Вариант №30

БПМ 211

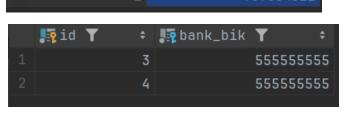
Ткаченко Никита

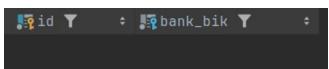
```
CREATE TABLE banks (
bik NUMERIC(9,0) PRIMARY KEY,
name VARCHAR(255) NOT NULL,
legal_address VARCHAR(255) NOT NULL,
correspondent_account VARCHAR(20) NOT NULL
Таблица банкоматов
CREATE TABLE atms (
 id SERIAL PRIMARY KEY,
 address VARCHAR(255) NOT NULL,
 bank bik INT NOT NULL,
 FOREIGN KEY (bank_bik) REFERENCES banks(bik)
Таблица клиентов
карточки
- (и нет полей, прописанных в условии, ибо возникает противоречие, а хранить массивы это
CREATE TABLE clients (
 id SERIAL PRIMARY KEY,
 full_name VARCHAR(100) NOT NULL,
 address VARCHAR(255) NOT NULL,
 bank bik NUMERIC(9,0) NOT NULL,
 FOREIGN KEY (bank_bik) REFERENCES banks(bik)
CREATE TABLE card_types (
 name VARCHAR(50) PRIMARY KEY
- Таблица валют
CREATE TABLE currencies (
 code VARCHAR(3) PRIMARY KEY,
 name VARCHAR(50) NOT NULL UNIQUE
CREATE TABLE exchange_rates (
 id SERIAL PRIMARY KEY,
 currency VARCHAR(3) NOT NULL,
 rate_to_rub NUMERIC(15, 6) NOT NULL,
 FOREIGN KEY (currency) REFERENCES currencies(code)
 Таблица карт клиентов
```

```
CREATE TABLE cards (
 id SERIAL PRIMARY KEY,
 number VARCHAR(16) NOT NULL UNIQUE,
 card_type VARCHAR(50) NOT NULL,
 currency VARCHAR(3) NOT NULL,
 client_id INT NOT NULL,
 FOREIGN KEY (card_type) REFERENCES card_types(name),
 FOREIGN KEY (currency) REFERENCES currencies(code),
 FOREIGN KEY (client_id) REFERENCES clients(id)
CREATE TABLE transactions (
 card_id INT NOT NULL,
 atm id INT NOT NULL,
 commission BOOLEAN NOT NULL,
 amount NUMERIC(15, 2) NOT NULL,
 amount_in_rub DECIMAL(15, 2) NOT NULL,
 FOREIGN KEY (card_id) REFERENCES cards(id),
 FOREIGN KEY (atm_id) REFERENCES atms(id)
INSERT INTO card_types (name) VALUES ('private'), ('business');
INSERT INTO currencies (code, name) VALUES ('RUB', 'Российский рубль'), ('USD', 'Доллар США'),
```

1) Получить список банкоматов заданного банка:







2) Получить выписку об операциях получения наличных заданным клиентом (ФИО) за заданный период в табличной форме:

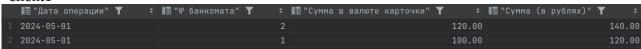
```
SELECT transactions.datetime::date AS "Дата операции",
    transactions.atm_id AS "№ банкомата",
    transactions.amount AS "Сумма в валюте карточки",
    transactions.amount_in_rub AS "Сумма (в рублях)"

FROM transactions
    JOIN cards ON transactions.card_id = cards.id
    JOIN clients ON cards.client_id = clients.id

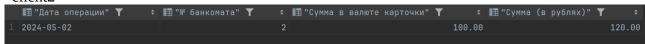
WHERE clients.full_name = 'client1'

AND transactions.datetime BETWEEN '2024-05-01' AND '2024-05-30';
```

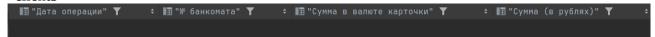
"client1"



"client2"

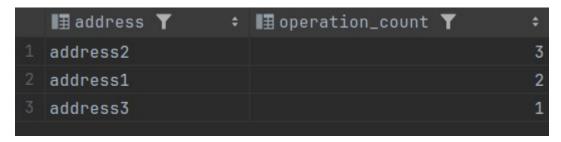


"client3"



3) Получить ТОП-3 адресов наиболее посещаемых банкоматов (критерий – количество операций):

```
SELECT atms.address, COUNT(transactions.id) AS operation_count
FROM transactions
JOIN atms ON transactions.atm_id = atms.id
GROUP BY atms.id
ORDER BY operation_count DESC
LIMIT 3;
```

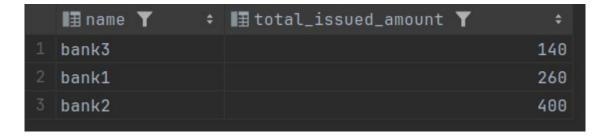


4) Получить список банков с наименее доходными банкоматами. Критерий доходности банка – объем выданных денег:

```
SELECT banks.name, SUM(transactions.amount_in_rub) AS total_issued_amount
FROM transactions

JOIN atms ON transactions.atm_id = atms.id

JOIN banks ON atms.bank_bik = banks.bik
GROUP BY banks.name, banks.bik
ORDER BY total_issued_amount;
```



5) Получить список клиентов заданного банка, наиболее часто обналичивающих деньги в «чужих» банкоматах за заданный период времени:

```
SELECT clients.full_name, COUNT(transactions.id) AS transaction_count
FROM transactions

JOIN cards ON transactions.card_id = cards.id

JOIN clients ON cards.client_id = clients.id

JOIN atms ON transactions.atm_id = atms.id

WHERE clients.bank_bik = 123456789

AND atms.bank_bik <> 123456789

AND transactions.datetime BETWEEN '2024-05-01' AND '2024-05-30'

GROUP BY clients.full_name

ORDER BY transaction count DESC;
```

123456789



6) Сформировать отчет «Сведения об операциях выдачи наличных денег клиентам банкоматами заданного банка с взиманием комиссионных вознаграждений за заданный период. Отчет представить в табличной форме:

