

# Agrupando dados - GROUP BY

# Estrutura do comando SELECT

SELECT <atributo 1, atributo 2, ..., atributo N> ← **Colunas**

FROM <tabela1> ← **Origem de dados**  
JOIN <tabela 1> ON <Condições>

WHERE <condição 1> AND (<condição 2> OR <condição 3>) ← **Filtro de linhas**

GROUP BY <todos os atributos não agrupados listados no SELECT> ← **Agrupamento**

ORDER BY <atributos listados no SELECT> ← **Ordenação**

# Funções de agrupamento

## Funções para calcular dados agrupados

- COUNT – quantidade
- SUM – soma
- MIN – menor valor
- MAX – maior valor
- AVG – média dos valores

# Funções de agrupamento

```
SELECT COUNT(*) AS QTD,  
       SUM(val_salario) AS Total,  
       MIN(val_salario) AS Menor,  
       MAX(val_salario) AS Maior,  
       AVG(val_salario) AS Media  
FROM empregado
```

	QTD	Total	Menor	Maior	Media
1	9	17300.00	500.00	4500.00	1922.222222

# Agrupando dados

```
SELECT sig_uf,  
       COUNT(*) AS QTD,  
       SUM(val_salario) AS Total,  
       MIN(val_salario) AS Menor,  
       MAX(val_salario) AS Maior,  
       AVG(val_salario) AS Media  
FROM empregado  
GROUP BY sig_uf  
ORDER BY 3
```

	sig_uf	QTD	Total	Menor	Maior	Media
1	RJ	1	1700.00	1700.00	1700.00	1700.000000
2	MG	3	5600.00	800.00	2800.00	1866.666666
3	SP	5	10000.00	500.00	4500.00	2000.000000

# HAVING

Comando de filtro para as funções agregadas

HAVING x WHERE

- **HAVING** filtra o resultado de funções agregadas e **WHERE** filtra linhas de uma tabela

# Estrutura do comando SELECT

SELECT <atributo 1, atributo 2, ..., atributo N> ← Colunas  
FROM <tabelas> ← Origem de dados  
WHERE <condição 1> AND (<condição 2> OR <condição 3>) ← Filtro de linhas  
GROUP BY <todos os atributos não agrupados  
listados no SELECT> ← Agrupamento  
HAVING <condições de agrupamento> ← Filtros dados agrupados  
ORDER BY <atributos listados no SELECT> ← Ordenação

# Usando HAVING

```
SELECT sig_uf,  
       COUNT(*) AS QTD,  
       SUM(val_salario) AS Total,  
       MIN(val_salario) AS Menor,  
       MAX(val_salario) AS Maior,  
       AVG(val_salario) AS Media  
FROM empregado  
GROUP BY sig_uf  
HAVING MIN(val_salario) < 1000  
ORDER BY 3
```

	sig_uf	QTD	Total	Menor	Maior	Media
1	MG	3	5600.00	800.00	2800.00	1866.666666
2	SP	5	10000.00	500.00	4500.00	2000.000000



# HAVING

Listar a média e a soma dos salários por departamento, quando a soma for maior que 2.500,00

```
SELECT e.cod_depto, AVG(e.val_salario) AS media, SUM(e.val_salario) AS soma  
FROM empregado e  
GROUP BY e.cod_depto  
HAVING SUM(e.val_salario) > 2500.00
```

	cod_depto	media	soma
1	NULL	500.000000	500.00
2	1	2200.000000	8800.00
3	2	2250.000000	4500.00
4	3	1500.000000	1500.00
5	4	2000.000000	2000.00



	cod_depto	media	soma
1	1	2200.000000	8800.00
2	2	2250.000000	4500.00

# HAVING

Listar a média e a soma dos salários por departamento, quando a soma for maior que 2.500,00 e a média maior que 2.200,00

```
SELECT e.cod_depto, AVG(e.val_salario) AS media, SUM(e.val_salario) AS soma FROM  
empregado e  
GROUP BY e.cod_depto  
HAVING SUM(e.val_salario) > 2500.00  
AND AVG(e.val_salario) > 2200.00
```

	cod_depto	media	soma
1	2	2250.000000	4500.00

# HAVING

Listar os departamentos com 2 ou mais empregados

```
SELECT e.cod_depto, d.nom_depto, COUNT(*) AS total_func  
FROM empregado e
```

```
INNER JOIN departamento d ON e.cod_depto = d.cod_depto  
GROUP BY e.cod_depto, d.nom_depto
```

```
HAVING COUNT(*) >= 2
```

	cod_depto	nom_depto	total_func
1	1	Informática	4
2	2	Vendas	2
3	3	Compras	1
4	4	RH	1



	cod_depto	nom_depto	total_func
1	1	Informática	4
2	2	Vendas	2



**PUC Minas**  
**Virtual**