



VERSI 2.0
13 FEBRUARY, 2023

BASIS DATA

MODUL 4 MATERI : AGGREGATE FUNCTION

TIM PENYUSUN :
BRIANSYAH SETIO WIYONO, S. KOM
CLARISSA SANINDITA REIKISYIFA
CHINTYA TRIA DIANA OKTAVIANI

PRESENTED BY: LAB. INFORMATIKA
UNIVERSITAS MUHAMMADIYAH MALANG

BASIS DATA

PERSIAPAN MATERI

Praktikan mempelajari tentang konsep aggregate function, grouping function, dan sorting.

TUJUAN

1. Praktikan mampu memahami konsep perintah dasar SELECT, WHERE, aggregate function, grouping function, dan sorting.
2. Praktikan mampu mengimplementasikan perintah dasar SELECT, WHERE, aggregate function, grouping function, dan sorting.

TARGET MODUL

1. Memahami konsep perintah dasar SELECT, WHERE, aggregate function, grouping function, dan sorting.
2. Mengimplementasikan perintah dasar SELECT, WHERE, aggregate function, grouping function, dan sorting

PERSIAPAN SOFTWARE/APLIKASI

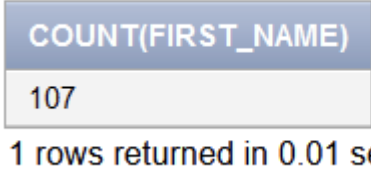
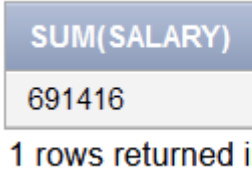
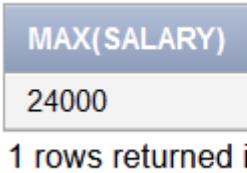
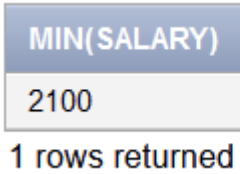
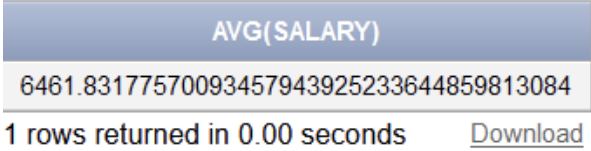
Oracle 11g XE

HR Schema

MATERI POKOK**1. AGGREGATE FUNCTION**

Aggregate Function di SQL melakukan perhitungan pada banyak nilai dan mengembalikan nilai tunggal. SQL menyediakan banyak Aggregate Function yang mencakup AVG, COUNT, SUM, MIN, MAX, dll. Aggregate Function mengabaikan nilai NULL saat melakukan perhitungan, kecuali untuk fungsi count. Aggregate Function sering digunakan dengan klausa GROUP BY dan HAVING dari pernyataan SELECT. Berikut merupakan contoh penggunaan Aggregate Function yang disediakan SQL pada skema HR :

Perintah	Fungsi	Contoh Penggunaan Query dan Outputnya
COUNT	Digunakan untuk menghitung jumlah record	SELECT COUNT (first_name) FROM employees;

		
SUM	Digunakan untuk menghitung total nilai dari kolom tertentu	SELECT SUM (salary) FROM employees; 
MAX	Digunakan untuk menampilkan nilai tertinggi dari suatu kolom	SELECT MAX (salary) FROM employees; 
MIN	Digunakan untuk menampilkan nilai terkecil dari suatu kolom	SELECT MIN (salary) FROM employees; 
AVG	Digunakan untuk menampilkan nilai rata-rata dari suatu kolom	SELECT AVG (salary) FROM employees; 

2. PERINTAH DASAR SELECT

SELECT menjadi perintah dasar dari SQL dengan tipe sintaks DML. SELECT digunakan untuk menampilkan, mengambil maupun memilah informasi dari database atau data dari satu tabel. Contoh percobaan menggunakan skema HR :

- a) Menampilkan semua kolom pada suatu tabel dapat menggunakan keyword * (bintang) setelah keyword SELECT.

Syntax :

```
SELECT *
FROM EMPLOYEES;
```

Output :

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
100	Steven	King	SKING	515.123.4567	06/17/2003	AD_PRES	24000	-	-	90
101	Neena	Kochhar	NKOCHHAR	515.123.4568	09/21/2005	AD_VP	17000	-	100	90
102	Lex	De Haan	LDEHAAN	515.123.4569	01/13/2001	AD_VP	17000	-	100	90
103	Alexander	Hunold	AHUNOLD	590.423.4567	01/03/2006	IT_PROG	9000	-	102	60
104	Bruce	Ernst	BERNST	590.423.4568	05/21/2007	IT_PROG	6000	-	103	60
105	David	Austin	DAUSTIN	590.423.4569	06/25/2005	IT_PROG	4800	-	103	60
106	Valli	Pataballa	VPATABAL	590.423.4560	02/05/2006	IT_PROG	4800	-	103	60
107	Diana	Lorentz	DLORENTZ	590.423.5567	02/07/2007	IT_PROG	4200	-	103	60
108	Nancy	Greenberg	NGREENBE	515.124.4569	08/17/2002	FI_MGR	12008	-	101	100
109	Daniel	Faviet	DFAVIET	515.124.4169	08/16/2002	FI_ACCOUNT	9000	-	108	100

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds [Download](#)

- b) Menampilkan beberapa kolom pada suatu tabel

Syntax :

```
SELECT first_name, salary
FROM EMPLOYEES;
```

Output :

FIRST_NAME	SALARY
Steven	24000
Neena	17000
Lex	17000
Alexander	9000
Bruce	6000
David	4800
Valli	4800
Diana	4200
Nancy	12008
Daniel	9000

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.01 seconds [Download](#)

- c) SELECT DISTINCT hanya menampilkan baris yang berbeda dari suatu kolom

Syntax :

```
SELECT DISTINCT department_id
FROM EMPLOYEES;
```

Output :

DEPARTMENT_ID
100
30
-
90
20
70
110
50
80
40
More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.01 seconds [Download](#)

- d) Perintah SELECT juga dapat digunakan bersamaan dengan operator aritmatika seperti penjumlahan (+), pengurangan (-), perkalian (*), dan pembagian (/).

Contoh syntax :

```
SELECT last_name, salary, salary + 300
FROM EMPLOYEES;
```

Output :

LAST_NAME	SALARY	SALARY+300
King	24000	24300
Kochhar	17000	17300
De Haan	17000	17300
Hunold	9000	9300
Ernst	6000	6300
Austin	4800	5100
Pataballa	4800	5100
Lorentz	4200	4500
Greenberg	12008	12308
Faviet	9000	9300
More than 10 rows available. Increase rows selector to view more rows.		

10 rows returned in 0.00 seconds [Download](#)

- e) Perintah AS untuk mengganti sementara nama tabel atau kolom. Disebut sementara karena pada dasarnya nama kolom dari suatu tabel tersebut tidak berubah, hanya diganti pada saat ditampilkan dengan query SELECT.

Syntax :

```
SELECT last_name AS name
FROM EMPLOYEES;
```

Output :

NAME
Abel
Ande
Atkinson
Austin
Baer
Baida
Banda
Bates
Bell
Bernstein
More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.01 seconds [Download](#)

- f) Perintah SELECT menggunakan operator gabungan (||) digunakan untuk menggabungkan dua atau lebih kolom menjadi satu kolom.

Syntax :

```
SELECT last_name||' '||first_name "Full Name"
FROM employees;
```

Output :

Full Name
Abel Ellen
Ande Sundar
Atkinson Mozhe
Austin David
Baer Hermann
Baida Shelli
Banda Amit
Bates Elizabeth
Bell Sarah
Bernstein David
More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds [Download](#)

3. PERINTAH DASAR WHERE

WHERE digunakan untuk memfilter hasil SELECT dengan menampilkan data yang memenuhi persyaratan saja yang akan ditampilkan/eksekusi.

Contoh syntax :

```
SELECT employee_id, last_name, job_id, department_id
FROM EMPLOYEES
```

```
WHERE department_id = 90;
```

Output hanya menampilkan department_id pada tabel EMPLOYEES yang memiliki nilai 90

EMPLOYEE_ID	LAST_NAME	JOB_ID	DEPARTMENT_ID
100	King	AD_PRES	90
101	Kochhar	AD_VP	90
102	De Haan	AD_VP	90

3 rows returned in 0.00 seconds [Download](#)

Perintah WHERE biasanya digunakan bersamaan dengan operator perbandingan dan operator logika.

a) Operator Perbandingan

Operator	Deskripsi
=	Sama dengan
>	Lebih besar
>=	Lebih besar sama dengan
<	Kurang dari
<=	Kurang dari sama dengan
<>	Tidak sama
BETWEEN...AND...	Antara rentang tertentu
IN (set)	Mencocokkan salah satu daftar nilai
LIKE	Mencocokkan pola karakter
IS NULL	Apakah bernilai NULL

Contoh :

```
SELECT first_name
FROM employees
```

```
WHERE first_name LIKE 'S%';
```

Output akan menampilkan kolom first_name yang memiliki huruf berawalan 'S'

FIRST_NAME
Sundar
Shelli
Sarah
Shelley
Steven
Sundita
Steven
Susan
Samuel
Sarath
More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds

[Download](#)

b) Operator Logika

Operator	Deskripsi
AND	Mengembalikan nilai TRUE jika kedua kondisi bernilai TRUE
OR	Mengembalikan nilai TRUE jika salah satu kondisi bernilai TRUE
NOT	Mengembalikan nilai TRUE jika kondisi bernilai FALSE dan mengembalikan nilai FALSE jika kondisi bernilai TRUE

Contoh :

```
SELECT employee_id, last_name, job_id, salary
FROM employees
WHERE salary >= 10000
AND job_id NOT LIKE '%MAN%';
```

Output akan menampilkan kolom employee_id, last_name, job_id, salary yang memiliki gaji lebih dari 10000 dan job_id yang tidak mengandung kata 'MAN'

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
100	King	AD_PRES	24000
101	Kochhar	AD_VP	17000
102	De Haan	AD_VP	17000
108	Greenberg	FI_MGR	12008
150	Tucker	SA_REP	10000
156	King	SA_REP	10000
162	Vishney	SA_REP	10500
168	Ozer	SA_REP	11500
169	Bloom	SA_REP	10000
174	Abel	SA_REP	11000
More than 10 rows available. Increase rows selector to view more rows.			

10 rows returned in 0.01 seconds

[Download](#)

4. GROUPING FUNCTION

a) Group By

Group by merupakan statement di dalam SQL yang digunakan untuk mengatur data yang identik ke dalam kelompok dengan bantuan beberapa fungsi.

Contoh :

```
SELECT job_id, avg(salary)
```

```
FROM employees
```

```
GROUP BY job_id;
```

Output :

JOB_ID	AVG(SALARY)
IT_PROG	5760
AC_MGR	12008
AC_ACCOUNT	8300
ST_MAN	7280
PU_MAN	11000
AD_ASST	4400
AD_VP	17000
SH_CLERK	3215
FI_ACCOUNT	7920
FI_MGR	12008
More than 10 rows available. Increase rows selector to view more rows.	

10 rows returned in 0.00 seconds

[Download](#)

b) Having

Having digunakan untuk menentukan pengelompokan-pengelompokan yang akan ditampilkan. Lebih jauh lagi having dapat membatasi pengelompokan berdasarkan informasi Aggregate Function.

Contoh :

```
SELECT job_id, avg(salary)
FROM employees
GROUP BY job_id
HAVING avg(salary) > 10000;
```

Output :

JOB_ID	AVG(SALARY)
AC_MGR	12008
PU_MAN	11000
AD_VP	17000
FI_MGR	12008
SA_MAN	12200
MK_MAN	13000
AD_PRES	24000

7 rows returned in 0.01 seconds

5. SORTING

a) Ascending

Pengurutan data dari kecil ke besar

Contoh :

```
SELECT job_id, avg(salary)
FROM employees
GROUP BY job_id
HAVING avg(salary) > 10000
ORDER BY avg(salary) asc;
```

Output :

JOB_ID	AVG(SALARY)
AC_MGR	12008
PU_MAN	11000
AD_VP	17000
FI_MGR	12008
SA_MAN	12200
MK_MAN	13000
AD_PRES	24000

7 rows returned in 0.01 seconds

b) Descending

Pengurutan data dari besar ke kecil

Contoh :

```
SELECT job_id, avg(salary)
FROM employees
GROUP BY job_id
HAVING avg(salary) > 10000
ORDER BY avg(salary) desc;
```

Output :

JOB_ID	AVG(SALARY)
AD_PRES	24000
AD_VP	17000
MK_MAN	13000
SA_MAN	12200
AC_MGR	12008
FI_MGR	12008
PU_MAN	11000

7 rows returned in 0.01 seconds

LATIHAN

Lakukan latihan ini dengan menggunakan skema HR

1. Tampilkan tanggal masuk dari karyawan yang pertama dan terakhir kali dipekerjakan

MIN(HIRE_DATE)	MAX(HIRE_DATE)
01/13/2001	04/21/2008

1 rows returned in 0.00 seconds [Download](#)

2. Tampilkan Gaji tertinggi dari seorang Programmer (IT_PROG) dan bekerja pada manager_id = 103

MAX(SALARY)
6000

1 rows returned in 0.00 seconds

3. Tampilkan total gaji dari karyawan yang memiliki nama depan berinisial B dan ganti nama kolomnya menjadi "GAJI_KARYAWAN"

GAJI_KARYAWAN
9900

1 rows returned in 0.01 seconds

4. Tampilkan rata - rata gaji dan departement_id dan dikelompokkan berdasarkan job_id dan urutkan dari yang memiliki rata - rata gaji terbesar ke terkecil

AVG(SALARY)	DEPARTMENT_ID
5760	60
12008	100
11000	30
7280	50
8396.55172413793103448275862068965517241	80
17000	90
4400	10
13000	20
6000	20
6500	40
More than 10 rows available. Increase rows selector to view more rows.	

10 rows returned in 0.00 seconds [Download](#)