

# Consulta Baseada em Funções

Prof. Msc. Luã Marcelo Muriana

Engenharia de Software– 2024/01

Experimentos Práticos de Banco de Dados



# Na aula anterior...

- Exibir os resultados de modo agregado:

```
SELECT função(atributo)
FROM tabelas
WHERE condições
GROUP BY atributo
HAVING condição
```

- Funções:

MIN

MAX

COUNT

SUM

AVG

# PROBLEMA



## APRESENTADORES

Id	apresentador	salario	horas_TV
1	Luciano Huck	100000	4,5
2	Ana maria	80000	22
3	Louro José	30000	20
4	Silvio Santos	200000.85	11
5	XuxA	5500.50	5
6	Angélica	35000	1

Suponha que você queira exibir os resultados em ordem alfabética ou em ordem crescente de salário.

## Como resolver?

# Ordenação dos Resultados – ORDER BY

- O comando ORDER BY exibe os resultado da consulta de dados de forma ordenada.
- Para isso, basta que se acrescente a instrução ORDER BY e a coluna ou relação de colunas.

## Sintaxe básica

```
SELECT atributo(s)  
FROM lista_de_tabelas  
WHERE condições  
ORDER BY atributo(s)
```

## DICA

Por padrão a ordenação é em ordem ascendente (do menor para o maior). Se você quiser ordenar em ordem descendente (do maior para o menor), adicione a comando DESC após o atributo.

```
ORDER BY atributo DESC
```



# Ordenação dos Resultados – ORDER BY

## Alunos

id	Nome	Sexo	Idade
1	Harry	M	35
2	Luna	F	33
3	Rony	M	34
4	Dumbledore	M	150
5	Snape	M	70
6	Hermione	F	33

Quais são os alunos do sexo feminino? Exiba os resultados em ordem alfabética.

```
SELECT nome  
FROM ALUNOS  
WHERE sexo = 'F'  
ORDER BY nome;
```

id	Nome	Sexo	Idade
2	Hermione	F	33
6	Luna	F	33

Quais são os alunos do sexo masculino? Exiba os resultados em ordem decrescente de idade.

```
SELECT *  
FROM ALUNOS  
WHERE sexo = 'M'  
ORDER BY idade DESC;
```

id	Nome	Sexo	Idade
4	Dumbledore	M	150
5	Snape	M	70
1	Harry	M	35
3	Rony	M	34

# PROBLEMAS

## APRESENTADORES

Id	apresentador	salario	horas_TV
1	luciano Huck	100000	4,468
2	Ana maria	80000	21,8546
3	louro josé	30000	20,45
4	silvio santos	200000.85	10,965
5	xuXA	5500.50	5,23
6	anGéLica	35000	1,2354

Você deseja  
padronizar (formatar)  
a saída dos nomes.  
Ex.: todas as letras  
maiúsculas, ou todas  
minúsculas....

Você deseja  
colocar o \$ para  
exibir o salário  
Ex.: \$100.000,00

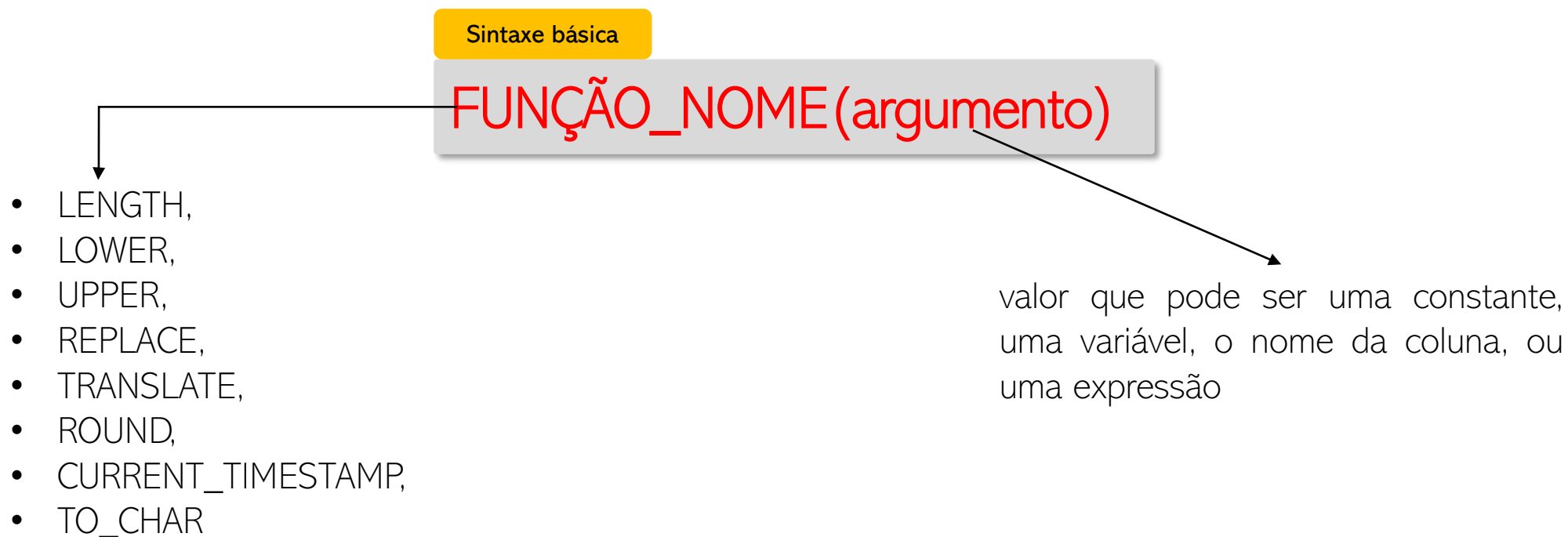
Você deseja  
arredondar os  
números.  
Ex.: 10,965 para 11

## Como resolver?



# Consulta Baseada em Funções

- As funções SQL, assim como em outras linguagens, recebem e processam argumentos, retornando um determinado resultado ao ambiente de chamada, de acordo com a solicitação.



# Função LENGTH

- A função LENGTH exibe a quantidade de caracteres de um argumento.

## Sintaxe básica

```
SELECT LENGTH (atributo)
FROM tabelas
WHERE condições;
```

Alunos

id	Nome	Sexo	Idade
1	Harry	M	35
2	Hermione	F	33
3	Rony	M	34
4	Dumbledore	M	150
5	Snape	M	70
6	Luna	F	33

Qual a quantidade de caracteres do nome de cada um dos alunos?

```
SELECT nome, LENGTH (nome) as tamanho
FROM ALUNOS;
```

Nome	tamanho
Harry	5
Hermione	8
Rony	4
Dumbledore	10
Snape	5
Luna	4



# Função LOWER

- A função LOWER converte o argumento para letras minúsculas.

## Sintaxe básica

```
SELECT LOWER (atributo)
FROM tabelas
WHERE condições;
```

Alunos

id	Nome	Sexo	Idade
1	HARRY	M	35
2	HERMIONE	F	33
3	RONY	M	34
4	DUMBLEDORE	M	150
5	SNAPE	M	70
6	LUNA	F	33

```
SELECT LOWER (nome) as nome_min
FROM ALUNOS;
```

nome_min
harry
hermione
rony
dumbledore
snapes
luna

# Função UPPER

- A função UPPER converte uma cadeia de caracteres em letras maiúsculas

## Sintaxe básica

```
SELECT UPPER (atributo)
FROM tabelas
WHERE condições;
```

Alunos

id	Nome	Sexo	Idade
1	harry	M	35
2	hermione	F	33
3	rony	M	34
4	dumbledore	M	150
5	snape	M	70
6	luna	F	33

```
SELECT UPPER (nome)
FROM ALUNOS;
```

Nome
HARRY
HERMIONE
RONY
DUMBLEDORE
SNAPE
LUNA

# Função INITCAP

- A função INITCAP converte as letras iniciais de um conjunto de caracteres em letras maiúsculas.

## Sintaxe básica

```
SELECT INITCAP (atributo)
FROM tabelas
WHERE condições;
```

Alunos

id	Nome	Sexo	Idade
1	harry potter	M	35
2	hermione granger	F	33
3	rony weasley	M	34
4	alvo Dumbledore	M	150
5	severo snape	M	70
6	luna	F	33

```
SELECT INITCAP (nome)
FROM ALUNOS;
```

Nome
Harry Potter
Hermione Granger
Rony Weasley
Alvo Dumbledore
Severo Snape
Luna

# Função REPLACE

- A função REPLACE pesquisa em argumento1 o valor de argumento 2 e substitui pelo argumento3.

## Sintaxe básica

```
SELECT REPLACE (arg1,  
'arg2', 'arg3')  
FROM tabelas  
WHERE condições;
```

## Alunos

id	Nome	Sexo	Idade
1	Harry Potter	M	35
2	Hermione Granger	F	33
3	Rony Weasley	M	34
4	Alvo Dumbledore	M	150
5	Severo Snape	M	70
6	Luna	F	33

```
SELECT nome,  
       REPLACE (nome, 'on', 'XW')  
       AS nome_replace  
FROM ALUNOS;
```

Nome	nome_replace
Harry Potter	Harry Potter
Hermione Granger	HermiXWe Granger
Rony Weasley	RXWy Weasley
Alvo Dumbledore	Alvo Dumbledore
Severo Snape	Severo Snape
Luna	Luna

# Função ROUND

- A função Round arredonda os valores seguindo as regras de maior ou igual a 5 ele arredonda para cima e menor que 5 ele arredonda para baixo.

## Sintaxe básica

```
SELECT ROUND(atributo)
FROM tabelas
WHERE condições;
```

Alunos

id	Nome	Sexo	Horas_estudo
1	Harry Potter	M	10.486
2	Hermione Granger	F	25.52
3	Rony Weasley	M	5.123
4	Alvo Dumbledore	M	60.4585
5	Severo Snape	M	55.10
6	Luna	F	20.364

```
SELECT nome,
       ROUND(horas_estudo, 1)
FROM ALUNOS;
```

Indica o número de casas decimais. Se não colocar nenhum valor, o dado será arredondado para um valor inteiro.



Nome	Horas_estudo
Harry Potter	10.5
Hermione Granger	25.2
Rony Weasley	5.1
Alvo Dumbledore	60.5
Severo Snape	55.1
Luna	20.4

# Função TO\_CHAR para conversão de números

- A função TO\_CHAR (argumento1, argumento2) converte o valor (data ou número) representado em argumento1, para uma cadeia de caracteres com um formato predeterminado, representado por argumento2.

## Sintaxe básica

```
SELECT  
TO_CHAR(arg1, arg2)  
FROM tabelas  
WHERE condições;
```

Elemento	Descrição	Formato	Resultado
\$	Exibir o cifrão do lado esquerdo do valor	\$99999	\$1234
,	Separador de milhar	9999,999	1,234
.	Separador de decimal	99999.99	1234.00
0	Inserir zeros à esquerda do número para completar o valor que contiver menos caracteres que o especificado	099999	01234
9	Inserir "espaços em branco" à esquerda do número que contiver menos caracteres que o especificado	99999	1234

# Função TO\_CHAR para conversão de números

## Sintaxe básica

```
SELECT TO_CHAR(arg1, arg2)
FROM tabelas
WHERE condições;
```

Elemento	Descrição	Formato	Resultado
\$	Exibir o cifrão do lado esquerdo do valor	\$99999	\$1234
,	Separador de milhar	9999,999	1,234
.	Separador de decimal	99999.99	1234.00
0	Inserir zeros à esquerda do número para completar o valor que contiver menos caracteres que o especificado	099999	01234
9	Inserir "espaços em branco" à esquerda do número que contiver menos caracteres que o especificado	99999	1234

## APRESENTADORES

Id	apresentador	salario
1	luciano Huck	100000
2	Ana maria	80000
3	louro josé	30000
4	silvio santos	200000.85
5	xuXA	5500.50
6	anGéLica	35000

```
SELECT nome,
       TO_CHAR(salario, '$999,999.99')
FROM ALUNOS;
```

O número de '9' indica a quantidade máxima de dígitos.



apresentador	salario
luciano Huck	\$100,000.00
Ana maria	\$800,00.00
louro josé	\$30,000.00
silvio santos	\$200,000.85
xuXA	\$5,500.50
anGéLica	\$35,000.00

# Função TO\_CHAR para conversão de datas

- A função TO\_CHAR também possibilita converter datas em caracteres e, assim, como com números, a função torna-se interessante para algum tipo de manipulação que requeira a apresentação de datas em formatos específicos.

## Sintaxe básica

```
SELECT  
TO_CHAR(arg1, arg2)  
FROM tabelas  
WHERE condições;
```

Formato	Descrição	Resultado
DD	Dia do mês	23
DAY	Nome do dia da semana	SEGUNDA
MM	Número do mês	05
YY	Ano com dois dígitos	22
HH:MI:SS	Hora, minuto e segundo	08:00:00



# Função TO\_CHAR para conversão de números

## Sintaxe básica

```
SELECT TO_CHAR(arg1, arg2)
FROM tabelas
WHERE condições;
```

Formato	Descrição	Resultado
DD	Dia do mês	23
DAY	Nome do dia da semana	SEGUNDA
MM	Número do mês	05
YY	Ano com dois dígitos	22
HH:MI:SS	Hora, minuto e segundo	08:00:00

Exibir a data da fatura do cliente 99132 de acordo com o formato de **São Paulo, DD do mês, ano**

```
SELECT TO_CHAR(data_vencimento, '"São Paulo," DD "de" month "do ano de " YY') AS "DATA FORMATADA"
FROM tabela_fatura
WHERE codigo_cliente = 99132;
```

## DATA FORMATADA

São Paulo, 23 de maio do ano de 22

# ATIVIDADE

Nome da Conexão: < um nome a sua escolha >  
Nome do usuário: <usuário> Senha: \*\*\*\*\*  
Tipo de conexão: Básica Atribuição: Padrão  
Nome do host ou String do Host: BD-ACD  
Porta: 1521  
SID: XE

No SQL Developer

Execute os códigos de DDL e DML disponíveis no  
módulo da aula do dia 29/04 disponível no  
Canvas (APA2\_POSTAGENS\_RS)

No SQL Developer

# Resolva os itens a seguir considerando a **tabela**

## **APA2\_POSTAGENS\_RS**

- 1) Exiba os usuários que fizeram postagem em ordem alfabética. (não repetir o dado)
- 2) **Quantas postagens cada usuário fez? Exiba os resultados em ordem decrescente baseado no número de postagens.**
- 3) Quantas vezes cada hashtag foi utilizada? Exiba os resultados em decrescente.
- 4) **Quantas postagens de cada tipo cada usuário fez? Exiba os resultados em ordem crescente.**
- 5) Quantos likes cada usuário obteve com suas postagens? Ordene os resultados em ordem crescente do número de likes. Formate o número de saída para que a pontuação de milhagem apareça (ex.: 1,000).
- 6) **Qual a média de alcance das postagens realizadas? Exiba o resultado arredondando o valor com duas casas decimais.**
- 7) Para cada um dos usuários que fez uma postagem que teve mais de 1000 comentários, exiba o nome do usuário e a seguinte mensagem: "Postagem feita no dia <dia> (<dia\_da\_semana>) de <mês> de <ANO>. Ex.: Postagem feita dia 15 de maio de 2023.
- 8) **Exiba todas as postagens feitas cuja localização tem 15 ou mais caracteres.**
- 9) Quantas postagens foram feitas em cada localização? Ordene o resultado em ordem alfabética.
- 10) **Quantos likes as postagens de cada localização obtiveram? Ordene o resultado em ordem decrescente de likes. Formate o número para que a pontuação de milhagem apareça.**

# Referência

Essa aula foi baseada nos slides do Prof. MSc. Isaias de Queiroz Ramos (PUC Campinas).