



El empleo
es de todos

Mintrabajo

DESARROLLO DE BASE DE DATOS CON MYSQL



@SENAcomunica

www.sena.edu.co



DESARROLLO DE BASE DE DATOS CON MYSQL



@SENAcomunica

www.sena.edu.co

4

FUNCIONES DE GRUPO



Después de completar esta lección usted estará en la capacidad de:

- **Identificar las funciones de grupo disponibles.**
- **Describir el uso de las funciones de grupo**
- **Agrupar mediante la clausula GROUP BY**
- **Incluir o excluir filas agrupadas utilizando la clausula HAVING**

¿Qué son las funciones de grupo?



- **Operan en juegos de filas para dar un resultado por grupo.**

Tipos de funciones de grupo



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

Funciones de grupo: Sintaxis



COUNT(*) devuelve el número de filas en una tabla.

```
MariaDB [HR]> SELECT COUNT(*)  
-> FROM EMPLOYEES  
-> WHERE DEPARTMENT_ID=50;  
  
+-----+  
| COUNT(*) |  
+-----+  
|         45 |  
+-----+  
1 row in set (0.01 sec)
```

Uso de la funciones de grupo AVG y SUM



```
MariaDB [HR]> SELECT AVG(SALARY),MAX(SALARY),  
-> MIN(SALARY), SUM(SALARY)  
-> FROM EMPLOYEES  
-> WHERE JOB_ID LIKE '%REP%';
```

AVG(SALARY)	MAX(SALARY)	MIN(SALARY)	SUM(SALARY)
8272.727273	11500.00	6000.00	273000.00

1 row in set (0.06 sec)

Uso de la funciones de grupo MIN y MAX



```
MariaDB [HR]> SELECT MIN(HIRE_DATE),MAX(HIRE_DATE)  
-> FROM EMPLOYEES;
```

```
+-----+-----+  
| MIN(HIRE_DATE) | MAX(HIRE_DATE) |  
+-----+-----+  
| 0000-00-00      | 2000-04-21      |  
+-----+-----+  
1 row in set (0.01 sec)
```

Uso de la funciones COUNT



```
MariaDB [HR]> SELECT COUNT(*)  
-> FROM EMPLOYEES  
-> WHERE DEPARTMENT_ID=80;
```

COUNT(*)
34

```
1 row in set (0.00 sec)
```

Uso de la funciones COUNT



```
MariaDB [HR]> SELECT COUNT(COMMISSION_PCT)
-> FROM EMPLOYEES
-> WHERE DEPARTMENT_ID=80;
+-----+
| COUNT(COMMISSION_PCT) |
+-----+
|                34    |
+-----+
1 row in set (0.00 sec)
```

Uso de la palabra clave DISTINCT



- **COUNT(DISTINCT(expr))** devuelve el número de valores no nulos distintos de expr.
- Para mostrar el número de valores de departamento distintos en la tabla **EMPLOYEES**

```
MariaDB [HR]> SELECT COUNT(DISTINCT(DEPARTMENT_ID))  
-> FROM EMPLOYEES;
```

```
+-----+  
| COUNT(DISTINCT(DEPARTMENT_ID)) |  
+-----+  
|                               11 |  
+-----+
```

```
1 row in set (0.00 sec)
```

Funciones de grupo y valores nulos



- Las funciones de grupo ignoran los valores nulos en una columna.

```
MariaDB [HR]> SELECT AVG(COMMISSION_PCT)
-> FROM EMPLOYEES;
+-----+
| AVG(COMMISSION_PCT) |
+-----+
|          0.222857   |
+-----+
1 row in set (0.00 sec)
```

- Controlando los valores nulos el promedio cambia.

```
MariaDB [HR]> SELECT AVG(IFNULL(COMMISSION_PCT,0))
-> FROM EMPLOYEES;
+-----+
| AVG(IFNULL(COMMISSION_PCT,0)) |
+-----+
|          0.072222   |
+-----+
1 row in set (0.00 sec)
```

Creación de grupos de datos con GROUP BY



- Puede dividir las filas de una tabla en grupos más pequeños mediante la clausula GROUP BY.

```
MariaDB [HR]> SELECT DEPARTMENT_ID, AVG(SALARY)
-> FROM EMPLOYEES
-> GROUP BY DEPARTMENT_ID;
```

DEPARTMENT_ID	AVG(SALARY)
NULL	3500.000000
10	4400.000000
20	9500.000000
30	4150.000000
40	6500.000000
50	3475.555556
60	5760.000000
70	10000.000000
80	8955.882353
90	19333.333333
100	8600.000000
110	10150.000000

12 rows in set (0.01 sec)

Uso de la clausula GROUP BY en varias filas.



```
MariaDB [HR]> SELECT DEPARTMENT_ID, JOB_ID, SUM(SALARY)
-> FROM EMPLOYEES
-> GROUP BY DEPARTMENT_ID, JOB_ID;
```

DEPARTMENT_ID	JOB_ID	SUM(SALARY)
NULL	SA_REP	7000.00
NULL	ST_CLERK	0.00
10	AD_ASST	4400.00
20	MK_MAN	13000.00
20	MK_REP	6000.00
30	PU_CLERK	13900.00
30	PU_MAN	11000.00
40	HR_REP	6500.00
50	SH_CLERK	64300.00
50	ST_CLERK	55700.00
50	ST_MAN	36400.00
60	IT_PROG	28800.00
70	PR_REP	10000.00
80	SA_MAN	61000.00
80	SA_REP	243500.00
90	AD_PRES	24000.00
90	AD_VP	34000.00
100	FI_ACCOUNT	39600.00
100	FI_MGR	12000.00
110	AC_ACCOUNT	8300.00
110	AC_MGR	12000.00

```
21 rows in set (0.00 sec)
```

Uso indebido de la cláusula GROUP BY

- No se puede utilizar la cláusula WHERE para restringir grupos.
- Utilice la cláusula HAVING para restringir grupos.
- No puede utilizar funciones de grupo en la cláusula WHERE.

```
MariaDB [HR]> SELECT DEPARTMENT_ID, AVG(SALARY)
-> FROM EMPLOYEES
-> WHERE AVG(SALARY)>8000
-> GROUP BY DEPARTMENT_ID;
ERROR 1111 (HY000): Invalid use of group function
```


Restricciones de resultados de grupo con la cláusula HAVING



- Si se usa la cláusula HAVING restringe los grupos así:
- Se agrupan las filas.
- Se aplica la función de grupo.
- Se muestran los grupos que satisface la cláusula HAVING

```
MariaDB [HR]> SELECT DEPARTMENT_ID, MAX(SALARY)
-> FROM EMPLOYEES
-> GROUP BY DEPARTMENT_ID
-> HAVING MAX(SALARY)>10000;
```

DEPARTMENT_ID	MAX(SALARY)
20	13000.00
30	11000.00
80	14000.00
90	24000.00
100	12000.00
110	12000.00

6 rows in set (0.00 sec)

Restricciones de resultados de grupo con la clausula HAVING



```
MariaDB [HR]> SELECT JOB_ID, SUM(SALARY)
-> FROM EMPLOYEES
-> WHERE JOB_ID NOT LIKE '%REP%'
-> GROUP BY JOB_ID
-> HAVING SUM(SALARY)>13000
-> ORDER BY SUM(SALARY);
```

JOB_ID	SUM(SALARY)
PU_CLERK	13900.00
AD_PRES	24000.00
IT_PROG	28800.00
AD_VP	34000.00
ST_MAN	36400.00
FI_ACCOUNT	39600.00
ST_CLERK	55700.00
SA_MAN	61000.00
SH_CLERK	64300.00

```
9 rows in set (0.01 sec)
```



GRACIAS

Línea de atención al ciudadano: 018000 910270
Línea de atención al empresario: 018000 910682



@SENAcomunica

www.sena.edu.co

CRÉDITOS



Realizado por el instructor José Fernando Galindo Suárez
jgalindos@sena.edu.co 2020

