

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»
Факультет инфокоммуникационных технологий

ОТЧЕТ
О ЛАБОРАТОРНОЙ РАБОТЕ № 4
по теме: Запросы на выборку данных к БД PostgreSQL.
ПРЕДСТАВЛЕНИЯ В PostgreSQL
по дисциплине: Проектирование и реализация баз данных

Специальность:

09.03.03 Мобильные и сетевые технологии

Проверил:

Говорова М.М. _____

Дата: «__» ____ 2021 г.

Оценка _____

Выполнил:

студент группы К3241

Матрохина А.С.

Санкт-Петербург 2021 г.

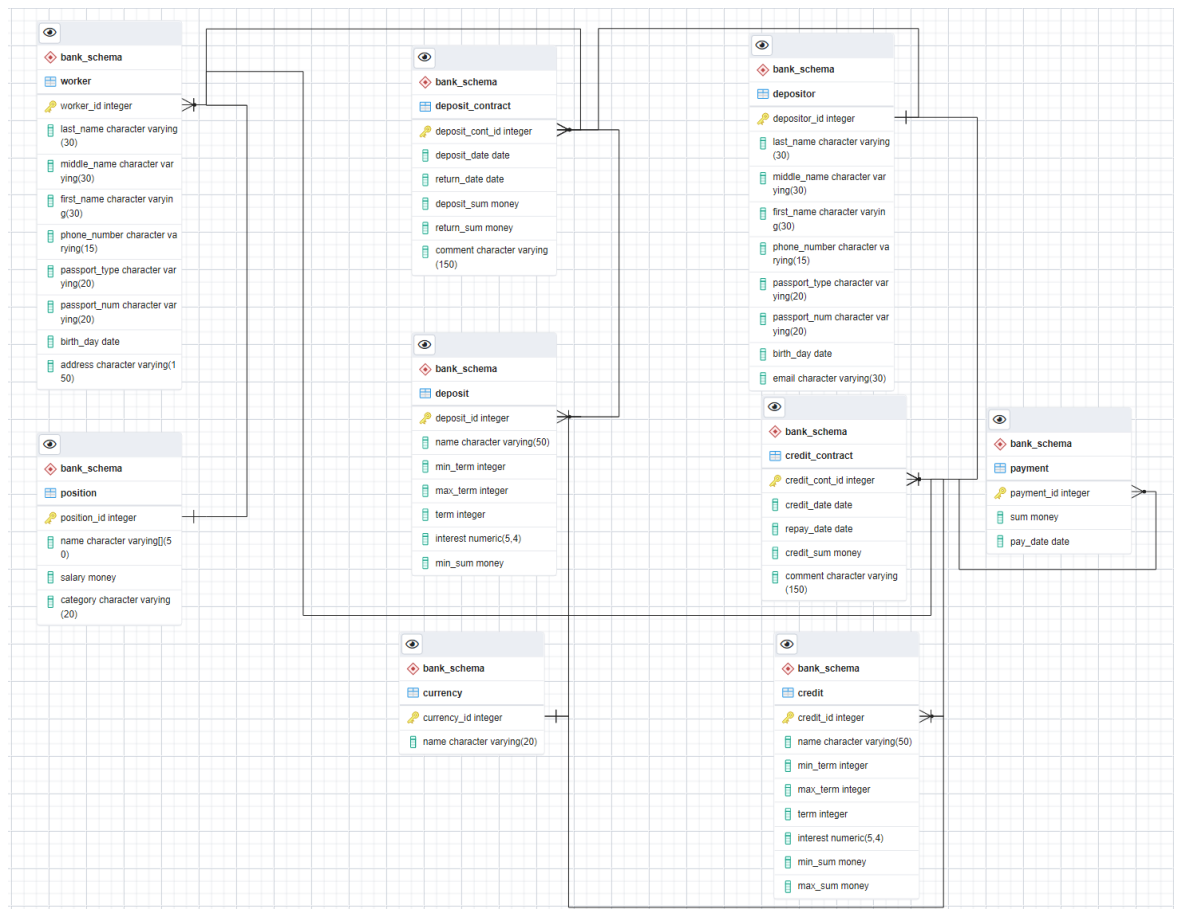
1. Цель работы

Овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL и использования подзапросов при модификации данных.

2. 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.
4. Посмотреть историю запросов.

3. Схема базы данных

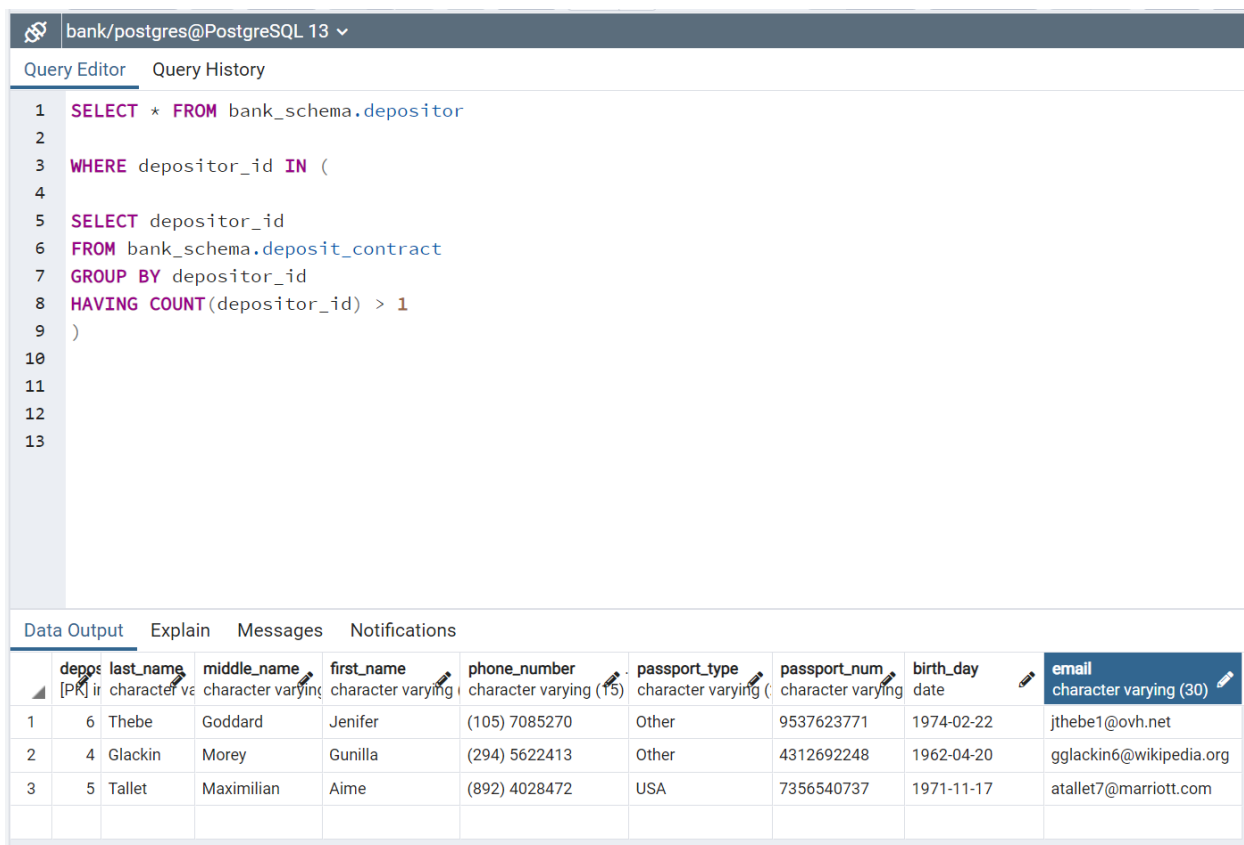


(рис 1 ER диаграмма БД “Банк”)

4. Выполнение

1. Запросы к базе данных

а. Найти вкладчика, имеющего несколько вкладов.



The screenshot shows a PostgreSQL query editor interface. The query editor tab is active, displaying a SQL query. Below the query editor, the 'Data Output' tab is active, showing the results of the query in a table format.

Query:

```
1 SELECT * FROM bank_schema.depositor
2
3 WHERE depositor_id IN (
4
5 SELECT depositor_id
6 FROM bank_schema.deposit_contract
7 GROUP BY depositor_id
8 HAVING COUNT(depositor_id) > 1
9 )
10
11
12
13
```

Data Output Table:

	depositor_id	last_name	middle_name	first_name	phone_number	passport_type	passport_number	birth_day	email
1	6	Thebe	Goddard	Jenifer	(105) 7085270	Other	9537623771	1974-02-22	jthebe1@ovh.net
2	4	Glackin	Morey	Gunilla	(294) 5622413	Other	4312692248	1962-04-20	gglackin6@wikipedia.org
3	5	Tallet	Maximilian	Aime	(892) 4028472	USA	7356540737	1971-11-17	atallet7@marriott.com

(рис 2 вкладчики с несколькими вкладами)

б. Найти вкладчика, имеющего вклады во всех видах валюты

bank/postgres@PostgreSQL 13

Query Editor Query History

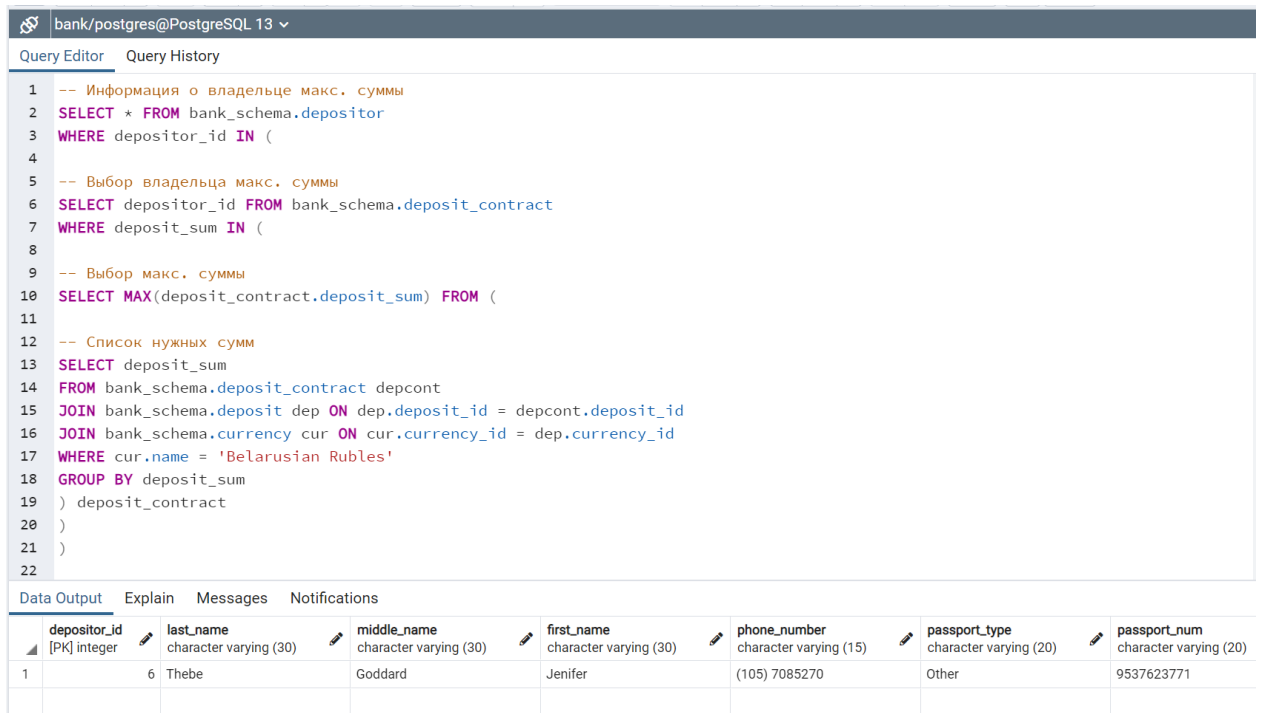
```
1 SELECT * FROM bank_schema.depositor
2 WHERE depositor_id IN (
3
4 SELECT depositor_id
5 FROM bank_schema.deposit_contract depcont
6 JOIN bank_schema.deposit dep ON depcont.deposit_id = dep.deposit_id
7 GROUP BY (depositor_id)
8 -- Посчитать количество вкладов с уникальными валютами, и сравнить их с общим количеством валют
9 HAVING COUNT(DISTINCT currency_id) = (SELECT COUNT(currency_id) FROM bank_schema.currency)
10 )
11
```

Data Output Explain Messages Notifications

	depositor_id [PK] integer	last_name character varying (30)	middle_name character varying (30)	first_name character varying (30)	phone_number character varying (15)	passport_type character varying (20)	passport_num character varying (20)
1	5	Tallet	Maximilian	Aime	(892) 4028472	USA	7356540737

(рис 3 вкладчики со всеми валютами)

с. Вывести данные вкладчика, имеющего максимальный вклад в белорусских рублях.



The screenshot shows a PostgreSQL Query Editor interface. The top bar indicates the connection is to 'bank/postgres@PostgreSQL 13'. Below the bar, there are tabs for 'Query Editor' and 'Query History'. The main area contains a SQL query with several comments in Russian explaining the steps: finding the maximum sum, selecting the depositor ID, and joining tables to get the depositor's details. The query is as follows:

```
1  -- Информация о владельце макс. суммы
2  SELECT * FROM bank_schema.depositor
3  WHERE depositor_id IN (
4
5  -- Выбор владельца макс. суммы
6  SELECT depositor_id FROM bank_schema.deposit_contract
7  WHERE deposit_sum IN (
8
9  -- Выбор макс. суммы
10 SELECT MAX(deposit_contract.deposit_sum) FROM (
11
12 -- Список нужных сумм
13 SELECT deposit_sum
14 FROM bank_schema.deposit_contract depcont
15 JOIN bank_schema.deposit dep ON dep.deposit_id = depcont.deposit_id
16 JOIN bank_schema.currency cur ON cur.currency_id = dep.currency_id
17 WHERE cur.name = 'Belarusian Rubles'
18 GROUP BY deposit_sum
19 ) deposit_contract
20 )
21 )
22
```

Below the query editor, there are tabs for 'Data Output', 'Explain', 'Messages', and 'Notifications'. The 'Data Output' tab is active, showing a table with 8 columns: depositor_id, last_name, middle_name, first_name, phone_number, passport_type, and passport_num. The table contains one row of data for depositor_id 6.

depositor_id	last_name	middle_name	first_name	phone_number	passport_type	passport_num
6	Thebe	Goddard	Jenifer	(105) 7085270	Other	9537623771

(рис 4 макс. сумма)

d. Какой из вкладов пользуется наибольшей популярностью.

bank/postgres@PostgreSQL 13

Query EditorQuery History

```
1 SELECT * FROM bank_schema.deposit
2 WHERE deposit_id IN (
3
4 SELECT deposit_id FROM (
5
6 SELECT deposit_id, COUNT(deposit_id) OVER (PARTITION BY deposit_id)
7 FROM bank_schema.deposit_contract
8 ORDER BY count DESC LIMIT 1
9 ) deposit_contract
10 )
```

Data OutputExplainMessagesNotifications

	deposit_id [PK] integer	name character varying (50)	min_term integer	max_term integer	term integer	interest numeric (5,4)	min_sum money	currency_id integer
1	4	Smelowskia calycina (Steph...	44	124	80	0.0500	10 000,00 ?	1

(рис 5 самый популярный вклад)

bank/postgres@PostgreSQL 13

Query Editor

Query History

1

2

SELECT

deposit_cont_id, deposit_date, return_date, deposit_sum, return_sum, comment, deposit_id, depositor_id, worker_id

FROM bank_schema.deposit_contract;

Data Output

Explain

Messages

Notifications

	deposit_cont_id [PK] integer	deposit_date date	return_date date	deposit_sum money	return_sum money	comment character varying (150)	deposit_id integer	depositor_id integer	worker_id integer
1	5	2021-04-10	2021-06-23	116 346,00 ?	143 208,00 ?	[null]		1	10
2	7	2021-04-28	2021-06-05	117 739,00 ?	163 761,00 ?	[null]		4	5
3	10	2021-04-05	2021-06-04	108 321,00 ?	147 534,00 ?	[null]		2	3
4	11	2021-04-20	2021-06-25	100 000,00 ?	440 000,00 ?	[null]		4	4
5	2	2021-04-19	2021-06-15	107 771,00 ?	144 430,00 ?	[null]		1	4
6	6	2021-04-27	2021-06-15	105 991,00 ?	149 983,00 ?	[null]		3	6
7	8	2021-04-04	2021-06-16	104 927,00 ?	171 234,00 ?	[null]		8	5
8	1	2021-05-25	2021-06-05	119 132,00 ?	160 166,00 ?	[null]		5	4
9	4	2021-05-20	2021-06-10	117 397,00 ?	163 386,00 ?	[null]		4	1
10	9	2021-05-21	2021-06-13	106 469,00 ?	155 470,00 ?	[null]		7	5
11	3	2021-05-24	2021-06-15	119 166,00 ?	169 618,00 ?	[null]		3	6

(рис 6 исходные данные в договорах о депозитах)

е. Кто из сотрудников заключил максимальное число договоров.

bank/postgres@PostgreSQL 13

Query Editor

Query History

```
1 SELECT * FROM bank_schema.worker worker
2 WHERE worker.worker_id IN (
3 -- Выбор сотрудника с максимальным кол-вом договоров
4 SELECT worker_id FROM (
5 -- Суммирование кол-ва договоров разного вида у работника
6 SELECT DISTINCT counts.worker_id, SUM(counts.count) OVER (PARTITION BY counts.worker_id) FROM (
7 -- Работник - количество вкладовых договоров
8 (SELECT DISTINCT worker_id, COUNT(deposit_cont_id) OVER(PARTITION BY worker_id)
9 FROM bank_schema.deposit_contract)
10 UNION
11 -- Работник - количество кредитных договоров
12 (SELECT DISTINCT worker_id, COUNT(credit_cont_id) OVER(PARTITION BY worker_id)
13 FROM bank_schema.credit_contract)
14 ) counts
15 ) counts_sum
16 WHERE counts_sum.sum IN (
17 -- Выбор максимального кол-ва договоров
18 SELECT MAX(counts_max.sum) FROM (
19 -- Суммирование кол-ва договоров разного вида у работника
20 SELECT DISTINCT counts.worker_id, SUM(counts.count) OVER (PARTITION BY counts.worker_id) FROM (
21 -- Работник - количество вкладовых договоров
22 (SELECT DISTINCT worker_id, COUNT(deposit_cont_id) OVER(PARTITION BY worker_id)
23 FROM bank_schema.deposit_contract)
24 UNION
25 -- Работник - количество кредитных договоров
26 (SELECT DISTINCT worker_id, COUNT(credit_cont_id) OVER(PARTITION BY worker_id)
27 FROM bank_schema.credit_contract)
28 ) counts
29 ) counts_max
30 )
```

Data Output

Explain

Messages

Notifications

worker_id	last_name	middle_name	first_name	phone_number	passport_type	passport_num
[PK] integer	character varying (30)	character varying (30)	character varying (30)	character varying (15)	character varying (20)	character varying (20)
1	5 Siggins	Sauveur	Donnamarie	563-429-1673	EU	2787726285

(рис 7 максимальное число кредитных и вкладовых договоров)

bank/postgres@PostgreSQL 13

Query Editor

Query History

1

2

SELECT

deposit_cont_id,

deposit_date,

return_date,

deposit_sum,

return_sum,

comment,

deposit_id,

depositor_id,

worker_id

FROM

bank_schema.deposit_contract;

Data Output

Explain

Messages

Notifications

	deposit_cont_id [PK] integer	deposit_date date	return_date date	deposit_sum money	return_sum money	comment character varying (150)	deposit_id integer	depositor_id integer	worker_id integer
1	5	2021-04-10	2021-06-23	116 346,00 ?	143 208,00 ?	[null]	1	10	1
2	7	2021-04-28	2021-06-05	117 739,00 ?	163 761,00 ?	[null]	4	5	5
3	10	2021-04-05	2021-06-04	108 321,00 ?	147 534,00 ?	[null]	2	3	4
4	11	2021-04-20	2021-06-25	100 000,00 ?	440 000,00 ?	[null]	4	4	5
5	2	2021-04-19	2021-06-15	107 771,00 ?	144 430,00 ?	[null]	1	4	3
6	6	2021-04-27	2021-06-15	105 991,00 ?	149 983,00 ?	[null]	3	6	1
7	8	2021-04-04	2021-06-16	104 927,00 ?	171 234,00 ?	[null]	8	5	5
8	1	2021-05-25	2021-06-05	119 132,00 ?	160 166,00 ?	[null]	5	4	3
9	4	2021-05-20	2021-06-10	117 397,00 ?	163 386,00 ?	[null]	4	1	5
10	9	2021-05-21	2021-06-13	106 469,00 ?	155 470,00 ?	[null]	7	5	5
11	3	2021-05-24	2021-06-15	119 166,00 ?	169 618,00 ?	[null]	3	6	1

(рис 8 исходные данные в договорах о депозитах)

bank/postgres@PostgreSQL 13											
Query Editor Query History											
<pre> 1 SELECT credit_cont_id, credit_date, repay_date, credit_sum, comment, credit_id, depositor_id, worker_id 2 FROM bank_schema.credit_contract;</pre>											
Data Output Explain Messages Notifications											
	credit_cont_id [PK] integer	credit_date date	repay_date date	credit_sum money	comment character varying (150)	credit_id integer	depositor_id integer	worker_id integer			
1		2021-04-21	2021-06-16	109 401,00 ?	[null]	10		1			4
2		2021-04-10	2021-06-19	104 932,00 ?	[null]	5		6			5
3		2021-04-07	2021-06-07	112 048,00 ?	[null]	3		6			1
4		2021-04-01	2021-06-29	103 325,00 ?	[null]	5		1			4
5		2021-04-12	2021-06-17	108 445,00 ?	[null]	3		9			5
6		2021-04-10	2021-06-27	124 369,00 ?	[null]	7		7			5
7		2021-04-15	2021-06-24	113 100,00 ?	[null]	3		4			3
8		2021-05-25	2021-06-04	100 235,00 ?	[null]	3		10			6
9		2021-05-21	2021-06-05	113 956,00 ?	[null]	3		7			1
10		2021-05-30	2021-06-29	120 077,00 ?	[null]	2		4			5

(рис 9 исходные данные в договорах о кредитах)

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

bank/postgres@PostgreSQL 13											
Query Editor Query History											
<pre> 1 SELECT first_name, last_name, dep.name, deposit_date, return_date, deposit_sum, return_sum, 2 (return_sum - deposit_sum) as "Could withdraw" 3 4 FROM bank_schema.deposit_contract depcont 5 JOIN bank_schema.depositor depositor ON depositor.depositor_id = depcont.depositor_id 6 JOIN bank_schema.deposit dep ON dep.deposit_id = depcont.deposit_id 7 WHERE return_date = (NOW() + INTERVAL '1 day')::date 8</pre>											
Data Output Explain Messages Notifications											
	first_name character varying (30)	last_name character varying (30)	name character varying (50)	deposit_date date	return_date date	deposit_sum money	return_sum money	Could withdraw money			
1	Gunilla	Glackin	Saritaea Dugand	2021-04-19	2021-06-15	107 771,00 ?	144 430,00 ?	36 659,00 ?			
2	Jenifer	Thebe	Calochortus argillosus	2021-04-24	2021-06-15	119 166,00 ?	169 618,00 ?	50 452,00 ?			
3	Jenifer	Thebe	Calochortus argillosus	2021-04-27	2021-06-15	105 991,00 ?	149 983,00 ?	43 992,00 ?			

(рис 10 вкладчики с начислениями)

g. Вывести список сотрудников, заключивших договоры на максимальную сумму за последний месяц

bank/postgres@PostgreSQL 13

Query Editor Query History

```

1 SELECT * FROM bank_schema.worker
2
3 WHERE worker_id IN (
4 -- Выбор работника с наибольшей суммой
5 SELECT worker_id FROM (
6 -- Выбор максимальной суммы вклада договора
7 SELECT DISTINCT worker_id, MAX(deposit_sum) OVER(PARTITION BY worker_id)
8 FROM bank_schema.deposit_contract
9 WHERE deposit_date > (NOW() - INTERVAL '1 month')::date
10 UNION
11 -- Выбор максимальной суммы кредитного договора
12 SELECT DISTINCT worker_id, MAX(credit_sum) OVER(PARTITION BY worker_id)
13 FROM bank_schema.credit_contract
14 WHERE credit_date > (NOW() - INTERVAL '1 month')::date
15 ) max_sum
16
17 WHERE max_sum.max IN (
18 -- Выбор числа максимальной суммы договоров
19 SELECT MAX(max_sum.max) FROM (
20 -- Выбор максимальной суммы вклада договора
21 SELECT DISTINCT worker_id, MAX(deposit_sum) OVER(PARTITION BY worker_id)
22 FROM bank_schema.deposit_contract
23 WHERE deposit_date > (NOW() - INTERVAL '1 month')::date
24 UNION
25 -- Выбор максимальной суммы кредитного договора
26 SELECT DISTINCT worker_id, MAX(credit_sum) OVER(PARTITION BY worker_id)
27 FROM bank_schema.credit_contract
28 WHERE credit_date > (NOW() - INTERVAL '1 month')::date
29 ) max_sum
30 )
31

```

Data Output Explain Messages Notifications

	worker_id [PK] integer	last_name character varying (30)	middle_name character varying (30)	first_name character varying (30)	phone_number character varying (15)	passport_type character varying (20)	passport_num character varying (20)
1	5	Siggins	Sauveur	Donnamarie	563-429-1673	EU	2787726285

(рис 11 максимальная сумма договора)

bank/postgres@PostgreSQL 13

Query Editor Query History

```

1 SELECT deposit_cont_id, deposit_date, return_date, deposit_sum, return_sum, comment, deposit_id, depositor_id, worker_id
2 FROM bank_schema.deposit_contract;

```

Data Output Explain Messages Notifications

	deposit_cont_id [PK] integer	deposit_date date	return_date date	deposit_sum money	return_sum money	comment character varying (150)	deposit_id integer	depositor_id integer	worker_id integer
1	5	2021-04-10	2021-06-23	116 346,00 ?	143 208,00 ?	[null]	1	10	1
2	7	2021-04-28	2021-06-05	117 739,00 ?	163 761,00 ?	[null]	4	5	5
3	10	2021-04-05	2021-06-04	108 321,00 ?	147 534,00 ?	[null]	2	3	4
4	11	2021-04-20	2021-06-25	100 000,00 ?	440 000,00 ?	[null]	4	4	5
5	2	2021-04-19	2021-06-15	107 771,00 ?	144 430,00 ?	[null]	1	4	3
6	6	2021-04-27	2021-06-15	105 991,00 ?	149 983,00 ?	[null]	3	6	1
7	8	2021-04-04	2021-06-16	104 927,00 ?	171 234,00 ?	[null]	8	5	5
8	1	2021-05-25	2021-06-05	119 132,00 ?	160 166,00 ?	[null]	5	4	3
9	4	2021-05-20	2021-06-10	117 397,00 ?	163 386,00 ?	[null]	4	1	5
10	9	2021-05-21	2021-06-13	106 469,00 ?	155 470,00 ?	[null]	7	5	5
11	3	2021-05-24	2021-06-15	119 166,00 ?	169 618,00 ?	[null]	3	6	1

(рис 12 исходные данные в договорах о депозитах)

bank/postgres@PostgreSQL 13										
Query Editor Query History										
<pre> 1 SELECT credit_cont_id, credit_date, repay_date, credit_sum, comment, credit_id, depositor_id, worker_id 2 FROM bank_schema.credit_contract;</pre>										
Data Output Explain Messages Notifications										
	credit_cont_id [PK] integer	credit_date date	repay_date date	credit_sum money	comment character varying (150)	credit_id integer	depositor_id integer	worker_id integer		
1		1 2021-04-21	2021-06-16	109 401,00 ?	[null]	10		1		4
2		2 2021-04-10	2021-06-19	104 932,00 ?	[null]	5		6		5
3		3 2021-04-07	2021-06-07	112 048,00 ?	[null]	3		6		1
4		4 2021-04-01	2021-06-29	103 325,00 ?	[null]	5		1		4
5		7 2021-04-12	2021-06-17	108 445,00 ?	[null]	3		9		5
6		9 2021-04-10	2021-06-27	124 369,00 ?	[null]	7		7		5
7		10 2021-04-15	2021-06-24	113 100,00 ?	[null]	3		4		3
8		5 2021-05-25	2021-06-04	100 235,00 ?	[null]	3		10		6
9		6 2021-05-21	2021-06-05	113 956,00 ?	[null]	3		7		1
10		8 2021-05-30	2021-06-29	120 077,00 ?	[null]	2		4		5

(рис 13 исходные данные в договорах о кредитах)

2. Представления

- а. Содержащее сведения обо всех сотрудниках банка и заключенных ими договорах по кредитам за прошедший месяц

```

1 CREATE VIEW bank_schema.bank_workers_contracts AS
2
3 SELECT worker.last_name AS "Фамилия работника",
4 worker.first_name AS "Имя работника",
5 CONCAT('Депозит') AS "Тип услуги",
6 deposit_cont_id AS "Контракт ID",
7 comment AS "Комментарий к контракту",
8 deposit.name AS "Название услуги",
9 deposit_sum AS "Сумма",
10 deposit_date AS "Дата начала",
11 return_date AS "Дата конца"
12 FROM bank_schema.deposit_contract depcont
13 JOIN bank_schema.worker worker ON worker.worker_id = depcont.worker_id
14 JOIN bank_schema.deposit deposit ON deposit.deposit_id = depcont.deposit_id
15 WHERE deposit_date > (NOW() - INTERVAL '1 month')::date
16
17 UNION
18
19 SELECT worker.last_name AS "Фамилия работника",
20 worker.first_name AS "Имя работника",
21 CONCAT('Кредит') AS "Тип услуги",
22 credit_cont_id AS "Контракт ID",
23 comment AS "Комментарий к контракту",
24 credit.name AS "Название услуги",
25 credit_sum AS "Сумма",
26 credit_date AS "Дата начала",
27 repay_date AS "Дата конца"
28 FROM bank_schema.credit_contract credcont
29 JOIN bank_schema.worker worker ON worker.worker_id = credcont.worker_id
30 JOIN bank_schema.credit credit ON credit.credit_id = credcont.credit_id
31 WHERE credit_date > (NOW() - INTERVAL '1 month')::date
32
33 UNION
34 -- Работники подписавшие ни одного договора в этом месяце
35 SELECT worker.last_name AS "Фамилия работника",
36 worker.first_name AS "Имя работника",
37 CAST(NULL AS TEXT) AS "Тип услуги",
38 CAST(NULL AS INT) AS "Контракт ID",
39 CAST(NULL AS TEXT) AS "Комментарий к контракту",
40 CAST(NULL AS TEXT) AS "Название услуги",
41 CAST(NULL AS MONEY) AS "Сумма",
42 CAST(NULL AS DATE) AS "Дата начала",
43 CAST(NULL AS DATE) AS "Дата конца"
44 FROM bank_schema.worker worker
45 WHERE (worker_id NOT IN (SELECT worker_id FROM bank_schema.deposit_contract WHERE deposit_date > (NOW() - INTERVAL '1 month')::date))
46 (worker_id NOT IN (SELECT worker_id FROM bank_schema.credit_contract WHERE credit_date > (NOW() - INTERVAL '1 month')::date))
47

```

(рис 14 создание представления о сотрудниках)

bank/postgres@PostgreSQL 13

Query Editor Query History

```

1 SELECT "Фамилия работника", "Имя работника", "Тип услуги", "Контракт ID", "Комментарий к контракту", "Название услуги", "Сумма", '
2 FROM bank_schema.bank_workers_contracts;

```

	Фамилия работника character varying (30)	Имя работника character varying (30)	Тип услуги text	Контракт ID integer	Комментарий к контракту character varying (255)	Название услуги character varying (100)	Сумма money	Дата начала date	Дата конца date
1	Beggini	Beggini	[null]	[null]	[null]	[null]	[null]	[null]	[null]
2	Cockroft	Jules	Кредит	5	[null]	Trifolium dubium Sibth.	100 235,00 ?	2021-05-25	2021-06-04
3	Dyka	Dyka	[null]	[null]	[null]	[null]	[null]	[null]	[null]
4	Howell	Ransell	Депозит	1	[null]	Phacelia ramosissima Douglas	119 132,00 ?	2021-05-25	2021-06-05
5	Rubberts	Vilma	Депозит	3	[null]	Calochortus argillosus	119 166,00 ?	2021-05-24	2021-06-15
6	Rubberts	Vilma	Кредит	6	[null]	Trifolium dubium Sibth.	113 956,00 ?	2021-05-21	2021-06-05
7	Siggins	Donnamarie	Депозит	4	[null]	Smelowskia calycina (Stephan...	117 397,00 ?	2021-05-20	2021-06-10
8	Siggins	Donnamarie	Депозит	9	[null]	Kleinhovia L.	106 469,00 ?	2021-05-21	2021-06-13
9	Siggins	Donnamarie	Кредит	8	[null]	Strophanthus kombe Oliv.	120 077,00 ?	2021-05-30	2021-06-29

(рис 15 использование представления о сотрудниках)

б. Найти клиентов банка, имеющих задолженности по кредитам.

```

1 CREATE VIEW bank_schema.overdue AS
2
3 SELECT credcont.credit_cont_id, depositor.depositor_id,
4 depositor.last_name, depositor.first_name,
5 credit.credit_id, credit.name,
6 credit_date, repay_date,
7 credit_sum,
8 (credit_sum*(interest+1)) "Sum with interest",
9 COALESCE(payment.sum, CAST(0 AS MONEY)) "Paid",
10 (credit_sum*(interest+1))-COALESCE(payment.sum, CAST(0 AS MONEY)) "Left to pay"
11
12 FROM bank_schema.credit_contract credcont
13 JOIN bank_schema.credit credit ON credit.credit_id = credcont.credit_id
14 JOIN bank_schema.depositor depositor ON depositor.depositor_id = credcont.depositor_id
15 LEFT JOIN (SELECT DISTINCT credit_cont_id, SUM(sum) OVER(PARTITION BY credit_cont_id) FROM bank_schema.payment pay) payment
16 ON payment.credit_cont_id = credcont.credit_cont_id
17 -- Выбрать те договоры, которые полностью оплачены
18 WHERE credcont.credit_cont_id IN (SELECT pay_table.credit_cont_id FROM (
19 -- Из списка договоров
20 SELECT credit_cont_id, credit_sum*(interest+1) FROM bank_schema.credit_contract credcont
21 JOIN bank_schema.credit credit ON credit.credit_id = credcont.credit_id
22 EXCEPT
23 -- И списка оплаты каждого из них
24 SELECT DISTINCT credit_cont_id, SUM(sum) OVER(PARTITION BY credit_cont_id) FROM bank_schema.payment pay
25 ) pay_table
26 )
27 AND (repay_date < NOW())
28 ORDER BY depositor_id

```

(рис 16 создание представления должников)

bank/postgres@PostgreSQL 13												
Query Editor Query History												
<pre> 1 SELECT credit_cont_id, depositor_id, last_name, first_name, credit_id, name, credit_date, repay_date, credit_sum, 2 "Sum with interest", "Paid", "Left to pay" 3 FROM bank_schema.overdue; </pre>												
Data Output Explain Messages Notifications												
	credit_cont_id	depositor_id	last_name	first_name	credit_id	name	credit_date	repay_date	credit_sum	Sum with interest	Paid	Left to pay
	integer	integer	character va	character va	integer	character varying (50)	date	date	money	money	money	money
1	3	6	Thebe	Jenifer	3	Trifolium dubium Sibh.	2021-04-07	2021-06-07	112 048,00 ?	128 855,20 ?	100 000,00 ?	28 855,20 ?
2	6	7	Tomaello	Ninetta	3	Trifolium dubium Sibh.	2021-05-21	2021-06-05	113 956,00 ?	131 049,40 ?	25 000,00 ?	106 049,40 ?
3	5	10	Edney	Boothe	3	Trifolium dubium Sibh.	2021-05-25	2021-06-04	100 235,00 ?	115 270,25 ?	0,00 ?	115 270,25 ?

(рис 17 использование представления должников)

bank/postgres@PostgreSQL 13 ▾

Query Editor Query History

1

2

```
SELECT payment_id, sum, pay_date, credit_cont_id
FROM bank_schema.payment;
```

Data Output Explain Messages Notifications

	<div>payment_id</div> <div>[PK] integer</div>	<div>sum</div> <div>money</div>	<div>pay_date</div> <div>date</div>	<div>credit_cont_id</div> <div>integer</div>	
1	1	50 000,00 ?	2021-05-01	1	
2	2	50 000,00 ?	2021-06-01	1	
3	3	9 401,00 ?	2021-06-14	1	
4	4	10 940,10 ?	2021-06-15	1	
5	5	100 000,0...	2021-06-01	3	
6	6	25 000,00 ?	2021-06-05	6	

(рис 18 исходные данные об оплате)

3. Запросы на модификацию данных

а. INSERT

bank/postgres@PostgreSQL 13

Query Editor Query History

```

1 INSERT INTO bank_schema.payment(
2 payment_id, sum, pay_date, credit_cont_id)
3 VALUES (
4 (SELECT MAX(payment_id)+1 FROM bank_schema.payment),
5 (SELECT "Left to pay" FROM bank_schema.overdue WHERE depositor_id = 6 LIMIT 1),
6 NOW()::date,
7 (SELECT credit_cont_id FROM bank_schema.overdue WHERE depositor_id = 6 LIMIT 1)
8 )

```

Data Output Explain Messages Notifications

INSERT 0 1

Query returned successfully in 48 msec.

(рис 19 погашение долга депозитчика с определенным идентификатором)

bank/postgres@PostgreSQL 13

Query Editor Query History

```

1 SELECT credit_cont_id, depositor_id, last_name, first_name, name, credit_date, repay_date, "Sum with interest", "Left to pay"
2 FROM bank_schema.overdue;

```

Data Output Explain Messages Notifications

	credit_cont_id integer	depositor_id integer	last_name character varying (50)	first_name character varying (50)	name character varying (50)	credit_date date	repay_date date	Sum with interest money	Left to pay money
1		3	Thebe	Jenifer	Trifolium dubium Sibth.	2021-04-07	2021-06-07	128 855,20 ?	28 855,20 ?
2		6	Tomaello	Ninetta	Trifolium dubium Sibth.	2021-05-21	2021-06-05	131 049,40 ?	106 049,40 ?
3		5	Edney	Boothe	Trifolium dubium Sibth.	2021-05-25	2021-06-04	115 270,25 ?	115 270,25 ?

(рис 20 до погашения)

bank/postgres@PostgreSQL 13

Query Editor Query History

```

1 SELECT credit_cont_id, depositor_id, last_name, first_name, name, credit_date, repay_date, "Sum with interest", "Left to pay"
2 FROM bank_schema.overdue;

```

Data Output Explain Messages Notifications

	credit_cont_id integer	depositor_id integer	last_name character varying (50)	first_name character varying (50)	name character varying (50)	credit_date date	repay_date date	Sum with interest money	Left to pay money
1		6	Tomaello	Ninetta	Trifolium dubium Sibth.	2021-05-21	2021-06-05	131 049,40 ?	106 049,40 ?
2		5	Edney	Boothe	Trifolium dubium Sibth.	2021-05-25	2021-06-04	115 270,25 ?	115 270,25 ?

(рис 21 после погашения)

b. UPDATE

bank/postgres@PostgreSQL 13 ▾

Query Editor Query History

```

1 UPDATE bank_schema.worker
2 SET position_id = CASE WHEN position_id > 1 THEN position_id-1 ELSE 1 END
3 WHERE worker_id IN
4 (SELECT worker_id FROM bank_schema.credit_contract
5 WHERE credit_date > (NOW() - INTERVAL '6 month')::date
6 GROUP BY worker_id
7 HAVING COUNT(credit_cont_id) > 3)

```

Data Output Explain Messages Notifications

UPDATE 1

Query returned successfully in 43 msec.

(рис 22 повышение сотрудников заключивших больше 3-х кредитных договоров за последние 6 месяцев на одну должность)

bank/postgres@PostgreSQL 13 ▾

Query Editor Query History

```

1 SELECT worker_id, last_name, middle_name, first_name, phone_number, address, position_id
2 FROM bank_schema.worker;

```

Data Output Explain Messages Notifications

	worker_id [PK] integer	last_name character varying (30)	middle_name character varying (30)	first_name character varying (30)	phone_number character varying (15)	address character varying (150)	position_id integer
1	1	Rubberts	Ginni	Vilma	554-310-5157	23 Marquette Way	3
2	2	Dyka	Aura	Tanny	985-601-7178	9 Walton Way	3
3	3	Howell	Cissy	Ransell	449-159-0949	9674 Caliangt Trail	1
4	4	Beggini	Melloney	Melvin	759-658-7228	96133 Fair Oaks Plaza	3
5	5	Siggins	Sauveur	Donnamarie	563-429-1673	046 Ridgeway Junction	3
6	6	Cockroft	Vinnie	Jules	501-150-6596	61388 Redwing Trail	2

(рис 23 до повышения)

bank/postgres@PostgreSQL 13 ▾							
Query Editor Query History							
<pre> 1 SELECT worker_id, last_name, middle_name, first_name, phone_number, address, position_id 2 FROM bank_schema.worker;</pre>							
Data Output Explain Messages Notifications							
	worker_id [PK] integer	last_name character varying (30)	middle_name character varying (30)	first_name character varying (30)	phone_number character varying (15)	address character varying (150)	position_id integer
1	1	Rubberts	Ginni	Vilma	554-310-5157	23 Marquette Way	3
2	2	Dyka	Aura	Tanny	985-601-7178	9 Walton Way	3
3	3	Howell	Cissy	Ransell	449-159-0949	9674 Caliangt Trail	1
4	4	Beggini	Melloney	Melvin	759-658-7228	96133 Fair Oaks Plaza	3
5	6	Cockroft	Vinnie	Jules	501-150-6596	61388 Redwing Trail	2
6	5	Siggins	Sauveur	Donnamarie	563-429-1673	046 Ridgeway Junction	2

(рис 24 после повышения)

с. DELETE

bank/postgres@PostgreSQL 13 ▾			
Query Editor Query History			
<pre> 1 DELETE FROM bank_schema.payment 2 WHERE credit_cont_id IN (3 SELECT DISTINCT credcont.credit_cont_id FROM bank_schema.payment payment 4 JOIN bank_schema.credit_contract credcont ON credcont.credit_cont_id = payment.credit_cont_id 5 WHERE depositor_id = 6 6)</pre>			
Data Output Explain Messages Notifications			
DELETE 2			
Query returned successfully in 51 msec.			

(рис 25 удаление информации о выплатах депозитчика с определенным идентификатором)

bank/postgres@PostgreSQL 13 ▾					
Query Editor Query History					
<pre> 1 SELECT payment_id, sum, pay_date, credit_cont_id 2 FROM bank_schema.payment;</pre>					
Data Output Explain Messages Notifications					
	payment_id [PK] integer	sum money	pay_date date	credit_cont_id integer	
1	1	50 000,00 ?	2021-05-01	1	
2	2	50 000,00 ?	2021-06-01	1	
3	3	9 401,00 ?	2021-06-14	1	
4	4	10 940,10 ?	2021-06-15	1	
5	5	100 000,0...	2021-06-01	3	
6	6	25 000,00 ?	2021-06-05	6	
7	7	28 855,20 ?	2021-06-14	3	

(рис 26 до удаления выплат)

bank/postgres@PostgreSQL 13 ▾					
Query Editor Query History					
<pre> 1 SELECT payment_id, sum, pay_date, credit_cont_id 2 FROM bank_schema.payment;</pre>					
Data Output Explain Messages Notifications					
	payment_id [PK] integer	sum money	pay_date date	credit_cont_id integer	
1	1	50 000,00 ?	2021-05-01	1	
2	2	50 000,00 ?	2021-06-01	1	
3	3	9 401,00 ?	2021-06-14	1	
4	4	10 940,10 ?	2021-06-15	1	
5	6	25 000,00 ?	2021-06-05	6	

(рис 27 после удаления выплат)

5. Выводы

В результате проделанной работы, были закреплены знания о основных командах в SQL, получена практика создания подзапросов и команд на модификацию данных.