



# Базы данных и Информационные системы

6/15 Условия отбора данных при запросе к БД

Кузиков Б.О.  
Сумы, СумГУ  
2013

## Задачи занятия

---

После завершения занятия вы должны уметь и знать следующее:

- Накладывать ограничения на выводимые данные
- Знать основные операторы и правила их применения
- Уметь сортировать выводимые данные



# Возможности выражения SELECT

---

**Выборка**


**Таблица 1**

**Проекция**


**Таблица 1**

**Соединение**


**Таблица 1**


**Таблица 2**



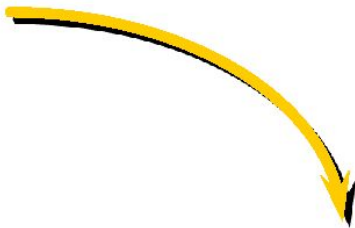
# Ограничение строк в выборке

---

## EMP

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

"...вывести всех служащих департамента 10"



## EMP

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





# Ограничение выбираемых строк

---

Ограничение производится с использованием предложения **WHERE**.

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

Предложение **WHERE** следует за предложением **FROM**.

---



# Использование предложения 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



# Символьные строки и даты

---

- ▶ Символьные строки и даты заключаются в одинарные кавычки
- ▶ Символьные значения чувствительны к регистру, а даты чувствительны к формату
- ▶ Для дат формат по умолчанию имеет вид 'DD-MON-YY'

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



# Операторы сравнения

Обозначение	Назначение
=	Равны
>	Больше
>=	Больше или равны
<	Меньше
<=	Меньше или равны
<>	Не равны
!=	
^=	
Is Null	Является NULL
Not is NULL	Не является NULL






# Использование операторов сравнения

---

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

ENAME	SAL	COMM
MARTIN	1250	1400



# Прочие операторы сравнения

---

Обозначение	Назначение
BETWEEN ... AND ...	Находится между
LIKE	Похоже на шаблон
IN (...)	Принадлежит списку



# Использование оператора BETWEEN

Используйте оператор BETWEEN для выбора значений, лежащих в диапазоне.

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

ENAME	SAL	Нижний предел	Верхний предел
MARTIN	1250		
TURNER	1500		
WARD	1250		
ADAMS	1100		
MILLER	1300		

# Использование оператора IN

---

Используйте оператор **IN** для проверки вхождения значения в список.

```
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



# Использование оператора LIKE

---

- Используйте оператор LIKE для поиска по шаблону.
- Условие поиска может содержать как символы, так и числа.
- % обозначает ноль или более символов
- \_ обозначает один символ

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





# Использование оператора LIKE

---

–Символы шаблона можно совмещать.

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

```
ENAME
```

```
-----
```

```
JAMES
```

```
WARD
```

– Вы можете использовать идентификатор ESCAPE для поиска символов "%" or "\_".

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

---



# Использование оператора IS NULL

---

Оператор IS NULL используется для проверки, совпадает ли значение с null

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

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



# Логические операторы

---

Оператор	Значение
<b>AND</b>	Возвращает TRUE если <i>оба</i> операнда имеют значение TRUE
<b>OR</b>	Возвращает TRUE если <i>любой</i> из операндов имеет значение TRUE
<b>NOT</b>	Возвращает TRUE если операнд имеет значение FALSE



# Использование оператора AND

---

- ▶ AND требует, чтобы оба операнда имели значение 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



# Использование оператора OR

OR требует, чтобы хотя бы один из операндов имел значение 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.



# Использование 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

# Приоритеты

---

Порядок выполнения	
1	Все операторы сравнения
2	NOT
3	AND
4	OR

Порядком вычисления можно управлять с помощью скобок.

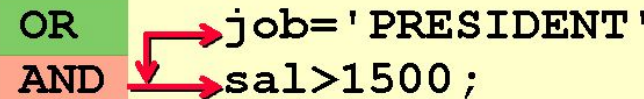
---



# Приоритеты

---

```
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



# Приоритеты

- ▶ Используйте скобки для указания приоритета вычисления.

```
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

# Предложение ORDER BY

---

- ▶ Предложение ORDER BY сортирует данные.
  - ASC: по возрастанию, по умолчанию
  - DESC: по убыванию
- ▶ Предложение ORDER BY идет последним в выражении 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.





# Сортировка по убыванию

---

```
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.



# Сортировка по псевдониму

---

```
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.

---

# Сортировка по нескольким столбцам

---

Порядок сортировки задается списком в предложении ORDER BY.

```
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.

- Вы можете сортировать по столбцу, не входящему в список выборки выражения SELECT.
- 



# Функция Rownum

---

```
SELECT *  
  FROM EMP  
 WHERE ROWNUM < 11
```



# Вопрос

---

▶ Какие из операторов, можно использовать в предложении Where?

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



# Выводы

---

- ▶ На сегодняшнем занятии мы рассмотрели:
  1. Использование предложения Where для ограничения выводимых данных:
    - ▶ Сравнение значений
    - ▶ Использование Between, In, Like, Is NULL
    - ▶ Использование логических операторов AND, OR, NOT
  2. Сортировку результатов запроса

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

