATE + interval '1 day'

```

	cl_name character varying (50) 🔒	term_of_deposit integer 🔒
1	Коля	10

7 Вывести список сотрудников, заключивших договоры по вкладам на максимальную сумму за последний месяц.

```

1 SELECT s.em_name, SUM(d.sum_dep+d.term_of_deposit) AS all_sum
2 from employee s
3 join deposit_agreement d ON s.empl_num_pas = d.em_number_pas
4 WHERE d.date_com > CURRENT_DATE - interval '1 month'
5 GROUP BY s.em_name;

```

	em_name character varying (50) 🔒	all_sum bigint 🔒
1	Виктория	36275

## INDEX, INSERT, UPDATE, DELETE

Создаем запрос для проверки работы индекса

```

1 EXPLAIN ANALYZE
2 select c.cl_name, o.number_dep_agr from client c
3 join deposit_agreement o ON c.cl_number=o.client_number_pas
4 group by c.cl_name, o.number_dep_agr
5

```

Planning Time: 6.571 ms  
Execution Time: 0.701 ms

## Создаем простой индекс

```
1 CREATE INDEX cl_names ON client(cl_name);
```

Data Output   Сообщения   Notifications

CREATE INDEX

Запрос завершён успешно, время выполнения: 92 msec.

**Planning Time: 5.230 ms**

**Execution Time: 0.590 ms**

## Составной индекс

```
1 CREATE INDEX idx_deposit_agreement_client_number_pas_sum_dep  
2 ON deposit_agreement (client_number_pas, sum_dep);
```

Data Output   Сообщения   Notifications

CREATE INDEX

Запрос завершён успешно, время выполнения: 78 msec.

**После создания индекса время запроса уменьшилась на 0.2 секунды**

## Update

```
1 UPDATE deposit_agreement AS d
2 SET status = 'error'
3 FROM client AS c
4 WHERE d.client_number_pas = c.number_pas
5 AND c.cl_name = 'Коля'
```

Data Output   Сообщения   Notifications

UPDATE 3

Запрос завершён успешно, время выполнения: 64 msec.

## DELETE



```

1
2
3 DELETE FROM deposit_agreement
4 WHERE em_number_pas IN (
5     SELECT empl_num_pas
6     FROM employee
7     WHERE em_name = 'Sam'
8 );
9

```

Data Output   **Сообщения**   Notifications

DELETE 0

Запрос завершён успешно, время выполнения: 66 мсек.

## INSERT

```

1 INSERT INTO deposit_agreements (number_dep_agr, date_com, term_of_deposit, status, client_number_pas, sum_dep)
2 SELECT
3     nextval('deposit_agreement_sequence'),
4     '21-04-2023'::date,
5     2323,
6     'done',
7     c.number_pas,
8     323
9 FROM
10     clients c
11 WHERE
12     NOT EXISTS (
13         SELECT 1
14         FROM deposit_agreements d
15         WHERE d.client_number_pas = c.number_pas
16         AND d.date_com >= NOW() - INTERVAL '1 year'
17     );
18

```

Data Output   **Сообщения**   Notifications

INSERT 0 2

Запрос завершён успешно, время выполнения: 165 мсек.

**Представление (VIEW)** — объект базы данных, являющийся результатом выполнения запроса к базе данных, определенного с помощью оператора SELECT, в момент обращения к представлению

1 Задание: содержащее сведения обо всех сотрудниках банка и заключенных ими договорах по кредитам за прошедший месяц

```
1 CREATE VIEW employees_and_credit_contracts AS
2 SELECT
3     e.empl_num_pas,
4     e.em_name,
5     c.num_credit,
6     c.date_staart,
7     c.per_credit
8 FROM
9     employee e
10 JOIN
11     credit_agrem c ON e.empl_num_pas = c.client_number_pas
12 WHERE
13     c.date_staart >= date_trunc('month', current_date) - interval '1 month'
14     AND c.date_staart < date_trunc('month', current_date);
15
16
17 SELECT * FROM employees_and_credit_contracts;
18
```

2 Задача: найти клиентов банка, имеющих задолженности по кредитам.

```
1 CREATE VIEW indebtedted_clients AS
2 SELECT
3     c.number_pas,
4     c.cl_name,
5     c.adres_cl,
6     c.cl_number,
7     cc.num_credit,
8     cc.per_credit,
9     cc.sum_pay_c
10 FROM
11     client c
12 JOIN
13     credit_agrem cc ON c.number_pas = cc.client_number_pas
14 WHERE
15     cc.per_credit > cc.sum_pay_c;
16 |
```

Data Output   Сообщения   Notifications

CREATE VIEW

Запрос завершён успешно, время выполнения: 174 msec.

**Выводы:**

В процессе работы с pgAdmin я научился делать запросы на выборку и модификацию данных в базе данных PostgreSQL. Я освоил создание представлений, которые помогают упростить работу с данными, а также настройку индексов, что повышает производительность работы с базой данных. Эти знания помогут мне более эффективно работать с PostgreSQL и создавать более сложные запросы и представления для обработки большого объема данных. В целом, работа с pgAdmin позволила мне более полно использовать возможности PostgreSQL и повысить эффективность работы с базой данных.