

Pertemuan 10

Logical Comparisons and Precedence Rules menggunakan Apex

Tujuan Intruksional	:
Pokok Bahasan ini mempelajari tentang <i>Database Programming</i> dengan SQL	
Kompetensi Yang Diharapkan	:
Mahasiswa diharapkan memahami tentang <i>Database Programming</i> termasuk cara instalasi dan penulisan <i>syntax SQL</i> menggunakan aplikasi <i>Oracle Application Express (APEX)</i> .	
Waktu Pertemuan	: 100 Menit

3-1 LOGICAL COMPARISONS AND PRECEDENCE RULES

Logical Operators: menggabungkan dua atau lebih kondisi untuk menghasilkan satu hasil.

Returned ONLY IF keseluruhan hasil dari kondisi ini benar.

AND : Returns TRUE jika kedua kondisi benar.

OR : Returns TRUE jika salah satu kondisi benar.

NOT : Returns TRUE jika kondisi salah.

Operator AND

Hasil yang dikembalikan akan menjadi baris yang memenuhi KEDUA kondisi yang ditentukan dalam klausa WHERE.

Contoh Syntax:

```
SELECT last_name, department_id, salary
FROM employees
WHERE department_id > 50 AND salary > 12000;
```

```
SELECT last_name, hire_date, job_id
FROM employees
WHERE hire_date > '01/jan/1998' AND job_id
      LIKE 'SA%';
```

Hasil :

LAST_NAME	DEPARTMENT_ID	SALARY
King	90	24000
Kochhar	90	17000
De Haan	90	17000

LAST_NAME	HIRE_DATE	JOB_ID
Zlotkey	29-Jan-2000	SA_MAN
Taylor	24-Mar-1998	SA_REP
Grant	24-May-1999	SA_REP

Operator OR

- Jika klausa WHERE menggunakan kondisi OR, hasil yang dikembalikan dari kueri akan menjadi baris yang memenuhi salah satu dari kondisi OR.
- Semua baris kembali memiliki location_id = 2500 OR mereka memiliki seorang manager_id = 124.

```
SELECT department_name, manager_id, location_id
FROM departments
WHERE location_id = 2500 OR manager_id=124;
```

Hasil:

DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
Shipping	124	1500
Sales	149	2500

Operator NOT

Mengembalikan baris yang TIDAK memenuhi syarat dalam klausa WHERE.

```
SELECT department_name, location_id
FROM departments
WHERE location_id NOT IN (1700,1800);
```

Hasil:

DEPARTMENT_NAME	LOCATION_ID
Shipping	1500
IT	1400
Sales	2500

Aturan yang diutamakan

🔑 Konsep penting:

ORDER	OPERATORS
1	Aritmatika + - * /
2	Concatenation
3	Comparison <, <=, >, >=, <>
4	IS (NOT) NULL, LIKE, (NOT) IN
5	(NOT) BETWEEN
6	NOT
7	AND
8	OR

AND – OR

Contoh:

```
SELECT last_name||' '||salary*1.05  
AS "Employee Raise", department_id, first_name  
FROM employees  
WHERE department_id IN(50,80)  
AND first_name LIKE 'C%'  
OR last_name LIKE '%s%';
```

1. Kondisi AND dievaluasi, sehingga semua karyawan yang bekerja di dept 80 atau 50, AND yang memiliki nama depan dimulai dengan huruf "C" dikembalikan.
2. Klausa OR kemudian dievaluasi dan mengembalikan employees dengan nama terakhir huruf "s".

Hasil:

Employee Raise	DEPARTMENT_ID	FIRST_NAME
Higgins 12600	110	Shelley
Mourgos 6090	50	Kevin
Rajs 3675	50	Trenna
Davies 3255	50	Curtis
Matos 2730	50	Randall
Vargas 2625	50	Peter
Ernst 6300	60	Bruce
Hartstein 13650	20	Michael

OR-AND

Contoh:

```
SELECT last_name||' '||salary*1.05  
AS "Employee Raise", department_id, first_name  
FROM employees  
WHERE department_id IN(50,80)  
OR first_name LIKE 'C%'  
AND last_name LIKE '%s%';
```

1. first_name dimulai dengan "C" AND last_name berisi "s". Kedua kondisi ini harus dipenuhi untuk dikembalikan.
2. Setiap employees di department 50 dan 80 akan dikembalikan.

Hasil:

Employee Raise	DEPARTMENT_ID	FIRST_NAME
Zlotkey 11025	80	Eleni
Abel 11550	80	Ellen
Taylor 9030	80	Jonathon
Mourgos 6090	50	Kevin
Rajs 3675	50	Trenna
Davies 3255	50	Curtis
Matos 2730	50	Randall
Vargas 2625	50	Peter

Menambahkan tanda kurung akan mengubah cara klausa WHERE dievaluasi, dan baris kembali.

```
SELECT last_name||' '||salary*1.05
AS "Employee Raise", department_id, first_name
FROM employees
WHERE (department_id IN(50,80)
OR first_name LIKE 'C%')
AND last_name LIKE '%s%';
```

1. Nilai dalam tanda kurung dipilih.
2. Semua nilai dalam tanda kurung yang juga mengandung huruf "s" pada last_name akan dikembalikan.

Hasil:

Employee Raise	DEPARTMENT_ID	FIRST_NAME
Mourgos 6090	50	Kevin
Rajs 3675	50	Trenna
Davies 3255	50	Curtis
Matos 2730	50	Randall
Vargas 2625	50	Peter

3-2 SORTING ROWS

ORDER BY

ORDER BY digunakan untuk mengurutkan data.

Contoh :

- **Ascending**

```
SELECT last_name, hire_date
FROM employees
ORDER BY hire_date;
```

Hasil :

LAST_NAME	HIRE_DATE
King	17-Jun-1987
Whalen	17-Sep-1987
Kochhar	21-Sep-1989
Hunold	03-Jan-1990
Ernst	21-May-1991

- **Descending**

Contoh Syntax:

```
SELECT last_name, hire_date
FROM employees
ORDER BY hire_date DESC;
```

Hasil :

LAST_NAME	HIRE_DATE
Zlotkey	29-Jan-2000
Mourgos	16-Nov-1999
Grant	24-May-1999
Lorentz	07-Feb-1999
Vargas	09-Jul-1998

- **Menggunakan kolom Alias**

```
SELECT last_name, hire_date AS "Date Started"
FROM employees
ORDER BY "Date Started";
```

Hasil :

LAST_NAME	Date Started
King	17-Jun-1987
Whalen	17-Sep-1987
Kochhar	21-Sep-1989
Hunold	03-Jan-1990
Ernst	21-May-1991

- **Menggunakan kolom lain**

Contoh Syntax :

```
SELECT employee_id, first_name
FROM employees
WHERE employee_id < 105
ORDER BY last_name;
```

Hasil:

EMPLOYEE_ID	FIRST_NAME
102	Lex
104	Bruce
103	Alexander
100	Steven
101	Neena

- **Menggunakan banyak kolom**

```
SELECT department_id, last_name  
FROM employees  
WHERE department_id <= 50  
ORDER BY department_id, last_name;
```

Hasil :

DEPARTMENT_ID	LAST_NAME
10	Whalen
20	Fay
20	Hartstein
50	Davies
50	Matos
50	Mourgos
50	Rajs
50	Vargas

TUGAS MINGGUAN 10

Tuliskan *syntax* untuk menghasilkan *output* sebagai berikut:

- Petunjuk:** 1. Semua soal menggunakan tabel employees.
2 No. soal sesuai nomor materi.

3-1

Employee Raise	HIRE_DATE	MANAGER_ID	DEPARTMENT_ID	FIRST_NAME
Ellen 110000	11-May-1996	149	80	Ellen
Jonathon 86000	24-Mar-1998	149	80	Jonathon
Kimberely 70000	24-May-1999	149	-	Kimberely
Trenna 35000	17-Oct-1995	124	50	Trenna
Curtis 31000	29-Jan-1997	124	50	Curtis
Randall 26000	15-Mar-1998	124	50	Randall
Peter 25000	09-Jul-1998	124	50	Peter

3-1

```
SELECT *  
FROM employees  
WHERE salary <= 10000 AND job_id LIKE 'ST%' OR  
last name LIKE 'a%';
```

Output dari *syntax* diatas adalah....

3-2

LAST_NAME	Date Started	SALARY	DEPARTMENT_ID	EMAIL
Mourgos	16-Nov-1999	5800	50	KMOURGOS
Grant	24-May-1999	7000	-	KGRANT
Lorentz	07-Feb-1999	4200	60	DLORENTZ
Vargas	09-Jul-1998	2500	50	PVARGAS
Matos	15-Mar-1998	2600	50	RMATOS
Fay	17-Aug-1997	6000	20	PFAY
Davies	29-Jan-1997	3100	50	CDAVIES
Rajs	17-Oct-1995	3500	50	TRAJS
Ernst	21-May-1991	6000	60	BERNST
Whalen	17-Sep-1987	4400	10	JWHALEN

1.

Nama	DEPARTMENT_ID	JOB_ID	SALARY
Steven King	90	AD_PRES	24000
Neena Kochhar	90	AD_VP	17000
Lex De Haan	90	AD_VP	17000
Shelley Higgins	110	AC_MGR	12000
William Gietz	110	AC_ACCOUNT	8300
Eleni Zlotkey	80	SA_MAN	10500
Ellen Abel	80	SA_REP	11000
Jonathon Taylor	80	SA_REP	8600

2. Tuliskan syntax untuk memunculkan first_name, last_name, hire_date dan salary dimana first_name dimulai dengan huruf S atau last_name diakhiri dengan huruf A.
 - a. Apabila NPM anda ganjil maka urutkan salary dengan cara Ascending.
 - b. Apabila NPM anda genap maka urutkan salary dengan cara Descending.