



Бази даних та інформаційні системи

Лекція 9. Вкладені запити

СумДУ, каф.КН
2020

Задачи заняття

Після завершення заняття ви будете вміти і знати наступне:

- ▶ Типи задач, які можуть бути вирішені за допомогою вкладених запитів;
- ▶ Визначення вкладеного запиту;
- ▶ Список типів вкладених запитів;
- ▶ Як писати вкладені запити, що повертають один рядок, багато рядків, багато стовпців.



Використання вкладеного запиту для рішення задачі

“У кого зарплата більше, ніж у Джона?”

Основний запит:



“Хто з працівників має зарплату більше, ніж Джон?”

Вкладений запит:



“А яка зарплата у Джона?”



Підзапити

```
SELECT    select_list
FROM      table
WHERE     expr operator
          (SELECT    select_list
           FROM      table);
```

- ▶ Підзапити (вкладені запити) виконуються один раз перед виконанням основного запиту.
- ▶ Результат виконання підзапиту використовується в основному запиті (зовнішньому запиті).




Підзапити

- ▶ Підзапити можна використовувати в:
 - ▶ Операторі SELECT:
 - ▶ SELECT
 - ▶ FROM
 - ▶ WHERE
 - ▶ HAVING
 - ▶ CREATE VIEW
 - ▶ CREATE TABLE
 - ▶



Використання підзапитів

```
SQL> SELECT  ename
      2  FROM    emp
      3  WHERE   sal >
      4          (SELECT sal
      5                FROM    emp
      6                WHERE   ename=' JONES ' ) ;
```



ENAME

KING

FORD

SCOTT

Створення таблиці з використанням підзапиту

```
SQL> CREATE TABLE dept30
  2  AS
  3      SELECT * FROM DEPT;
Table created.
```

```
SQL> DESCRIBE dept30
```

Name	Null?	Type
DEPTNO		NUMBER (2)
DNAME		VARCHAR2 (14)
LOC		VARCHAR2 (13)

Створення представлення (подання)

```
SQL> CREATE VIEW salvu30
  2 AS SELECT empno EMP_NUMBER, ename NAME, sal SALARY
  3 WHERE depno = 10
  4 FROM emp;
View created.
```

```
SQL> SELECT *
  2 FROM salvu30;
```

EMP_NUMBER	NAME	SALARY
7698	BLAKE	2850
7654	MARTIN	1250
7499	ALLEN	1600
7844	TURNER	1500
...		

Поради щодо використання підзапитів

- ▶ Укладайте підзапити в дужки.
- ▶ Розташовуйте підзапити в правій частині операторів порівняння.
- ▶ Не використовуйте умову сортування **ORDER BY** в підзапитах.
- ▶ Використовуйте «однорядкові» оператори для результатів підзапитів, які повертають один рядок.
- ▶ Використовуйте «багаторядкові» оператори для результатів підзапитів, які повертають набір рядків.



Види підзапитів

- ▶ Підзапит, що повертає один рядок



- ▶ Підзапит, що повертає багато рядків



- ▶ Підзапит, що повертає багато стовпців



Підзапит, що повертає один рядок

- ▶ Використовуйте «однорядкові» оператори

Оператор	Значення
=	Дорівнює
>	Більше
>=	Більше або дорівнює
<	Менше
<=	Менше або дорівнює
<>	Не дорівнює



Виконання запитів, що повертають один рядок

```
SQL> SELECT      ename, job
  2  FROM        emp
  3  WHERE       job =
  4              (SELECT      job
  5                  FROM      emp
  6                  WHERE      empno = 7369)
  7  AND         sal >
  8              (SELECT      sal
  9                  FROM      emp
 10                  WHERE      empno = 7876) ;
```

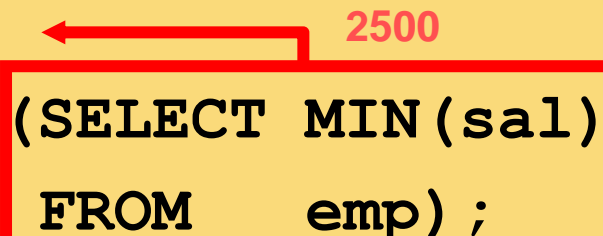
CLERK

1100

ENAME	JOB
-----	-----
MILLER	CLERK

Використання групування в підзапитах

```
SELECT ename, job
FROM emp
WHERE sal =
```



2500

```
(SELECT MIN(sal)
FROM emp);
```

ENAME	JOB
-----	-----
MILLER	CLERK

Чи буде виконано цей вираз?

```
SQL> SELECT  ename, job
      2  FROM    emp
      3  WHERE   job =
      4          (SELECT  job
      5                  FROM    emp
      6                  WHERE   ename='no one employer') ;
```

no rows selected

Підзапит не повертає значень



Підзапити, що повертають багато рядків

- ▶ Повертають більше одного рядка
- ▶ Використовуйте «багаторядкові» оператори порівняння

Оператор	Значення
IN	Збігається з будь-яким зі списку
ANY <i>SOME</i>	Порівнює значення з усіма значеннями в сенсі ІСНУЄ
ALL	Порівнює значення з усіма значеннями в сенсі ДЛЯ ВСІХ



Використання оператора ANY в підзапитах, що повертають багато рядків


```
SQL> SELECT empno, ename, job 1300
      2 FROM emp 1100
      3 WHERE sal < ANY 800
      4 (SELECT sal 950
      5 FROM emp
      6 WHERE job = 'CLERK')
      7 AND job <> 'CLERK';
```

EMPNO	ENAME	JOB
7654	MARTIN	SALESMAN
7521	WARD	SALESMAN

Використання оператора ALL в підзапитах, що повертають багато рядків

```
SELECT empno, ename, job, sal
FROM emp
WHERE sal < ALL (SELECT sal
                  FROM empl
                  WHERE job = 'CLERK')
AND job <> 'CLERK';
```

850, 900, 1100, 1300



Якщо клерки отримують найнижчу зарплату в компанії, то:

```
no rows selected
```



Значення NULL в підзапитах

Відобразити всіх працівників, які не мають підлеглих

```
SQL> SELECT  employee.ename
2  FROM      emp employee
3  WHERE     employee.empno NOT IN
4
5            (SELECT manager.mgr
              FROM      emp manager) ;
```

no rows selected.

Одне з повернутих
значень - NULL

Еквівалентні

NOT IN	!=ALL
IN	=ANY

Підзапити, що повертають багато стовпців

Парні порівняння

- ▶ Виводить ім'я, номер підрозділу, зарплату і комісійні будь-якого працівника, у якого **обидва** значення **зарплати** і **комісійних** збігаються зі значеннями **зарплати** і **комісійних** будь-якого працівника в підрозділі 30.

Непарні порівняння

- ▶ Виводить ім'я, номер підрозділу, зарплату і комісійні будь-якого працівника, **зарплата** і **комісійні** якого збігаються з **зарплатою** і **комісійними** будь-якого працівника в підрозділі 30.



Порівняння стовпців

Парне

SAL	COMM
1600	300
1250	500
1250	1400
2850	
1500	0
950	

Непарне

SAL	COMM
1600	300
1250	500
1250	1400
2850	
1500	0
950	

Підзапит з парним порівнянням

- Виводить ім'я, номер підрозділу, зарплату і комісійні будь-якого працівника, у якого **обидва** значення зарплати і комісійних збігаються зі значеннями зарплати і комісійних будь-якого працівника в підрозділі 30.

```
SQL> SELECT  ename, deptno, sal, comm
2  FROM      emp
3  WHERE      (sal, NVL(comm,-1)) IN
4              (SELECT sal, NVL(comm,-1)
5                  FROM      emp
6                  WHERE      deptno = 30) ;
```

ENAME	DEPNO	SAL	COMM
CLARK	10	1500	300
...			
ALLEN	30	1600	300
TURNER	30	1500	0
MISTER X	30	1500	300

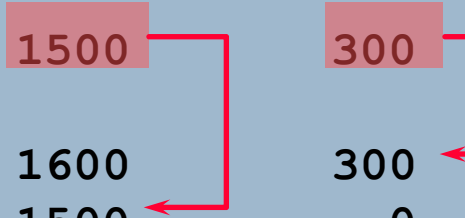


Підзапит з непарним порівнянням

- Виводить ім'я, номер підрозділу, зарплату і комісійні будь-якого працівника, зарплата якого збігається з зарплатою будь-якого працівника в підрозділі 30 і комісійні збігаються з комісійними будь-якого працівника в підрозділі 30.

```
SQL> SELECT  ename, deptno, sal, comm
2  FROM      emp
3  WHERE      sal IN (SELECT sal FROM emp
4                WHERE deptno = 30)
5  AND        NVL(comm, -1) IN (SELECT NVL(comm, -1)
6                                FROM emp
7                                WHERE deptno = 30);
```

ENAME	DEPNO	SAL	COMM
CLARK	10	1500	300
...			
ALLEN	30	1600	300
TURNER	30	1500	0
...			



Парні і непарні порівняння у запити

```
SQL> SELECT  ename, deptno, sal, comm
  2  FROM      emp
  3  WHERE      (sal, NVL(comm,-1)) IN
  4
  5              (SELECT sal, NVL(comm,-1)
  6              FROM      emp
              WHERE      deptno = 30) ;
```

```
SQL> SELECT  ename, deptno, sal, comm
  2  FROM      emp
  3  WHERE      sal IN
  4
  5              (SELECT sal
  6              FROM      emp
  7              WHERE      deptno = 30)
  8  AND
  9              NVL(comm,-1) IN
  10             (SELECT NVL(comm,-1)
  11             FROM      emp
  12             WHERE      deptno = 30) ;
```

Підзапит з парним порівнянням

```
SQL> SELECT  ename, deptno, sal, comm
  2  FROM      emp
  3  WHERE      (sal, NVL(comm,-1)) IN
  4              (SELECT sal, NVL(comm,-1)
  5                  FROM      emp
  6                  WHERE      deptno = 30) ;
```

ENAME	DEPTNO	SAL	COMM
JAMES	30	950	
WARD	30	1250	500
MARTIN	30	1250	1400
TURNER	30	1500	0
ALLEN	30	1600	300
BLAKE	30	2850	

6 rows selected.

Підзапит з непарним порівнянням

```
SQL> SELECT  ename, deptno, sal, comm
2  FROM      emp
3  WHERE      sal IN          (SELECT sal
4                               FROM    emp
5                               WHERE    deptno = 30)
6  AND
7            NVL(comm, -1) IN (SELECT NVL(comm, -1)
8                               FROM    emp
9                               WHERE    deptno = 30);
```

ENAME	DEPTNO	SAL	COMM
JAMES	30	950	
BLAKE	30	2850	300
TURNER	30	1500	0
CLARK	10	1500	300
...			
7 rows selected.			

Підсумки

- ▶ Підзапити корисні, якщо запит ґрунтується на невідомих даних

```
SELECT    select_list
FROM      table
WHERE     expr operator
          (SELECT select_list
           FROM    table);
```



Зв'язані підзапити

- Корельовані (зв'язані) підзапити виконуються для кожного рядка вхідних даних



Зв'язані підзапити

- ▶ Зв'язані підзапити використовують значення з основного запиту

```
SELECT    select_list
FROM      table
WHERE     expr operator
          (SELECT select_list
           FROM    table);
```

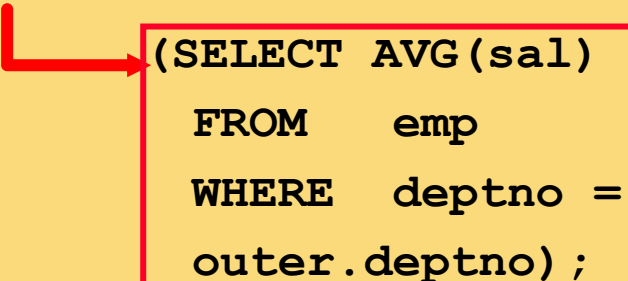


```
SELECT column1, column2, ...
FROM   table1 outer
WHERE  column1 operator
          (SELECT  column2
           FROM    table2
           WHERE    expr1 =
                   outer.expr2) ;
```

Використання зв'язаних підзапитів

- ▶ Знайдіть співробітників, які отримують більше, ніж середня зарплата по їх відділу

```
SELECT ename, salary, deptno
FROM emp outer
WHERE sal >
      (SELECT AVG(sal)
       FROM emp
       WHERE deptno =
         outer.deptno);
```



Підзапит виконається для
кожного співробітника

EXISTS

Знайти співробітників, які є керівниками хоча-б одного підлеглого

```
SELECT empno, job, deptno
FROM   emp tbl1
WHERE  EXISTS ( SELECT 'X'
                  FROM   emp
                  WHERE  mng =
                        tbl1.empno) ;
```

- Умову **EXISTS** (або **NOT EXISTS**) використовують із зв'язаними підзапитами;
- Якщо підзапит повертає хоча б один запис, то умова **EXISTS** поверне **True**, інакше – **False**;
- Внутрішній запит не потребує повернення певного значення, тому:
 - можна використовувати константу;
 - результат запиту може містити довільну кількість стовпців;

NOT EXISTS

Знайти співробітників, які не є керівниками

```
SELECT empno, job, deptno
FROM   emp tbl1
WHERE  NOT EXISTS ( SELECT 'X'
                    FROM   emp
                    WHERE  mgr = tbl1.empno );
```

Згадаємо приклад з NOT IN :

```
SQL> SELECT   employee.ename
      2 FROM    emp employee
      3 WHERE   employee.empno NOT IN
      4         (SELECT manager.mgr
      5         FROM   emp manager);
no rows selected.
```

Знайдіть відділи, в яких немає робітників

```
SELECT deptno, dname
FROM dept d
WHERE NOT EXISTS (SELECT 'X'
                  FROM emp
                  WHERE deptno = d.deptno);
```



Зв'язані або не зв'язані запити

- ▶ Вивести імена співробітників, які працюють в Нью-Йорку

```
SQL>SELECT ename FROM emp
2      WHERE depno IN (
3          SELECT depno FROM dept
4          WHERE loc='NEW_YORK' )
```

```
SQL>SELECT ename FROM emp e
2      WHERE EXIST (
3          SELECT depno FROM dept d
4          WHERE loc='NEW_YORK'
5          AND e.depno = d.depno)
```

Використання підзапиту в FROM

```
SQL> SELECT ename, sal  
2  FROM (SELECT * FROM EMP ORDER BY sal)  
3  WHERE rownum < 10;
```

```
SQL> SELECT ename, sal  
2  FROM (SELECT * FROM EMP WHERE depno = 20)  
3  WHERE sal > 1000
```

Зазвичай використовують підзапити, що повертають багато рядків і стовпців.



Використання підзапиту в SELECT

Підзапит має повертати одне значення - один рядок,
один стовпець!

```
SQL> SELECT  ename, (SELECT SYSDATE FROM DUAL)  
2          FROM emp;
```



Практичне використання підзапитів

Знайти 5 осіб з найвищою зарплатою

► Спроба №1 (Невдала)

```
1 SELECT e.ename, e.sal, rownum rn
2 FROM emp e
3 WHERE ROWNUM <= 5
4 ORDER BY sal DESC
```

ENAME	SAL	RN
-----	-----	-----
KING	5000	1
JONES	2975	4
BLAKE	2850	2
CLARK	1500	3
MARTIN	1250	5

Проблема: де працівники з зарплатою в 3000?

Знайти 5 осіб з найвищою зарплатою

► Спроба №2 (те, що треба!)

```
1  SELECT * from (  
2      SELECT e.ename, e.sal  
3      FROM emp e  
4      ORDER BY sal DESC  
5  )  
6  WHERE ROWNUM <= 5
```

ENAME	SAL
KING	5000
FORD	3000
SCOTT	3000
JONES	2975
BLAKE	2850

Розбиття на сторінки (pagination)



Перші 5:

```
select * from
  (select * from emp order by sal DESC)
where rownum <= 5
```

Решта (**не правильно**):

```
select * from
  (select * from emp order by sal DESC)
where rownum >= 5
```

Проблема:

No rows selected



Правильна пагінація

- ▶ <http://www.oracle.com/technetwork/issue-archive/2006/06-sep/o56asktom-086197.html>

```
1 Select  * from (
2   Select /*+ FIRST_ROWS(n) */ a.*, ROWNUM rnum from(
3     Select * from emp order by sal DESC
2   ) a
2   where ROWNUM <= :MAX_ROW_TO_FETCH )
1 where rnum  >= :MIN_ROW_TO_FETCH ;
```

- ▶ Вибрати з 20-го по 60-го:
 - ▶ MAX_ROW_TO_FETCH = 60
 - ▶ MIN_ROW_TO_FETCH = 20

