

Aula 4

Funções de Agrupamento

UNISUL – Computação - Banco de Dados

O que são funções de grupo?
Funções de grupo operam com um conjunto de linhas para resultar uma linha por grupo.

EMP

DEPTNO	SAL
10	2450
10	5000
10	1300
20	800
20	1100
20	3000
20	3000
20	2975
30	1600
30	2850
30	1250
30	950
30	1500
30	1250

“Maior salário da tabela EMP”

MAX (SAL)
5000

-2

UNISUL – Computação - Banco de Dados

Tipos de Funções de Agrupamento

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

-3

UNISUL – Computação - Banco de Dados

Utilização de Funções de Agrupamento

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

-4

UNISUL – Computação - Banco de Dados

Funções AVG e SUM

São funções que operam com dados numéricos.

```
SQL> SELECT  AVG(sal), MAX(sal),
2             MIN(sal), SUM(sal)
3 FROM      emp
4 WHERE     job LIKE 'SALES%';
```

AVG (SAL)	MAX (SAL)	MIN (SAL)	SUM (SAL)
1400	1600	1250	5600

-5

UNISUL – Computação - Banco de Dados

Funções MIN e MAX

Podem ser utilizadas com qualquer tipo de dado.

```
SQL> SELECT  MIN(hiredate), MAX(hiredate)
2 FROM      emp;
```

MIN (HIRED)	MAX (HIRED)
17-DEC-80	12-JAN-83

-6

UNISUL – Computação - Banco de Dados

Função COUNT

A função COUNT(*) retorna o número de linhas de uma tabela.

```
SQL> SELECT  COUNT(*)
2 FROM      emp
3 WHERE     deptno = 30;
```

COUNT (*)
6

-7

UNISUL – Computação - Banco de Dados

Função COUNT

A função COUNT(expr) retorna o número de linhas não nulas.

```
SQL> SELECT  COUNT(comm)
2 FROM      emp
3 WHERE     deptno = 30;
```

COUNT (COMM)
4

-8

UNISUL – Computação - Banco de Dados

Funções de Agrupamento e Valores Nulos

As funções de Agrupamento ignoram valores nulos.

```
SQL> SELECT AVG(comm)
2 FROM emp;
```

AVG (COMM)
550

-9

UNISUL – Computação - Banco de Dados

Função NVL com função de Agrupamento

A função NVL força a função de agrupamento a incluir todos as linhas, inclusive as que possuem valores nulos.

```
SQL> SELECT AVG(NVL(comm, 0))
2 FROM emp;
```

AVG (NVL (COMM, 0))
157.14286

-10

UNISUL – Computação - Banco de Dados

Criando Grupos de Dados

EMP

DEPTNO	SAL
10	2450
10	5000
10	1300
20	800
20	1100
20	3000
20	3000
20	2975
30	1600
30	2850
30	1250
30	950
30	1500
30	1250

2916.6667
2175
1566.6667

"média salarial da tabela EMP para cada departamento"

DEPTNO	AVG (SAL)
10	2916.6667
20	2175
30	1566.6667

-11

UNISUL – Computação - Banco de Dados

Cláusula GROUP BY

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

Divide as linhas de uma tabela em grupos.

-12

UNISUL – Computação - Banco de Dados

Cláusula GROUP BY

Todas as colunas do comando SELECT que não pertencem ao agrupamento devem estar presentes na cláusula GROUP BY.

```
SQL> SELECT deptno, AVG(sal)
2 FROM emp
3 GROUP BY deptno;
```

DEPTNO	AVG (SAL)
10	2916.6667
20	2175
30	1566.6667

-13

UNISUL – Computação - Banco de Dados

Cláusula GROUP BY

A coluna pertencente ao GROUP BY não necessita estar presente na lista de colunas do comando SELECT.

```
SQL> SELECT AVG(sal)
2 FROM emp
3 GROUP BY deptno;
```

AVG (SAL)
2916.6667
2175
1566.6667

-14

UNISUL – Computação - Banco de Dados

Agrupando mais de uma coluna

EMP

DEPTNO	JOB	SAL
10	MANAGER	2450
10	PRESIDENT	5000
10	CLERK	1300
20	CLERK	800
20	CLERK	1100
20	ANALYST	3000
20	ANALYST	3000
20	MANAGER	2975
30	SALESMAN	1600
30	MANAGER	2850
30	SALESMAN	1250
30	CLERK	950
30	SALESMAN	1500
30	SALESMAN	1250

soma de salários na tabela EMP para cada emprego, agrupada por departamentos"

DEPTNO	JOB	SUM(SAL)
10	CLERK	1300
10	MANAGER	2450
10	PRESIDENT	5000
20	ANALYST	6000
20	CLERK	1900
20	MANAGER	2975
30	CLERK	950
30	MANAGER	2850
30	SALESMAN	5600

-15

UNISUL – Computação - Banco de Dados

Utilização da Cláusula GROUP BY com Múltiplas Colunas

```
SQL> SELECT deptno, job, sum(sal)
2 FROM emp
3 GROUP BY deptno, job;
```

DEPTNO	JOB	SUM(SAL)
10	CLERK	1300
10	MANAGER	2450
10	PRESIDENT	5000
20	ANALYST	6000
20	CLERK	1900
...		

9 rows selected.

-16

UNISUL – Computação - Banco de Dados

Erro no agrupamento de Funções

Qualquer coluna ou expressão no comando SELECT que não é agregável, deve estar presente na cláusula GROUP BY.

```
SQL> SELECT deptno, COUNT(ename)
2 FROM emp;
```

```
SELECT deptno, COUNT(ename)
*
ERROR at line 1:
ORA-00937: not a single-group group function
```

-17

UNISUL – Computação - Banco de Dados

Erro no agrupamento de Funções

- Não é permitido restrição de grupos pela cláusula WHERE.
- Utiliza-se a cláusula HAVING para restringir grupos.

```
SQL> SELECT deptno, AVG(sal)
2 FROM emp
3 WHERE AVG(sal) > 2000
4 GROUP BY deptno;
```

```
WHERE AVG(sal) > 2000
*
ERROR at line 3:
ORA-00934: group function is not allowed here
```

-18

UNISUL – Computação - Banco de Dados

Excluindo Grupos

EMP

DEPTNO	SAL
10	2450
10	5000
10	1300
20	900
20	1100
20	3000
20	3000
20	2975
30	1600
30	2850
30	1250
30	950
30	1500
30	1250

5000

3000

2850

“Maior salário por departamento maior que \$2900”

DEPTNO	MAX (SAL)
10	5000
20	3000

-19

UNISUL – Computação - Banco de Dados

Cláusula HAVING

Utiliza-se a cláusula HAVING para restringir grupos

- As linhas são agrupadas.
- A função de agrupamento é aplicada.
- Os grupos são restringidos através da cláusula HAVING.

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

-20

UNISUL – Computação - Banco de Dados

Cláusula HAVING

```
SQL> SELECT deptno, max(sal)
2 FROM emp
3 GROUP BY deptno
4 HAVING max(sal) > 2900;
```

DEPTNO	MAX (SAL)
10	5000
20	3000

-21

UNISUL – Computação - Banco de Dados

Cláusula HAVING

```
SQL> SELECT job, SUM(sal) PAYROLL
2 FROM emp
3 WHERE job NOT LIKE 'SALES%'
4 GROUP BY job
5 HAVING SUM(sal) > 5000
6 ORDER BY SUM(sal);
```

JOB	PAYROLL
ANALYST	6000
MANAGER	8275

-22

UNISUL – Computação - Banco de Dados

Aninhamento de Funções de Agrupamento

Maior média salarial.

```
SQL> SELECT max(avg(sal))
2 FROM emp
3 GROUP BY deptno;
```

MAX (AVG (SAL))
2916.6667

-23

UNISUL – Computação - Banco de Dados