



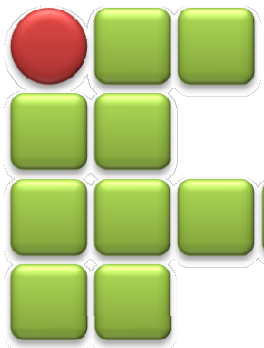
INSTITUTO FEDERAL
ESPÍRITO SANTO
Campus Colatina

BANCO DE DADOS I

BANCO DE DADOS I

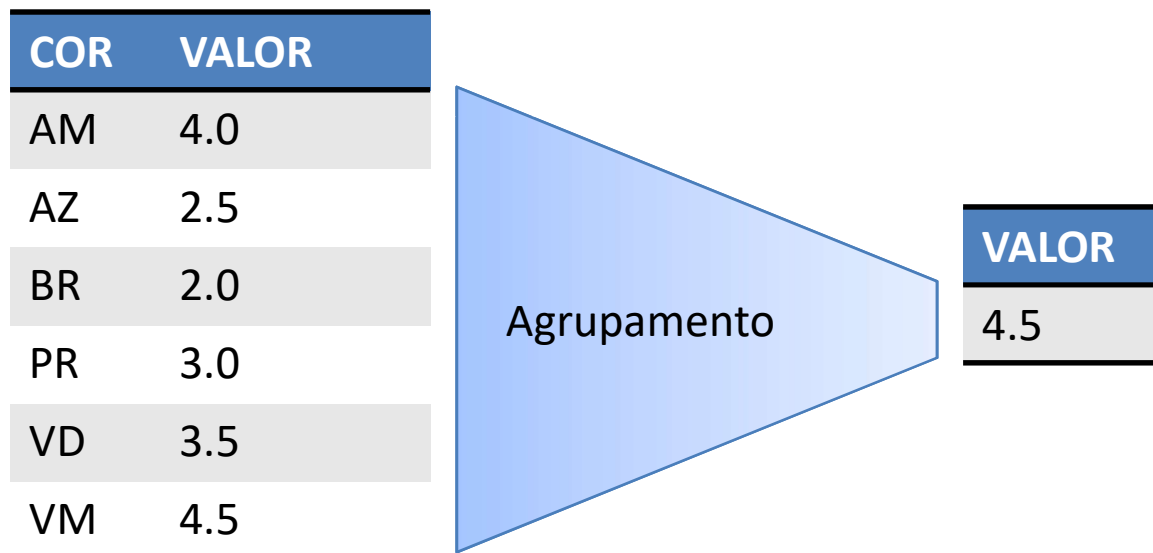
Agrupando e Resumindo Dados
(Funções de Grupo)

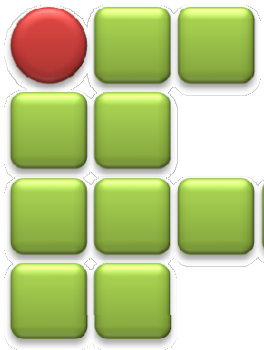
Prof. Gustavo Ludovico Guidoni



Funções de Grupo

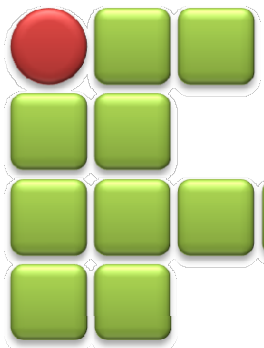
- Operam sobre um conjunto de linhas para fornecer um resultado por grupo





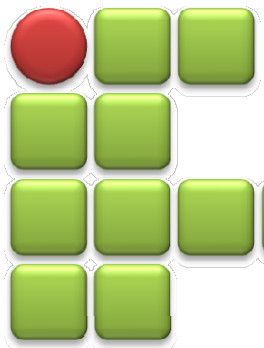
Funções de Grupo

- AVG
- SUM
- MIN
- MAX
- COUNT



Funções de Grupo - SUM e AVG

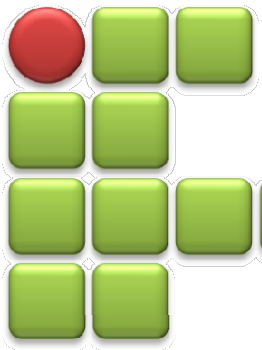
- **SUM(expressão)**
 - Retorna a soma de um conjunto de valores
- **AVG(expressão)**
 - Retorna a média aritmética de um conjunto de valores
- Podem ser utilizadas somente para dados numéricos



Funções de Grupo - SUM e AVG

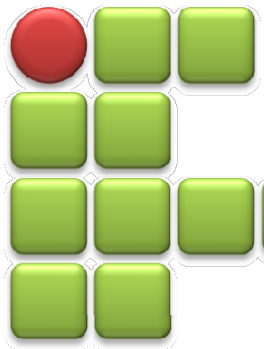
```
SELECT SUM(val_salario) 'Soma'  
      ,      AVG(val_salario) 'Média'  
      ,      COUNT(*) 'Qtd Funcionários'  
FROM    empregado  
WHERE   num_departamento = 20
```

Soma	Média	Qtd Funcionários
10875.00	2175.00	5



Funções de Grupo - MIN e MAX

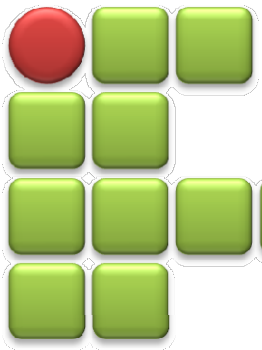
- **MIN(expressão)**
 - Retorna o valor mínimo de um conjunto de valores
- **MAX(expressão)**
 - Retorna o valor máximo de um conjunto de valores
- Podem ser utilizadas para quaisquer tipos de dados (exceto BIT)



Funções de Grupo - MIN e MAX

```
SELECT MIN(dat_locacao) 'Mínimo'  
      , MAX(dat_locacao) 'Máximo'  
FROM    locacao
```

Mínimo	Máximo
2003-01-20 00:00:00.000	2003-10-20 00:00:00.000

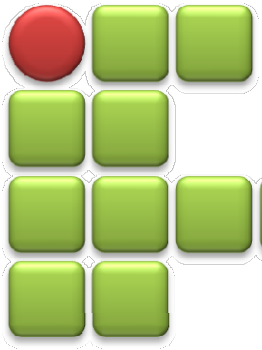


Funções de Grupo - COUNT

- COUNT(*)
 - Retorna o número total de linhas
- COUNT(expressão)
 - Retorna o número total de linhas **não nulas**

```
SELECT COUNT(*) 'Clientes MG'
, COUNT(ctl_cliente) 'Nº Cel. Registrados'
FROM cliente
WHERE estado = 'MG'
```

Clientes MG	Nº Cel. Registrados
5	3

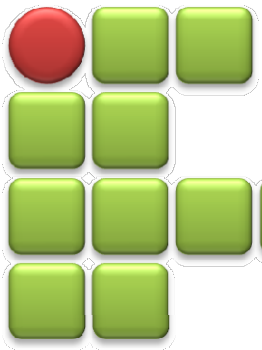


Funções de Grupo – e Valores Nulos

- As funções de grupo, **exceto COUNT(*)**, ignoram valores nulos na expressão.

```
SELECT AVG(val_comissao)  
FROM empregado
```

```
-----  
550.00
```

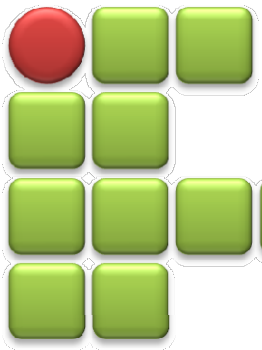


Identificando Valores Nulos

- Utilize a função **COALESCE** para que as funções de grupo considere valores nulos nos cálculos

```
SELECT AVG( COALESCE( val_comissao, 0) )  
FROM    empregado
```

```
-----  
157.1428
```

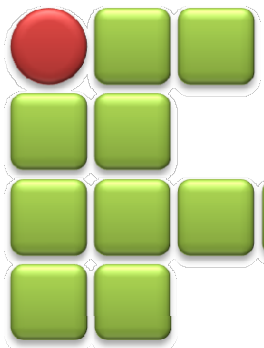


DISTINCT e ALL

- `função_grupo([ALL] expressão)`
 - Considera todos os valores nos cálculos, inclusive os repetidos. Esta é a condição padrão
- `função_grupo(DISTINCT expressão)`
 - Considera somente valores distintos nos cálculos

```
SELECT COUNT(ALL estado)      Todos
,      COUNT(DISTINCT estado) Distintos
FROM    cliente
```

Todos	Distintos
11	3

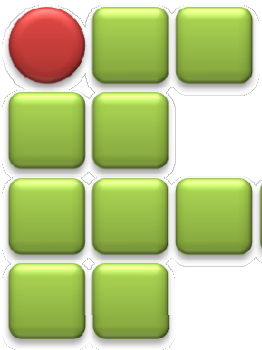


Criando Grupos de Dados

ESTADO	NOME
MG	João da Silva
MG	Maria José de Souza
MG	Antônio Carlos
MG	Patrícia dos Santos
MG	Paulo Correia
RJ	Daniela Marinho
RJ	Fabília Passos
RJ	Danilo Passos
SP	Roberto Assunção
SP	Sílvia Cordeiro
SP	Ricardo Andrade

Quantidade de
clientes por
estado

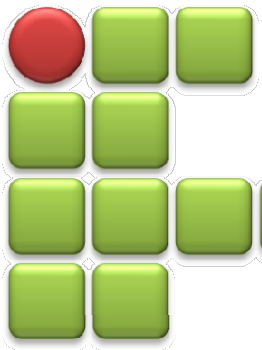
ESTADO	CLIENTES
MG	5
RJ	3
SP	3



A Cláusula GROUP BY

```
SELECT      [coluna,] função_grupo(expressão)
FROM        tabela
[WHERE      condição]
[GROUP BY  coluna]
[ORDER BY  coluna]
```

- A cláusula **GROUP BY** divide a tabela em grupos menores
 - Cria grupos no resultado exibido
 - Os grupos são criados de acordo do que é exibido

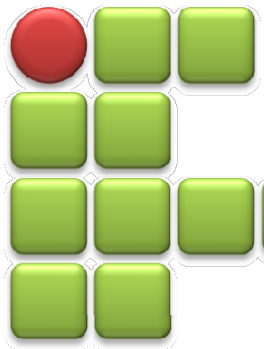


A Cláusula GROUP BY

- Todas as colunas na cláusula **SELECT** que não estiverem em funções de grupo devem estar na cláusula **GROUP BY**

```
SELECT estado  
      , COUNT(*) 'Nº de Clientes'  
FROM   cliente  
GROUP BY estado
```

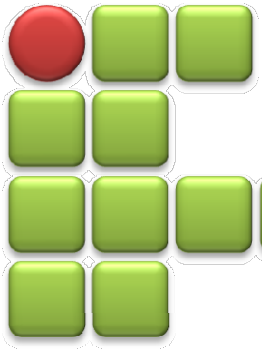
estado	Nº de Clientes
MG	5
RJ	3
SP	3



A Cláusula GROUP BY

```
SELECT num_departamento
,      AVG(val_salario) 'Média salarial'
FROM    empregado
GROUP BY num_departamento
```

num_departamento	Média salarial
10	2916.66
20	2175.00
30	1566.66

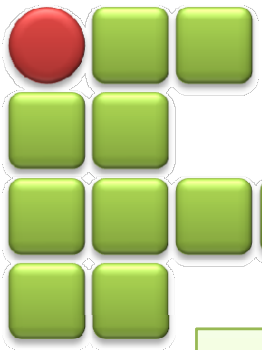


A Cláusula GROUP BY

- As colunas que estiverem na cláusula GROUP BY não precisam estar na cláusula SELECT

```
SELECT  MIN(dat_locacao)
FROM    locacao
GROUP BY cod_cliente
```

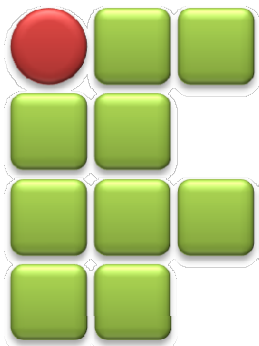
```
-----
2003-01-20 00:00:00.000
2003-03-22 00:00:00.000
2003-10-05 00:00:00.000
2003-10-10 00:00:00.000
2003-01-20 00:00:00.000
2003-01-20 00:00:00.000
```

Agrupando por Mais de uma Coluna

```
SELECT uf_cliente  
      , cid_cliente  
      , COUNT(*)  
FROM   cliente  
GROUP BY uf_cliente, cid_cliente  
ORDER BY uf_cliente, cid_cliente
```

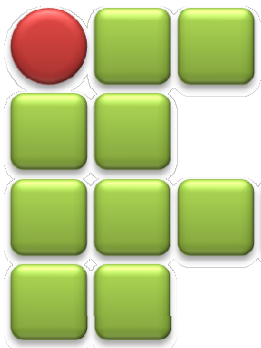
uf_cliente	cid_cliente	
MG	Belo Horizonte	3
MG	Sete Lagoas	2
RJ	Rio de Janeiro	3
SP	Jacareí	1
SP	São Paulo	2



A Cláusula HAVING: Excluindo Resultados do Grupo

- Não é possível utilizar a cláusula **WHERE** para restringir dados do resultado do agrupamento
 - Utilize a cláusula **HAVING** para tal

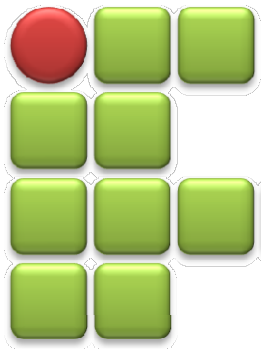
```
SELECT num_departamento  
      , SUM(val_salario)  
FROM   emp  
WHERE COUNT(*) > 3  
GROUP BY num_departamento
```



A Cláusula HAVING: Excluindo Resultados do Grupo

- Utilize a cláusula **HAVING** para restringir grupos:
 - As linhas são agrupadas
 - A função de grupo é aplicada
 - Os grupos que correspondem à cláusula HAVING são exibidos

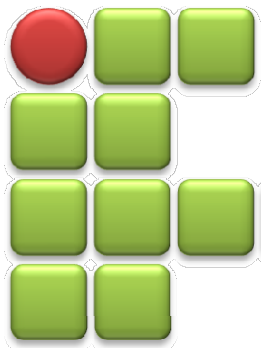
```
SELECT [coluna,] função_grupo(expressão)
FROM   tabela
[WHERE condição]
[GROUP BY coluna]
[HAVING condição_grupo]
[ORDER BY coluna]
```



A Cláusula HAVING: Excluindo Resultados do Grupo

```
SELECT num_departamento  
      , SUM(val_salario)  
FROM   empregado  
GROUP BY num_departamento  
HAVING SUM(val_salario) > 9000
```

num_departamento	
20	10875.00
30	9400.00



A Cláusula HAVING: Excluindo Resultados do Grupo

```
SELECT cargo
      ,   AVG(val_salario) 'Média'
FROM   empregado
WHERE  cargo <> 'GERENTE'
GROUP BY cargo
HAVING COUNT(*) > 2
```

cargo	Média
-----	-----
BALCONISTA	1037.50
VENDEDOR	1400.00

