

Group/Agregat Function



4

Melaporkan Data Agregat
Menggunakan *Group Functions*

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Apakah *Group Function* Itu ?

Group function beroperasi pada sekelompok baris-baris untuk memberikan satu hasil per kelompok.

EMPLOYEES

DEPARTMENT_ID	SALARY
30	24000
50	17000
90	17000
60	9000
60	6000
60	4200
50	5900
50	9500
50	3100
60	2500
50	2500
80	10500
80	11000
80	8900
	7000
110	48000

20 rows selected

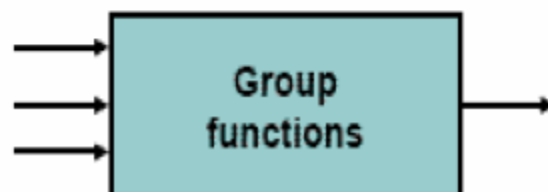
Penghasilan
Maksimum
dalam tabel
EMPLOYEES

MAX(SALARY)
24000

ORACLE

Type-Type *Group Functions*

- AVG
- COUNT
- MAX
- MIN
- STDDEV
- SUM
- VARIANCE



DATA MANIPULATION LANGUAGE (Review)

- **SELECT**

SSELECT [DISTINCT] < attribute list >

FFROM (< table name > { < alias > } | < joined table >)

{ , (< table name > { < alias > } | < joined table >) }

[**W**HERE < condition >]

[**G**ROUP BY < grouping attributes >

[**H**AVING < group selection condition >]]

[**O**RDER BY < col.name > [< order >] { , < col.name > [< order >] }]

< attribute list > ::= (* | (< col.name > | < function >

(([DISTINCT] < col.name > | *)))

{ , < col.name > | < function >

(([DISTINCT] < col.name > | *)) }))

< grouping attributes > := < col.name > { , < col.name > }

< order > := (ASC | DESC)

Group Functions : Sintak

```
SELECT      [column,] group_function(column), ...  
FROM        table  
[WHERE      condition]  
[GROUP BY  column]  
[ORDER BY  column];
```

Menggunakan Fungsi-Fungsi MIN dan MAX

Anda dapat menggunakan MIN dan MAX untuk tipe-tipe data *numeric*, *character*, dan *date*.

```
SELECT MIN(hire date), MAX(hire date)
FROM employees;
```

MIN(HIRE_	MAX(HIRE_
17-JUN-07	29-JAN-00

Menggunakan Fungsi-Fungsi AVG dan SUM

Anda dapat menggunakan AVG dan SUM untuk data numerik.

```
SELECT AVG(salary), MAX(salary),
       MIN(salary), SUM(salary)
FROM employees
WHERE job_id LIKE '%REP%';
```

AVG(SALARY)	MAX(SALARY)	MIN(SALARY)	SUM(SALARY)
8150	11000	8000	32800

Menggunakan Fungsi COUNT

COUNT (*) mengembalikan jumlah baris-baris dalam suatu tabel :

1

```
SELECT COUNT(*)  
FROM employees  
WHERE department_id = 50;
```

COUNT(*)

6

COUNT (expr) mengembalikan jumlah baris-baris *non-null* untuk suatu *expr* :

2

```
SELECT COUNT(commission_pct)  
FROM employees  
WHERE department_id = 80;
```

COUNT(COMMISSION_PCT)

3

ORACLE

Menggunakan Kata Kunci DISTINCT

- COUNT (DISTINCT expr) mengembalikan jumlah dari nilai-nilai *non-null* berbeda dari *expr*.
- Untuk menampilkan jumlah dari nilai-nilai departemen berbeda dalam tabel EMPLOYEES :

```
SELECT COUNT(DISTINCT department_id)
FROM employees;
```

COUNT(DISTINCTDEPARTMENT_ID)
7

Membuat Data-Data Terkelompok : Sintak Klausa GROUP BY

```
SELECT    column, group_function(column)
FROM      table
[WHERE    condition]
[GROUP BY group_by_expression]
[ORDER BY column];
```

Anda dapat membagi baris-baris dalam suatu tabel kedalam sekelompok kecil dengan menggunakan klausa GROUP BY.

Data Terkelompok

Mahasiswa



Mahasiswa Prodi S1 SI



Mahasiswa Prodi S1 SI Perempuan



Mahasiswa Prodi S1 SI Laki-Laki

Mahasiswa Prodi S1 TK



Mahasiswa Prodi S1 TK Perempuan



Mahasiswa Prodi S1 TK Laki-Laki

Menggunakan Klausa GROUP BY

Semua kolom pada daftar SELECT yang bukan *group functions* harus ada pada klausa GROUP BY.

```
SELECT department_id, AVG(salary)
FROM employees
GROUP BY department_id ;
```

DEPARTMENT_ID	AVG(SALARY)
10	4400
20	9500
60	3500
60	6000
80	10033.3333
90	19333.3333
110	10150
	7000

8 rows selected.

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Membatasi Hasil-hasil Pengelompokan dengan Klausa **HAVING**

Saat Anda menggunakan klausa **HAVING**, server Oracle membatasi pengelompokan sebagai berikut :

1. Baris-baris dikelompokkan.
2. *Group Function* diterapkan.
3. Pengelompokan yang memenuhi klausa **HAVING** ditampilkan.

```
SELECT      column, group_function
FROM        table
[WHERE      condition]
[GROUP BY  group_by_expression]
[HAVING    group_condition]
[ORDER BY  column];
```

Praktekkan Klausu Agregat Function!

1. Tampilkan berapa total semua stok Barang yang ada!
2. Tampilkan berapa rata-rata stok Barangnya!
3. Tampilkan berapa banyak data barang yang dimiliki!
4. Tempilkan berapa stok Barang paling sedikit!
5. Tampilkan berapa stok Barang paling banyak!
6. Tampilkan berapa banyak data Supplier yang dimiliki!
7. Tampilkan jumlah stok per jenis Barang. Urutkan berdasarkan jumlah paling banyak!
8. Tampilkan jumlah Supplier per jenis kelamin. Urutkan berdasarkan jumlah paling sedikit!
9. Tampilkan jumlah Barang per jenis Barang per jenis kemasan!
10. Tampilkan jumlah Supplier per tahun lahir per jenis kelamin!
11. **Buat tangkapan layar** untuk hasilnya klausa Agregat Function diatas. Hasil kumpulan tangkapan layar diunggah melalui Brilian dengan nama file : **Demo4_Q1_NimPanjang_Nama.PDF** atau **Demo4_P1_NimPanjang_Nama.PDF** paling lambat hari Sabtu, 12 Juni 2021 pukul 23:59 WIB.

Tetap
Semangat