

Лабораторна робота 7

«Table production»

Зверніть увагу, що в рамках цієї лабораторної роботи ми НЕ використовуємо агрегатні та віконні (аналітичні ($f()$ over (...)) функції, групування (GROUP BY), Join. Підзапити – тільки у тому випадку, коли виконання завдання без них неможливе.

Завдання лабораторної роботи виконую всі

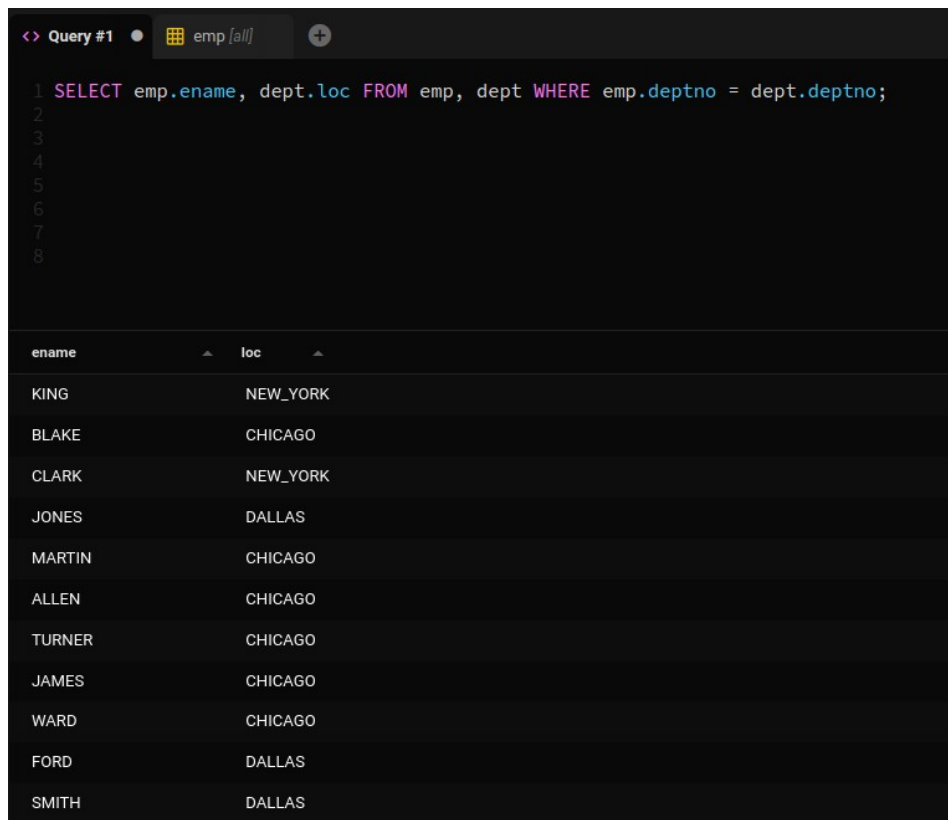
Для завдань/розділів відмічених «📸» – скріншот – обов'язково. Для інших – за домовленістю із викладачем.

Основне завдання

Екві-з'єднання

1. 📸 Створіть звіт який містить ім'я співробітника та місто, в якому він працює.

```
SELECT emp.ename, dept.loc FROM emp, dept WHERE emp.deptno = dept.deptno;
```



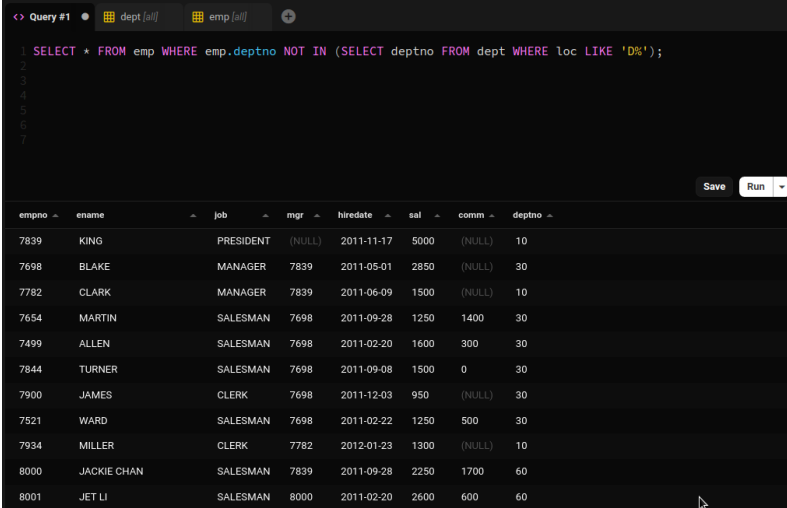
ename	loc
KING	NEW_YORK
BLAKE	CHICAGO
CLARK	NEW_YORK
JONES	DALLAS
MARTIN	CHICAGO
ALLEN	CHICAGO
TURNER	CHICAGO
JAMES	CHICAGO
WARD	CHICAGO
FORD	DALLAS
SMITH	DALLAS

2. Створіть звіт який містить ім'я та зарплату для всіх співробітників з Dallas.

```
SELECT emp.ename, emp.sal FROM emp WHERE emp.deptno IN (SELECT deptno FROM dept WHERE loc = 'DALLAS');
```

3. Створіть звіт який містить всіх співробітників, місто яких починається не з "D" (Dallas)

```
SELECT * FROM emp WHERE emp.deptno NOT IN (SELECT deptno FROM dept WHERE loc LIKE 'D%');
```



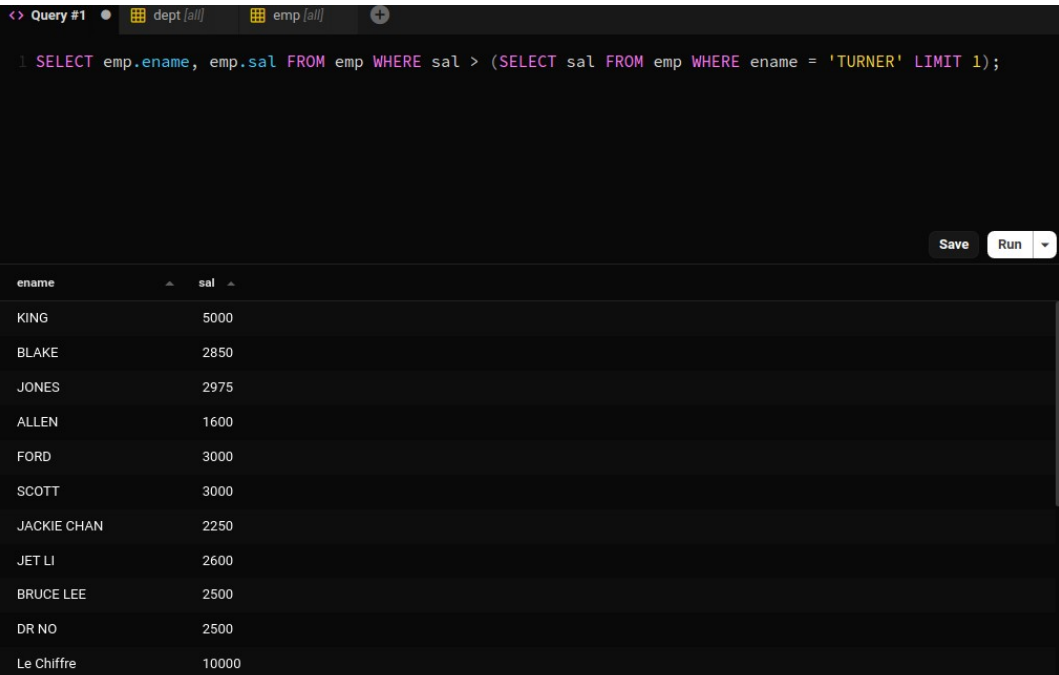
The screenshot shows a SQL query editor with the following query: `SELECT * FROM emp WHERE emp.deptno NOT IN (SELECT deptno FROM dept WHERE loc LIKE 'D%');`. The results are displayed in a table with columns: empno, ename, job, mgr, hiredate, sal, comm, deptno.

empno	ename	job	mgr	hiredate	sal	comm	deptno
7839	KING	PRESIDENT	(NULL)	2011-11-17	5000	(NULL)	10
7698	BLAKE	MANAGER	7839	2011-05-01	2850	(NULL)	30
7782	CLARK	MANAGER	7839	2011-06-09	1500	(NULL)	10
7654	MARTIN	SALESMAN	7698	2011-09-28	1250	1400	30
7499	ALLEN	SALESMAN	7698	2011-02-20	1600	300	30
7844	TURNER	SALESMAN	7698	2011-09-08	1500	0	30
7900	JAMES	CLERK	7698	2011-12-03	950	(NULL)	30
7521	WARD	SALESMAN	7698	2011-02-22	1250	500	30
7934	MILLER	CLERK	7782	2012-01-23	1300	(NULL)	10
8000	JACKIE CHAN	SALESMAN	7839	2011-09-28	2250	1700	60
8001	JET LI	SALESMAN	8000	2011-02-20	2600	600	60

Не-екви з'єднання

4. Створіть звіт який виведе імена та зарплату всіх службовців із зарплатою вище ніж у Turner.

```
SELECT emp.ename, emp.sal FROM emp WHERE sal > (SELECT sal FROM emp WHERE ename = 'TURNER' LIMIT 1);
```



The screenshot shows a SQL query editor with the following query: `SELECT emp.ename, emp.sal FROM emp WHERE sal > (SELECT sal FROM emp WHERE ename = 'TURNER' LIMIT 1);`. The results are displayed in a table with columns: ename, sal.

ename	sal
KING	5000
BLAKE	2850
JONES	2975
ALLEN	1600
FORD	3000
SCOTT	3000
JACKIE CHAN	2250
JET LI	2600
BRUCE LEE	2500
DR NO	2500
Le Chiffre	10000

Тренувальні вправи

5. Виведіть перелік відділів, в яких є співробітники.

```
SELECT DISTINCT dept.* FROM dept, emp WHERE emp.deptno = dept.deptno;
```

6. Створіть звіт який виведе всіх керівників.

```
SELECT DISTINCT mgr.* FROM emp worker, emp mgr WHERE mgr.empno =  
worker.mgr;
```

7. Виведіть співробітників, у яких керівник працює у іншому місті.

```
SELECT worker.* FROM emp worker, emp mgr  
WHERE mgr.empno = worker.mgr  
AND NOT  
      (SELECT loc FROM dept WHERE deptno = mgr.deptno)  
      =  
      (SELECT loc FROM dept WHERE deptno = worker.deptno)  
;
```

8. Виведете список співробітників, які є начальниками, але працюють не на посаді «MANAGER»

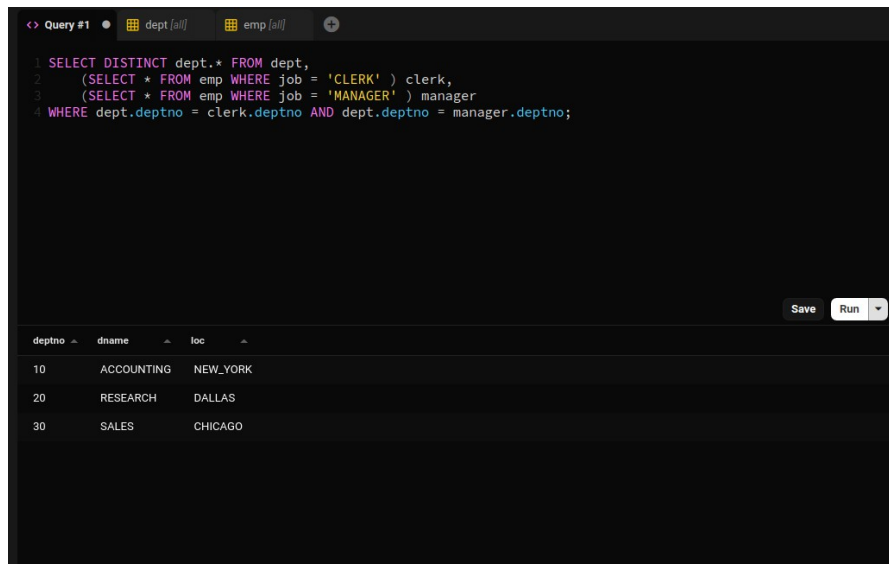
```
SELECT DISTINCT mgr.* FROM emp worker, emp mgr  
WHERE mgr.empno = worker.mgr AND NOT mgr.job = 'MANAGER';
```

9. Виведіть співробітників, у яких дохід (зарплата + премія) більша за дохід їх керівників.

```
SELECT DISTINCT worker.* FROM emp worker, emp mgr  
WHERE worker.mgr = mgr.empno  
AND (worker.sal + COALESCE(worker.comm, 0)) >  
(mgr.sal + COALESCE(mgr.comm, 0));
```

10. 📷 (⚠ Зверніть увагу на це завдання) Виведіть відділ де є менеджер та клерк (принаймні по одна особа – клерк та принаймні одна – менеджер).

```
SELECT DISTINCT dept.* FROM dept,  
    (SELECT * FROM emp WHERE job = 'CLERK' ) clerk,  
    (SELECT * FROM emp WHERE job = 'MANAGER' ) manager  
WHERE dept.deptno = clerk.deptno AND dept.deptno = manager.deptno;
```



11. * Виведіть міста де є менеджер та клерк.

```
SELECT DISTINCT dept.loc "City" FROM dept,  
    (SELECT * FROM emp WHERE job = 'CLERK' ) clerk,  
    (SELECT * FROM emp WHERE job = 'MANAGER' ) manager  
WHERE dept.deptno = clerk.deptno AND dept.deptno = manager.deptno;
```

12. ** Виведіть міста де є керівник-менеджер та керівник не-менеджер.
(«менеджер» - посада, «керівник» - є підлеглі).

```
SELECT DISTINCT dept.loc "City" FROM dept,  
    (SELECT DISTINCT mangr.* FROM emp worker, emp mangr  
    WHERE worker.mgr = mangr.empno AND NOT mangr.job = 'MANAGER')  
mangr,  
    (SELECT DISTINCT mangr.* FROM emp worker, emp mangr  
    WHERE worker.mgr = mangr.empno AND mangr.job = 'MANAGER') manager  
WHERE dept.deptno = mangr.deptno AND dept.deptno = manager.deptno;
```

13. * Які посади у начальників MANAGER'ів ?


```
SELECT DISTINCT mgr.job FROM emp mgr, (SELECT * FROM emp WHERE job =  
'MANAGER') manager  
WHERE mgr.empno = manager.mgr;
```

14. * Виведіть список підрозділів, в яких працює хоча б один начальник

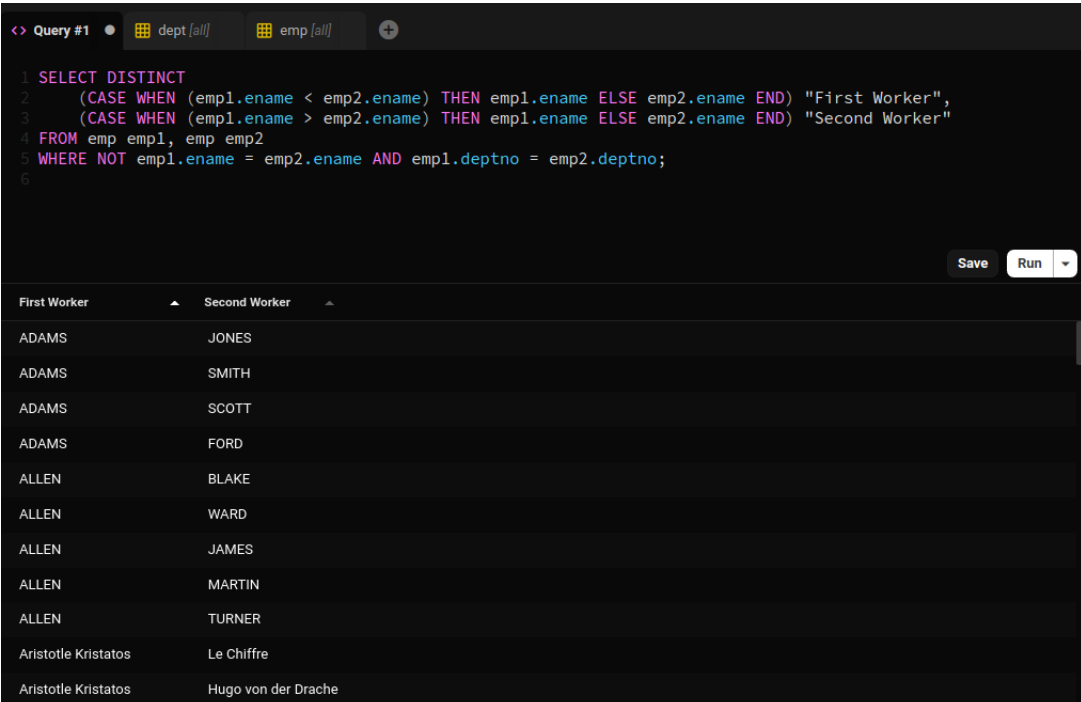
```
SELECT DISTINCT dept.* FROM dept, (  
    SELECT mgr.* FROM emp worker, emp mgr WHERE worker.mgr = mgr.empno  
) manager  
WHERE dept.deptno = manager.deptno;
```

15. * Виведіть список міст, яких працює хоча б один начальник

```
SELECT DISTINCT dept.loc "City" FROM dept, (  
    SELECT mgr.* FROM emp worker, emp mgr WHERE worker.mgr = mgr.empno  
) manager  
WHERE dept.deptno = manager.deptno;
```

16. *  Створіть запит, який виведе всі пари імен співробітників, які працюють в одному відділі. При цьому не включати до списку комбінації імена співробітників самих із собою (тобто комбінацію типу «Іванів-Іванів») та комбінації імен співробітників, що відрізняються порядком прямування (тобто включати одну з двох комбінацій типу «Іванів-Петрів» та «Петрів-Іванів»).

```
SELECT DISTINCT  
    (CASE WHEN (emp1.ename < emp2.ename) THEN emp1.ename ELSE  
emp2.ename END) "First Worker",  
    (CASE WHEN (emp1.ename > emp2.ename) THEN emp1.ename ELSE  
emp2.ename END) "Second Worker"  
FROM emp emp1, emp emp2  
WHERE NOT emp1.ename = emp2.ename AND emp1.deptno = emp2.deptno;
```



The screenshot shows a SQL query editor with the following query:

```
1 SELECT DISTINCT  
2     (CASE WHEN (emp1.ename < emp2.ename) THEN emp1.ename ELSE emp2.ename END) "First Worker",  
3     (CASE WHEN (emp1.ename > emp2.ename) THEN emp1.ename ELSE emp2.ename END) "Second Worker"  
4 FROM emp emp1, emp emp2  
5 WHERE NOT emp1.ename = emp2.ename AND emp1.deptno = emp2.deptno;  
6
```

The results are displayed in a table with two columns: "First Worker" and "Second Worker".

First Worker	Second Worker
ADAMS	JONES
ADAMS	SMITH
ADAMS	SCOTT
ADAMS	FORD
ALLEN	BLAKE
ALLEN	WARD
ALLEN	JAMES
ALLEN	MARTIN
ALLEN	TURNER
Aristotle Kristatos	Le Chiffre
Aristotle Kristatos	Hugo von der Drache

17. ** Модифікуйте запит 12 так, щоб пари будувалися для співробітників, що працюють в одному місті, а не в одному відділі. А заробітна плата відрізняється не більш як на 25%.

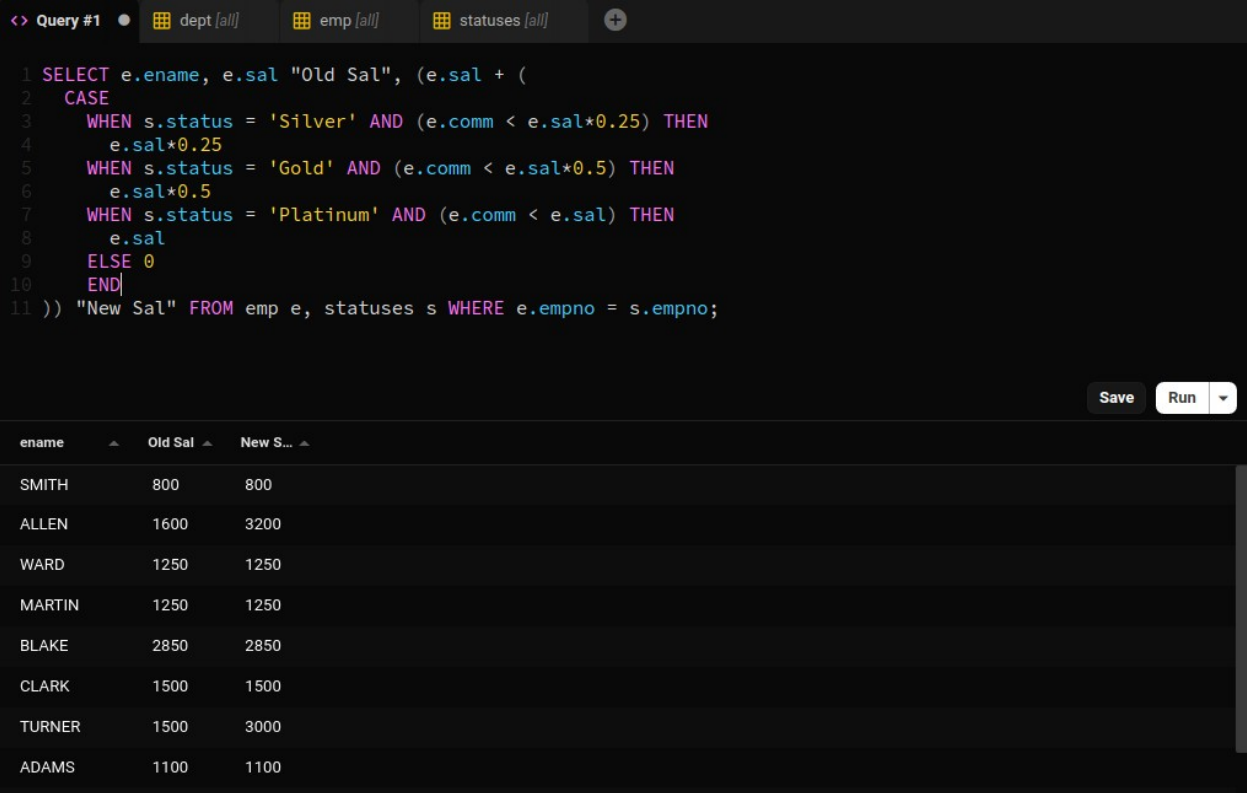
```
SELECT DISTINCT dept.loc "City" FROM dept,  
  (SELECT DISTINCT mangr.* FROM emp worker, emp mangr  
    WHERE worker.mgr = mangr.empno AND NOT mangr.job = 'MANAGER')  
mgr,  
  (SELECT DISTINCT mangr.* FROM emp worker, emp mangr  
    WHERE worker.mgr = mangr.empno AND mangr.job = 'MANAGER') manager  
WHERE  
  dept.loc IN (SELECT loc FROM dept WHERE deptno = mgr.deptno)  
  AND  
  dept.loc IN (SELECT loc FROM dept WHERE deptno = manager.deptno)  
  AND mgr.sal BETWEEN manager.sal*0.75 AND manager.sal*1.25;
```

18. 📺 У лабораторній роботі DML було завдання «Статуси» (Срібний-Золотий-Платиновий)

* Для співробітників, у яких встановлено статус, виведіть ім'я співробітника та його статус.

** Для працівників, у яких встановлено статус, виведіть ім'я співробітника, його стару премію та нову премію відповідно до його статусу (пам'ятаємо, що премії знижувати не можна)

```
SELECT e.ename, e.sal "Old Sal", (e.sal + (  
  CASE  
    WHEN s.status = 'Silver' AND (e.comm < e.sal*0.25) THEN  
      e.sal*0.25  
    WHEN s.status = 'Gold' AND (e.comm < e.sal*0.5) THEN  
      e.sal*0.5  
    WHEN s.status = 'Platinum' AND (e.comm < e.sal) THEN  
      e.sal  
    ELSE 0  
  END  
)) "New Sal" FROM emp e, statuses s WHERE e.empno = s.empno;
```



Query #1

```
1 SELECT e.ename, e.sal "Old Sal", (e.sal + (  
2   CASE  
3     WHEN s.status = 'Silver' AND (e.comm < e.sal*0.25) THEN  
4       e.sal*0.25  
5     WHEN s.status = 'Gold' AND (e.comm < e.sal*0.5) THEN  
6       e.sal*0.5  
7     WHEN s.status = 'Platinum' AND (e.comm < e.sal) THEN  
8       e.sal  
9     ELSE 0  
10    END  
11 )) "New Sal" FROM emp e, statuses s WHERE e.empno = s.empno;
```

Save Run

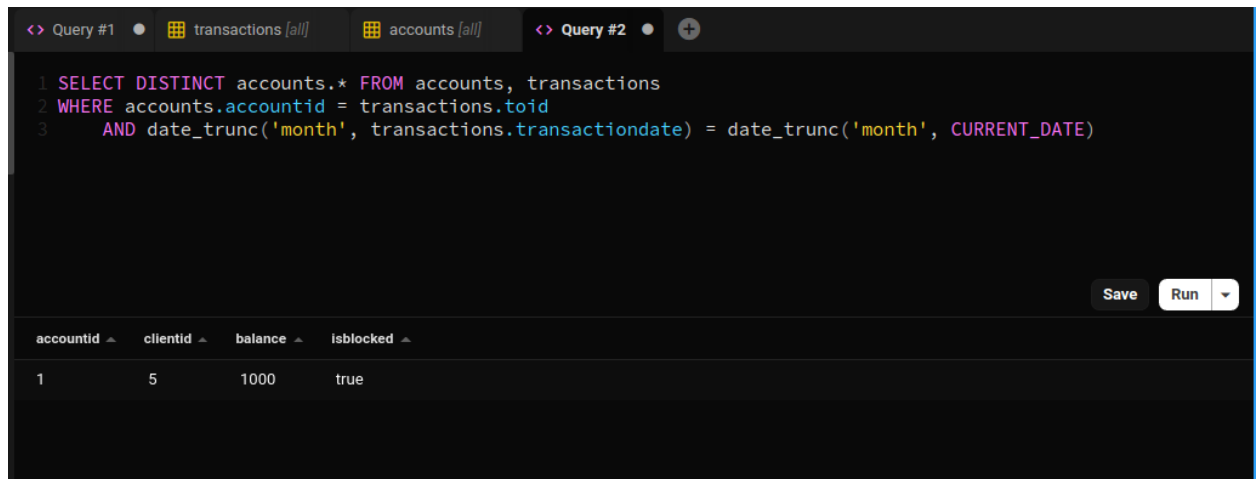
ename	Old Sal	New Sal
SMITH	800	800
ALLEN	1600	3200
WARD	1250	1250
MARTIN	1250	1250
BLAKE	2850	2850
CLARK	1500	1500
TURNER	1500	3000
ADAMS	1100	1100

Завдання за варіантами

Варіант 1 📄

1. Виведіть рахунки, на які надходили гроші цього місяця.

```
SELECT DISTINCT accounts.* FROM accounts, transactions
WHERE accounts.accountid = transactions.toid
      AND date_trunc('month', transactions.transactiondate) =
date_trunc('month', CURRENT_DATE)
```



```
<> Query #1 • transactions [all] accounts [all] <> Query #2 • +
```

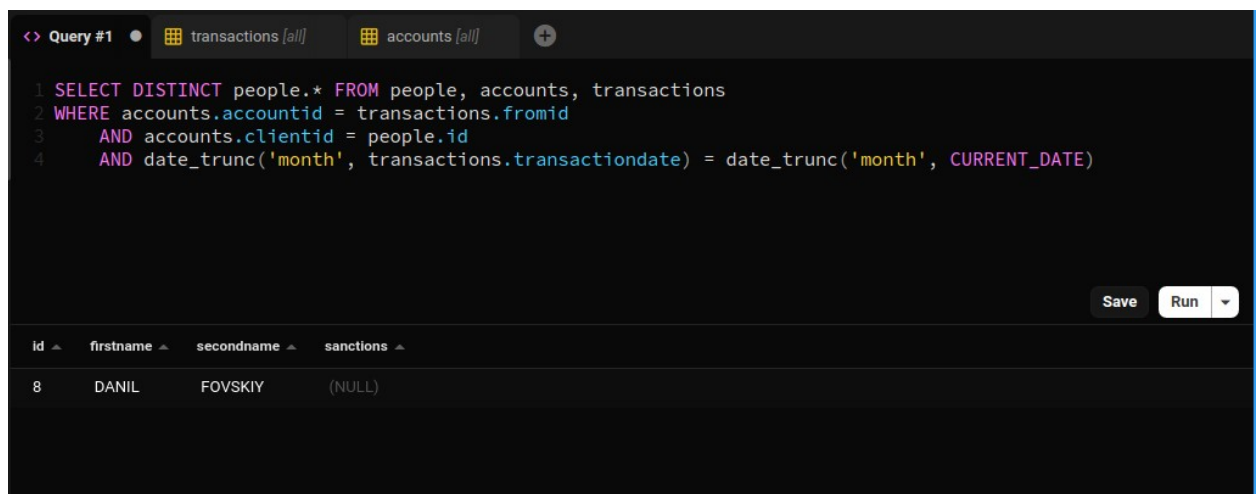
```
1 SELECT DISTINCT accounts.* FROM accounts, transactions
2 WHERE accounts.accountid = transactions.toid
3     AND date_trunc('month', transactions.transactiondate) = date_trunc('month', CURRENT_DATE)
```

Save Run

accountid	clientid	balance	isblocked
1	5	1000	true

2. Виведіть людей, які витрачали гроші цього місяця.

```
SELECT DISTINCT people.* FROM people, accounts, transactions
WHERE accounts.accountid = transactions.fromid
      AND accounts.clientid = people.id
      AND date_trunc('month', transactions.transactiondate) =
date_trunc('month', CURRENT_DATE)
```



```
<> Query #1 • transactions [all] accounts [all] +
```

```
1 SELECT DISTINCT people.* FROM people, accounts, transactions
2 WHERE accounts.accountid = transactions.fromid
3     AND accounts.clientid = people.id
4     AND date_trunc('month', transactions.transactiondate) = date_trunc('month', CURRENT_DATE)
```

Save Run

id	firstname	secondname	sanctions
8	DANIL	FOVSKIY	(NULL)

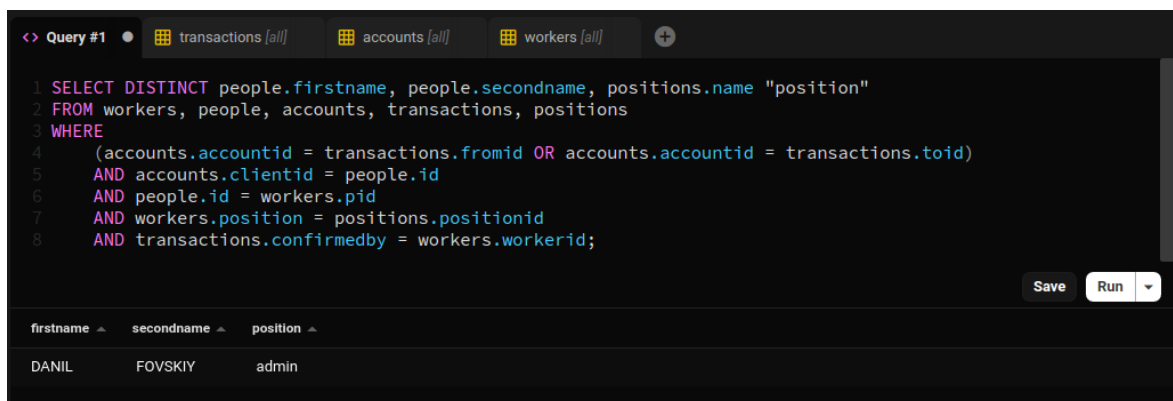
3. Уважно подивіться на Ваші таблиці «клієнт» та «співробітник». За умовою клієнт може бути співробітником. Реалізуйте запит, що знайде людей про яких в різних таблицях міститься різна інформація (наприклад: людина одна, а ім'я чи адреса не співпадає).

Посніть, якщо це неможливо у Вашій БД.

Неможливо, бо дані про клієнта/співробітника зберігаються лише в таблиці «люди». А таблиця співробітників містить лише id людини у відповідній таблиці.

4. * Співробітник може бути клієнтом. Операції по рахунку обов'язково перевіряються співробітником. Знайдіть співробітників, які перевіряли операції самі собі.

```
SELECT DISTINCT people.firstname, people.secondname, positions.name  
"position"  
FROM workers, people, accounts, transactions, positions  
WHERE  
    (accounts.accountid = transactions.fromid OR accounts.accountid =  
transactions.toid)  
    AND accounts.clientid = people.id  
    AND people.id = workers.pid  
    AND workers.position = positions.positionid  
    AND transactions.confirmedby = workers.workerid;
```



The screenshot shows a database query editor with the following SQL query:

```
1 SELECT DISTINCT people.firstname, people.secondname, positions.name "position"  
2 FROM workers, people, accounts, transactions, positions  
3 WHERE  
4     (accounts.accountid = transactions.fromid OR accounts.accountid = transactions.toid)  
5     AND accounts.clientid = people.id  
6     AND people.id = workers.pid  
7     AND workers.position = positions.positionid  
8     AND transactions.confirmedby = workers.workerid;
```

The results table shows the following data:

transactionid	fromid	toid	amount	confirmed	confirmedby	transactiondate
1	1	2	500	false	(NULL)	2010-12-11 00:00:00
2	2	4	1500	false	(NULL)	2011-12-11 00:00:00
5	4	1	100	false	(NULL)	2011-12-11 00:00:00
4	3	1	10	false	(NULL)	2022-04-11 00:00:00
3	5	2	300	false	(NULL)	2022-05-11 00:00:00
6	4	1	100	true	5	2022-11-09 00:00:00