

Oracle Academic Initiative

Oracle9i Introduction to SQL



Oleh:

Tessy Badriyah, SKom.MT

**Politeknik Elektronika Negeri Surabaya
Institut Teknologi Sepuluh Nopember
Surabaya**

BAB 7 : Subquery

7.1. Sasaran


- Menggambarkan tipe persoalan yang dapat dipecahkan oleh sub query
- Mendefinisikan sub query
- Memahami tipe-tipe dari subquery
- Menulis subquery baris tunggal dan baris berganda

7.2. Menggunakan Subquery untuk Menyelesaikan Masalah

Pengertian subquery akan dijelaskan melalui ilustrasi berikut ini :


Misal kita ingin membuat suatu query untuk mencari gaji pegawai yang lebih besar dari gaji yang dimiliki oleh pegawai bernama 'ABEL'.

Main Query:



Pegawai yang mana yang memiliki salary lebih besar dari salary kepunyaan Abel ?

Subquery



Berapa salary Abel ?

Untuk memecahkan persoalan ini, kita membutuhkan dua query, satu query untuk mencari gaji yang dimiliki oleh ABEL dan query lain untuk mencari pegawai yang memiliki gaji lebih besar daripada gaji ABEL.

Inner query atau subquery akan menghasilkan suatu nilai yang nantinya dipakai oleh outer query atau main query.

7.3. Sintak SubQuery

Sintak (cara penulisan) sub query :

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

7.4. Penggunaan SubQuery

Penggunaan Sub Query untuk menyelesaikan masalah seperti yang sudah dijelaskan sebelumnya :

```

SELECT last_name
FROM employees 11000
WHERE salary >
  (SELECT salary
   FROM employees
   WHERE last_name = 'Abel');

```

LAST_NAME
King
Kochhar
De Haan
Hartstein
Higgins

7.5. Petunjuk Penggunaan SubQuery

Berikut petunjuk dari penggunaan Sub Query :

- Letakkan subquery di dalam tanda kurung
- Tempatkan subquery pada sisi kanan dari kondisi perbandingan
- Klausula ORDER BY dalam subquery tidak diperlukan jika yang dibentuk adalah Top-N analysis.
- Gunakan operator baris tunggal dengan subquery baris tunggal dan operator baris ganda dengan sub query baris ganda.

7.6. Tipe-tipe SubQuery

Tipe-tipe dari subquery :

- Single row, sub query baris tunggal
- Multiple-row, sub query baris ganda
- Multiple-column, sub query kolom ganda

7.7. SubQuery Baris Tunggal

Single row subquery memberikan hasil hanya satu baris pada bagian subquery.

Untuk single row subquery ini yang digunakan adalah operator perbandingan :

= , > , >= , < , <= , atau <>.

7.8. Menjalankan SubQuery Baris Tunggal

Berikut ini contoh penggunaan Subquery baris tunggal :

```

SELECT last_name, job_id, salary
FROM employees
WHERE job_id =
  (SELECT job_id
   FROM employees
   WHERE employee id = 141)
AND salary >
  (SELECT salary
   FROM employees
   WHERE employee id = 143);

```

LAST_NAME	JOB_ID	SALARY
Rajs	ST_CLERK	3500
Davies	ST_CLERK	3100

7.9. Menggunakan Fungsi Group dalam SubQuery

Fungsi Group bisa digunakan dalam subquery.

```
SELECT last_name, job_id, salary
FROM employees
WHERE salary =
  (SELECT MIN(salary)
   FROM employees);
```

LAST_NAME	JOB_ID	SALARY
Vargas	ST_CLERK	2500

7.10. Klausa HAVING dengan SubQuery

Klausa HAVING juga dapat digunakan dengan subquery. Pada contoh berikut, Oracle Server menjalankan subquery terlebih dahulu kemudian mengembalikan hasilnya ke klausa HAVING.

```
SELECT department_id, MIN(salary)
FROM employees
GROUP BY department_id
HAVING MIN(salary) >
  (SELECT MIN(salary)
   FROM employees
   WHERE department_id = 50);
```

7.11. Apa yang salah dengan statement ini ?

Kesalahan dalam subquery yang mungkin terjadi adalah operator baris tunggal (=) digunakan pada subquery yang menghasilkan lebih dari satu baris, seperti pada contoh berikut :

Apa yang salah dengan statement ini ?

```
SELECT employee_id, last_name
FROM employees
WHERE salary =
  (SELECT MIN(salary)
   FROM employees
   GROUP BY department_id);
```

```
ERROR at line 4:
ORA-01427: single-row subquery returns more than
one row
```

Statement tersebut salah, karena : operator baris tunggal (single-row operator) digunakan bersama dengan subquery baris ganda (multiple-row subquery).

7.12. Apakah statement ini menghasilkan suatu baris data ?

```

SELECT last_name, job_id
FROM employees
WHERE job_id =
      (SELECT job_id
       FROM employees
       WHERE last_name = 'Haas');

```

```
no rows selected
```

7.13. SubQuery Baris Ganda

Multiple Row Subquery adalah subquery yang menghasilkan lebih dari satu baris. Untuk multiple row subquery ini yang digunakan adalah operator perbandingan : IN, ANY atau ALL.

Operator	Meaning
IN	Sama dengan anggota dalam daftar
ANY	Bandingkan suatu nilai terhadap setiap nilai yang dikembalikan oleh subquery
ALL	Bandingkan nilai terhadap setiap nilai yang dikembalikan oleh subquery

Ciri-ciri dari Subquery baris ganda :

- Mengembalikan lebih dari satu baris
- Digunakan dengan menggunakan Operator Baris-Ganda

7.14. Penggunaan Operator ANY dalam Subquery Baris Ganda

Operator ANY identik dengan operator SOME, yang membandingkan suatu nilai dengan tiap nilai yang ada dalam subquery.

Operator =ANY ekuivalen dengan IN.

Operator <ANY ekuivalen dengan MAXIMUM.

Operator >ANY ekuivalen dengan MINIMUM

```

SELECT employee_id, last_name, job_id, salary
FROM employees
WHERE salary < ANY
      (SELECT salary
       FROM employees
       WHERE job_id = 'IT_PROG')
AND job_id <> 'IT_PROG';

```

9000, 6000, 4200

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
124	Mourgos	ST_MAN	5800
141	Rajs	ST_CLERK	3500
142	Davies	ST_CLERK	3100
143	Matos	ST_CLERK	2600
144	Vargas	ST_CLERK	2500

10 rows selected.

7.15. Penggunaan Operator ALL dalam Subquery Baris Ganda

Operator ALL membandingkan suatu nilai dengan semua nilai yang ada dalam subquery.

Operator >ALL ekuivalen dengan MAKSIMUM.

Operator <ALL ekuivalen dengan MINIMUM

```

SELECT employee_id, last_name, job_id, salary
FROM employees
WHERE salary < ALL
      (SELECT salary
       FROM employees
       WHERE job_id = 'IT_PROG')
AND job_id <> 'IT_PROG';

```

9000, 6000, 4200

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
141	Rajs	ST_CLERK	3500
142	Davies	ST_CLERK	3100
143	Matos	ST_CLERK	2600
144	Vargas	ST_CLERK	2500

7.16. Nilai NULL dalam SubQuery

```

SELECT emp.last_name
FROM employees emp
WHERE emp.employee_id NOT IN
      (SELECT mgr.manager_id
       FROM employees mgr);

```

no rows selected

7.17. Latihan

1. Tampilkan data nama pegawai dan tanggal mulai bekerja untuk semua pegawai yang bekerja di department yang sama dengan 'BLAKE' tidak termasuk 'BLAKE'.

ENAME	HIREDATE
ALLEN	20-FEB-81
WARD	22-FEB-81
MARTIN	28-SEP-81
TURNER	08-SEP-81
JAMES	03-DEC-81

2. Tampilkan nomer dan nama pegawai untuk semua pegawai yang gajinya lebih dari rata-rata. Urutkan menurun berdasarkan besar gaji.

EMPNO	ENAME
7566	JONES
7698	BLAKE
7782	CLARK
7788	SCOTT
7839	KING
7902	FORD

6 rows selected.

3. Tampilkan nomer dan nama pegawai untuk semua pegawai yang bekerja di department yang sama dengan pegawai yang memiliki nama yang mengandung huruf 'T'.

EMPNO	ENAME
7369	SMITH
7876	ADAMS
7902	FORD
7788	SCOTT
7566	JONES
7499	ALLEN
7698	BLAKE
7654	MARTIN
7900	JAMES
7844	TURNER
7521	WARD

11 rows selected.

4. Tampilkan nama pegawai, nomer department dan pekerjaan untuk semua pegawai yang department tempatnya bekerja berlokasi di kota 'DALLAS'.

ENAME	DEPTNO	JOB
SMITH	20	CLERK
JONES	20	MANAGER
SCOTT	20	ANALYST
ADAMS	20	CLERK
FORD	20	ANALYST

5. Tampilkan nama dan gaji dari semua pegawai yang merupakan bawahan dari 'KING'

ENAME	SAL
JONES	2975
BLAKE	2850
CLARK	2450

6. Tampilkan nomer department, nama pegawai dan pekerjaan dari semua pegawai yang ada di department SALES.

DEPTNO	ENAME	JOB
30	ALLEN	SALESMAN
30	WARD	SALESMAN
30	MARTIN	SALESMAN
30	BLAKE	MANAGER
30	TURNER	SALESMAN
30	JAMES	CLERK

6 rows selected.

7. Tampilkan nomer dan nama pegawai untuk semua pegawai yang bekerja di department yang sama dengan pegawai yang memiliki nama yang mengandung huruf 'T', dan gaji yang dimiliki lebih besar daripada rata-rata gaji .

EMPNO	ENAME
7902	FORD
7788	SCOTT
7566	JONES
7698	BLAKE