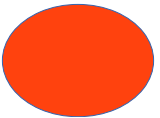


Modelo Relacional

SQL – GROUP BY

Prof. André Britto



Group By

- Muitas vezes queremos aplicar as funções e agregação para um grupo de tuplas.
 - Média geral de um aluno.
 - Maior nota de uma disciplina.
 - Número de alunos matriculados em uma disciplina.
- Cláusula GROUP BY

Group By

Liste a média dos salários por departamento:

Group By

Liste a média dos salários por departamento:

```
SELECT departamento, avg(salario)
```

```
FROM professor p JOIN cargo ca
```

```
    ON (p.cargo = ca.id_cargo)
```

```
GROUP BY departamento
```

Group By

- Se usarmos o group by sem a função de agregação, o resultado será uma tabela com os valores distintos do campo agrupado

```
SELECT departamento
```

```
FROM professor
```

```
GROUP BY departamento
```

Group By

- Podemos agrupar mais de um atributo
Liste o número de professores que ganham o mesmo salário por departamento

Group By

- Podemos agrupar mais de um atributo

Liste o número de professores que ganham o mesmo salário por departamento

```
SELECT departamento, salario, count(*)
```

```
FROM professor p JOIN cargo ca
```

```
ON (p.cargo = ca.id_cargo)
```

```
GROUP BY departamento, salario
```

Group By

- Podemos também efetuar mais de uma função de agregação por grupo

Liste o número de professores e média dos salários por departamento.

```
SELECT departamento, count(*), avg(salario)
```

```
FROM professor p JOIN cargo ca
```

```
ON (p.cargo = ca.id_cargo)
```

```
GROUP BY departamento
```


Group By

- Podemos também listar mais de um atributo no SELECT
 - Desde que ele esteja no GROUP BY ou o atributo do GROUP BY seja explicitamente a chave primária da tabela

Liste o nome e número de professores por departamento.

Group By

- Podemos também listar mais de um atributo no SELECT
 - Desde que ele esteja no GROUP BY ou o atributo do GROUP BY seja explicitamente a chave primária da tabela

Liste o nome e número de professores por departamento

```
SELECT nome, count(*),
```

```
FROM professor p JOIN cargo ca
```

```
ON (p.cargo = ca.id_cargo) JOIN departamento de
```

```
ON (p.departamento = de.cod_depto )
```

```
GROUP BY de.cod_depto
```

Having

- É útil definirmos uma condição que seja aplicada a um grupo de tuplas.
- Por exemplo:
 - Listar os alunos que tenham média geral acima de 7.
 - Os departamentos que a média salarial seja maior que 3000.
 - Os professores que tenham turmas com mais de 5 alunos.

Having

- Essas condições não são aplicadas a somente uma tupla.
- É aplicada a cada grupo obtido através da cláusula GROUP BY.
- Para isso deve-se utilizar a cláusula HAVING.
- O predicado lógico é aplicado após os grupos terem sido formados.

Having

- Listar os alunos com média maior do que 7

Having

- Listar os alunos com média maior do que 7
SELECT c.mat_estudante, avg(c.nota) AS media
FROM cursa c
GROUP BY c.mat_estudante
HAVING avg(c.nota)>7

Having

- Liste o número de disciplinas por departamento, exceto o departamento 'DMA'

Having

- Liste o número de disciplinas por departamento, exceto o departamento 'DMA'
- **SELECT** depto_responsavel, count(*)
FROM disciplina
GROUP BY depto_responsavel
HAVING depto_responsavel != 'DMA'

Group by e having

- Sequencia de operações:
SELECT <lista de atributos>
FROM <lista de tabelas>
WHERE <predicado>
GROUP BY <lista de atributos>
HAVING <predicado>

Agregação e valores NULL

- Os valores NULL podem dificultar as agregações.
 - Por exemplo, alunos pode ter nota igual a NULL, como deve ser feito o cálculo da média?
- Em geral, as funções de agregação seguem uma regra:
 - Todas as funções, exceto a **count** desconsideram os nulos.
 - Count de uma tabela vazia é igual a 0.
 - As demais operações numa tabela vazia retornam NULL.

Leitura recomendada

ELMASRI, R; NAVATHE, S.B. **Sistemas de Banco de Dados**, Addison Wesley, 6ª Edição.

- Capítulo 4 e 5

Silberschatz, A; Korth H.F.; Sudarshan S. **Sistemas de Banco de Dados**, Editora Campus, 6ª Edição.

- Capítulo 3, 4 e 5

