

Бази даних та Інформаційні системи

Лекція 7. Умови відбору даних при запиті до БД

СумДУ, каф. КН
2020
2013

Задачі заняття

Після завершення заняття ви повинні вміти та знати наступне:

- Як встановлювати обмеження на дані, що виводяться;
- Основні оператори і правила їх використання;
- Як сортувати дані вибірки.



Можливості оператора **SELECT**

Вибірка

Таблиця 1

Проекція

Таблиця 1

З'єднання

Таблиця 1



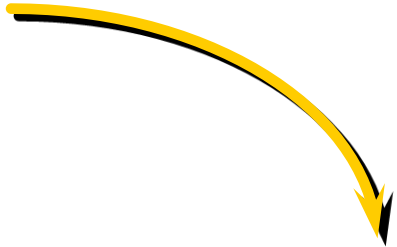
Таблиця 2

Обмеження рядків у виборці

EMP

EMPNO	ENAME	JOB	...	DEPTNO
7839	KING	PRESIDENT		10
7698	BLAKE	MANAGER		30
7782	CLARK	MANAGER		10
7566	JONES	MANAGER		20
...				

**"...вивести всіх
службовців
департаменту 10"**



EMP

EMPNO	ENAME	JOB	...	DEPTNO
7839	KING	PRESIDENT		10
7782	CLARK	MANAGER		10
7934	MILLER	CLERK		10

Обмеження рядків у виборці

Обмеження проводиться з використанням умови **WHERE**

▪

```
SELECT      [DISTINCT] {*, column [alias], ...}  
FROM        table  
[WHERE      condition(s)];
```

Умова **WHERE** пишеться після специфікатора **FROM**.



Використання умови **WHERE**

```
SQL> SELECT ename, job, deptno  
2 FROM emp  
3 WHERE job='CLERK';
```

ENAME	JOB	DEPTNO
-----	-----	-----
JAMES	CLERK	30
SMITH	CLERK	20
ADAMS	CLERK	20
MILLER	CLERK	10

Символьні рядки і дати

- ▶ Символьні рядки і дати вкладаються в одинарні лапки;
- ▶ Символьні значення чутливі до регістру, а дати чутливі до формату.
- ▶ За замовчуванням формат дати має вигляд **'DD.MM.YY'**

```
SQL> SELECT      ename, job, deptno  
      2  FROM      emp  
      3  WHERE     ename = ' JAMES ' ;
```



Оператори порівняння

Позначення	Призначення
=	дорівнює
>	Більше ніж
>=	Більше або дорівнює
<	Менше ніж
<=	Менше або дорівнює
<>, !=, ^=	Не дорівнює
is null	є NULL
is not null	Не є NULL

Try it!

```
select * from emp where ename ^= 'ADAMS';
```

```
select * from emp where ename <> 'Adams';
```

```
select * from emp where ename = "ADAMS";
```

```
select * from emp where ename = 'ADAMS';
```



Використання операторів порівняння

Порівняння стовпців:

```
SQL> SELECT ename, sal, comm  
2 FROM emp  
3 WHERE sal<=comm;
```

ENAME	SAL	COMM
MARTIN	1250	1400

Інші оператори порівняння

Позначення	Призначення
BETWEEN ... AND ...	Знаходиться між ...
LIKE	Схоже на шаблон
IN (...)	Належить списку



Використання оператора BETWEEN

Використовуйте умову **BETWEEN** для вибору значень, які знаходяться в межах діапазону.

```
SQL> SELECT      ename, sal
  2  FROM emp
  3  WHERE sal BETWEEN 1000 AND 1500 ;
```

ENAME	SAL
MARTIN	1250
TURNER	1500
WARD	1250
ADAMS	1100
MILLER	1300

Нижня
межа

Верхня
межа

Використання оператора IN

Використовуйте умову `IN` для перевірки входження значень у список.

```
expression [ NOT ] IN ( expression, [...]
```

```
SQL> SELECT empno, ename, sal, mgr
2  FROM emp
3  WHERE mgr IN (7902, 7566, 7788);
```

EMPNO	ENAME	SAL	MGR
7902	FORD	3000	7566
7369	SMITH	800	7902
7788	SCOTT	3000	7566
7876	ADAMS	1100	7788

Використання оператора **LIKE**

- ▶ Використовуйте оператор **LIKE** для пошуку за шаблоном.
- ▶ Умова пошуку може містити як символи, так і числа.

% - означає нуль або більше символів;

_ - означає один символ.

```
SQL> SELECT ename  
2 FROM emp  
3 WHERE ename LIKE 'S%';
```



Використання оператора **LIKE**

- Символи шаблону можна поєднувати.

```
SQL> SELECT ename  
2 FROM      emp  
3 WHERE     ename LIKE '_A%';
```

```
ENAME
```

```
-----
```

```
JAMES
```

```
WARD
```



Використання умови **LIKE**

- ▶ Ви можете використовувати ідентифікатор **ESCAPE** для пошуку символів "%" і "_".

```
... WHERE Columns LIKE '%/%' ESCAPE '/'
```

Ім'я «KING»

```
SELECT * FROM emp WHERE ename LIKE 'KI%'
```

```
SELECT * FROM emp WHERE ename LIKE 'KI%' ESCAPE '/'
```

```
SELECT * FROM emp WHERE ename LIKE 'KI%' ESCAPE 'I'
```

no rows selected

```
SELECT * FROM emp WHERE ename LIKE 'KII%' ESCAPE 'I'
```

Використання оператора IS NULL

Оператор IS NULL використовується для перевірки, чи є значення null.

```
SQL> SELECT  ename, mgr
      2  FROM    emp
      3  WHERE  mgr IS NULL;
```

ENAME	MGR
-----	-----
KING	

Try it!

~~select * from emp where comm = NULL;~~

select * from emp where comm is NULL;

~~select ename from emp where ename != ' ' ;~~

select ename from emp where ename is not NULL;



Логічні оператори

Оператор	Значення
AND	Повертає TRUE якщо <i>обидва</i> операнда мають значення TRUE
OR	Повертає TRUE якщо <i>будь-який</i> з операндів має значення TRUE
NOT	Повертає TRUE якщо операнд має значення FALSE



Використання оператора **AND**

AND потребує, щоб обидва операнда мали значення TRUE.

```
SQL> SELECT empno, ename, job, sal  
2 FROM emp  
3 WHERE sal >= 1100  
4 AND job = 'CLERK';
```

EMPNO	ENAME	JOB	SAL
7876	ADAMS	CLERK	1100
7934	MILLER	CLERK	1300

Використання оператора OR

OR потребує, щоб хоча б один з операндів мав значення TRUE.

```
SQL> SELECT empno, ename, job, sal
2 FROM emp
3 WHERE sal >= 1100
4 OR job = 'CLERK';
```

EMPNO	ENAME	JOB	SAL
7839	KING	PRESIDENT	5000
7698	BLAKE	MANAGER	2850
7782	CLARK	MANAGER	2450
7566	JONES	MANAGER	2975
7654	MARTIN	SALESMAN	1250
...			

14 rows selected.

Використання оператора NOT

```
SQL> SELECT ename, job
2 FROM emp
3 WHERE job NOT IN ('CLERK', 'MANAGER', 'ANALYST');
```

ENAME	JOB
KING	PRESIDENT
MARTIN	SALESMAN
ALLEN	SALESMAN
TURNER	SALESMAN
WARD	SALESMAN

```
SQL> SELECT ename, job
2 FROM emp
3 WHERE ename NOT LIKE 'A%';
```


AND, OR, NOT

AND	TRUE	FALSE	NULL
TRUE	TRUE	FALSE	NULL
FALSE	FALSE	FALSE	FALSE
NULL	NULL	FALSE	NULL

OR	TRUE	FALSE	NULL
TRUE	TRUE	TRUE	TRUE
FALSE	TRUE	FALSE	NULL
NULL	TRUE	NULL	NULL

NOT	TRUE	FALSE	NULL
	FALSE	TRUE	NULL



Пріоритети


Порядок виконання	
1	Арифметичні операції
2	Конкатенація ()
3	Порівняння(>, >=, =, ...)
4	IS [NOT] NULL, LIKE, [NOT] IN
5	[NOT] BETWEEN
6	!=
7	NOT
8	AND
9	OR

Послідовністю обчислення можна керувати за допомогою дужок.



Пріоритети

```
SQL> SELECT  ename, job, sal
2  FROM      emp
3  WHERE     job='SALESMAN'
4  OR        job='PRESIDENT'
5  AND       sal>1500;
```

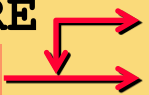


ENAME	JOB	SAL
KING	PRESIDENT	5000
MARTIN	SALESMAN	1250
ALLEN	SALESMAN	1600
TURNER	SALESMAN	1500
WARD	SALESMAN	1250

Пріоритети

- ▶ Використовуйте дужки для зазначення пріоритету обчислення.

```
SQL> SELECT      ename, job, sal
  2  FROM          emp
  3  WHERE         (job='SALESMAN'
  4  OR           job='PRESIDENT')
  5  AND          sal>1500;
```



ENAME	JOB	SAL
KING	PRESIDENT	5000
ALLEN	SALESMAN	1600

Умова сортування ORDER BY

- ▶ Умова ORDER BY сортує дані:
 - **ASC** – за зростанням (за замовчуванням);
 - **DESC** – за спаданням.
- ▶ Умова сортування ORDER BY - остання у виразі SELECT.

```
SQL> SELECT      ename, job, deptno, hiredate
2  FROM          emp
3  ORDER BY hiredate;
```

ENAME	JOB	DEPTNO	HIREDATE
SMITH	CLERK	20	17-DEC-80
ALLEN	SALESMAN	30	20-FEB-81
...			

14 rows selected.

Сортування за спаданням

```
SQL> SELECT          ename, job, deptno, hiredate  
  2  FROM            emp  
  3  ORDER BY hiredate DESC;
```

ENAME	JOB	DEPTNO	HIREDATE
ADAMS	CLERK	20	12-JAN-83
SCOTT	ANALYST	20	09-DEC-82
MILLER	CLERK	10	23-JAN-82
JAMES	CLERK	30	03-DEC-81
FORD	ANALYST	20	03-DEC-81
KING	PRESIDENT	10	17-NOV-81
MARTIN	SALESMAN	30	28-SEP-81
...			

14 rows selected.

Використання псевдоніма в умові сортування

```
SQL> SELECT      empno, ename, sal*12 annsal  
2 FROM          emp  
3 ORDER BY      annsal;
```

EMPNO	ENAME	ANNSAL
7369	SMITH	9600
7900	JAMES	11400
7876	ADAMS	13200
7654	MARTIN	15000
7521	WARD	15000
7934	MILLER	15600
7844	TURNER	18000

...
14 rows selected.

Сортування за кількома стовпцями

- ▶ Порядок сортування задається переліком в умові ORDER BY.
- ▶ Ви можете сортувати за стовпцем, який не входить до переліку вибірки оператора SELECT.

```
SQL> SELECT  ename, deptno, sal  
2  FROM      emp  
3  ORDER BY   deptno, sal DESC;
```

ENAME	DEPTNO	SAL
KING	10	5000
CLARK	10	2450
MILLER	10	1300
FORD	20	3000

...

14 rows selected.

Сортування

1

```
SQL> SELECT  ename, deptno, sal  
2  FROM      emp  
3  ORDER BY deptno ASC, sal DESC;
```

2

```
SQL> SELECT  ename, deptno, sal  
2  FROM      emp  
3  ORDER BY 12*sal;
```

3

```
SQL> SELECT  ename, deptno, sal annual_sal  
2  FROM      emp  
3  ORDER BY annual_sal;
```

4

```
SQL> SELECT  ename, deptno, sal annual_sal  
2  FROM      emp  
3  ORDER BY 1+1;
```

5

```
SQL> SELECT  ename, deptno, sal annual_sal  
2  FROM      emp  
3  ORDER BY 2;
```

Використання псевдонімів

Послідовність виконання запиту:

```
3 SELECT  ename, deptno, sal*1.2+1000 as new_sal
1 FROM    emp
2 WHERE   sal*1.2+1000 > 4500
4 ORDER BY new_sal DESC;
```

ENAME	DEPTNO	NEW_SAL
KING	10	7000
SCOTT	20	4600
FORD	20	4600
JONES	20	4570

Тому:

- В умові WHERE не можна використовувати псевдоніми;
- В сортуванні ORDER BY можна.

Питання

► Які з операторів можна використовувати в умові Where?

1. >=
2. IS NULL
3. !=
4. IS LIKE
5. IN BETWEEN
6. <>



Висновки

На сьогоднішньому занятті ми розглянули:

1. Використання умови `Where` для обмеження виведених даних:
 - порівняння значень;
 - використання `Between`, `In`, `Like`, `Is NULL`;
 - використання логічних операторів `AND`, `OR`, `NOT`;
2. Сортування результатів запиту:

```
SELECT      [DISTINCT] {*, column [alias], ...}  
FROM        table  
[WHERE      condition(s)]  
[ORDER BY   {column, expr, alias} [ASC|DESC]];
```

