

Apresentação de Modelo Relacional |

Relacionamentos- Comandos DML



DEVinHouse

Parcerias para desenvolver a sua carreira

SENAI

<LAB365>

Considerações Iniciais

- Nosso horário é das **19:00 às 22:00 (10 min tolerância de chegada)**
- Intervalo às **20:30 de 20 minutos**
- Interaja na aula!
- Se tiver dúvidas, levanta a mão no chat e eu explico novamente
- Me corrija!!!!

Erros são comuns no mundo do Software

Eu posso errar. Corrija-me quando acontecer

Aceito feedbacks :D

- Façam as atividades práticas, só se aprende praticando!
- Qualquer coisa me procurem no **slack**



Agenda

- SQL JOINS
 - INNER JOIN
 - LEFT JOIN
 - RIGHT JOIN
 - OUTER JOIN

Pergunta

- Qual o retorno esperado da query abaixo?

```
select count(r.num_chamado), l.departamento
```

```
from registros r, login l
```

```
where r.data > '2020/01/01' AND l.departamento not like 'Tecnolo%' AND  
r.equipe = 'tecnologia' AND r.id = l.id group by l.departamento
```

banco de dados: sistema de chamados

tabela registros: histórico dos chamados

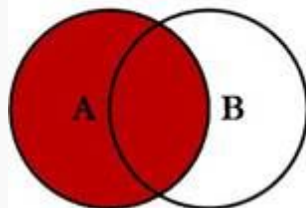
tabela login: dados dos usuários do sistema de chamados

Aula 03: SQL

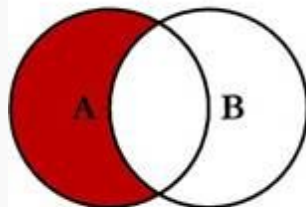
Vamos aprender sobre **SQL JOINS** e funções de agregações.

Além disso, vamos praticar a modelagem conceitual, lógica e física, usando algumas sugestões de banco de banco de dados.

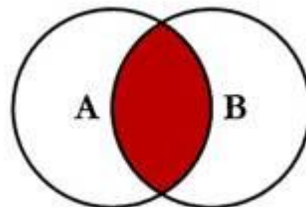
SQL JOINS



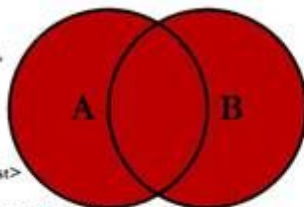
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
```



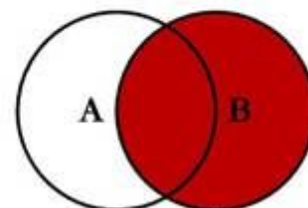
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
WHERE B.Key IS NULL
```



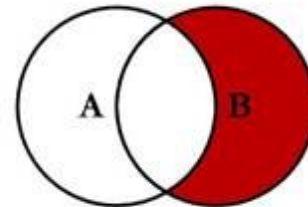
```
SELECT <select_list>
FROM TableA A
INNER JOIN TableB B
ON A.Key = B.Key
```



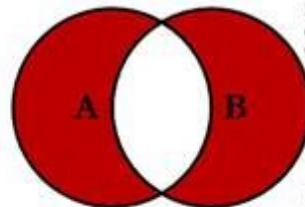
```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
```



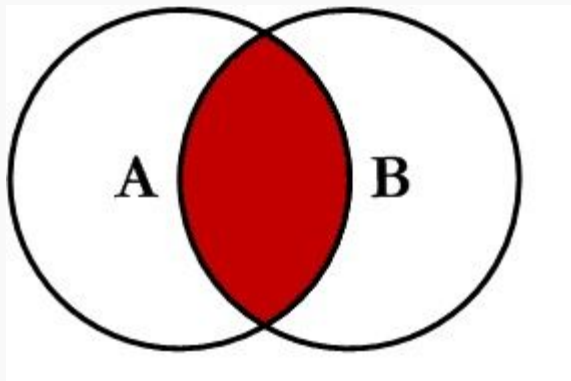
```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
OR B.Key IS NULL
```

SQL

Inner Join

- Retorna registros comuns às duas tabelas.

```
SELECT * FROM cliente as A  
INNER JOIN endereco as B ON A.codigo = B.codigo
```



Inner Join

Exemplo:

```
SELECT * FROM pessoas p  
INNER JOIN veiculos v ON p.modelo_veiculo = v.modelo;
```

id	nome	modelo_veiculo	id	marca	modelo	ano
1	1 Michael Nascimento	Palio	1	1 Honda	Civic	2020
2	2 João da Conceicao	Civic	2	2 Fiat	Palio	2013

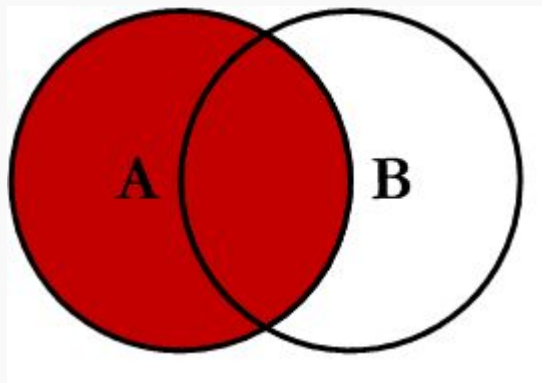
id	nome	modelo_veiculo	id	marca	modelo	ano
1	2 João da Conceicao	Civic	1	Honda	Civic	2020
2	1 Michael Nascimento	Palio	2	Fiat	Palio	2013

SQL

Left Join

- Retorna registros que estão na tabela A (cliente) e os registros comuns de B (endereço) e A.

```
SELECT * FROM cliente as A  
LEFT JOIN endereco as B ON cliente.codigo = endereco.codigo
```



Left Join

Exemplo:

```
SELECT * FROM veiculos v  
LEFT JOIN pessoas p ON p.modelo_veiculo = v.modelo;
```

	id	nome	modelo_veiculo
1	1	Michael Nascimento	Palio
2	2	João da Conceicao	Civic

	id	marca	modelo	ano
1	1	Honda	Civic	2020
2	2	Fiat	Palio	2013
3	3	Fiat	Doblo	2009

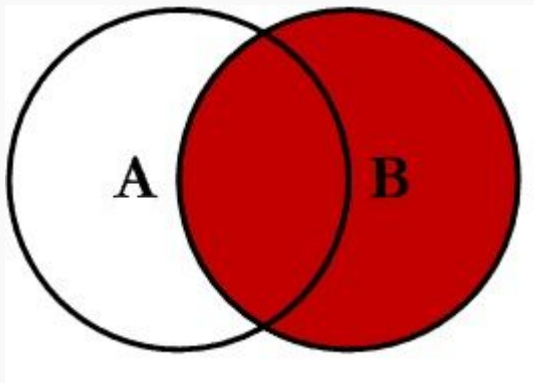
	id	marca	modelo	ano	id	nome	modelo_veiculo
1	1	Honda	Civic	2020	2	João da Conceicao	Civic
2	2	Fiat	Palio	2013	1	Michael Nascimento	Palio
3	3	Fiat	Doblo	2009	<null>	<null>	<null>

SQL

Right Join

- Retorna registros que estão na tabela B (endereco) e os registros comuns de A (cliente) e B.

```
SELECT * FROM cliente as A  
RIGHT JOIN endereco as B ON cliente.codigo = endereco.codigo
```

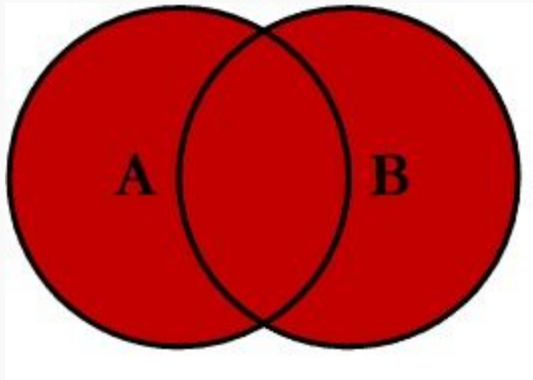


SQL

Outer Join

- Retorna todos registros que estão na tabela A (cliente) e todos os registros na tabela B (endereço).

```
SELECT * FROM cliente as A  
FULL OUTER JOIN endereco as B ON cliente.codigo = endereco.codigo
```

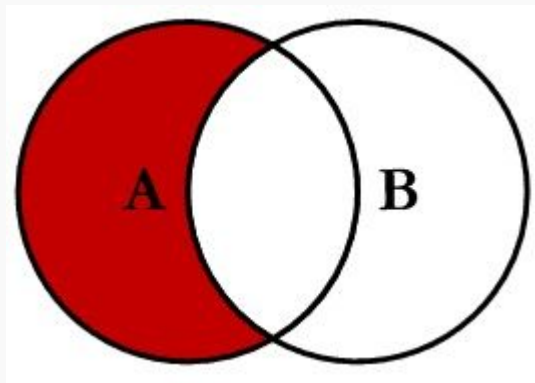


SQL

Left Excluding Join

- Retorna todos registros que estão na tabela A (cliente) e que não estejam na tabela B (endereço).

```
SELECT * FROM cliente as A  
LEFT JOIN endereco as B ON A.codigo = B.codigo  
WHERE B.codigo is null
```

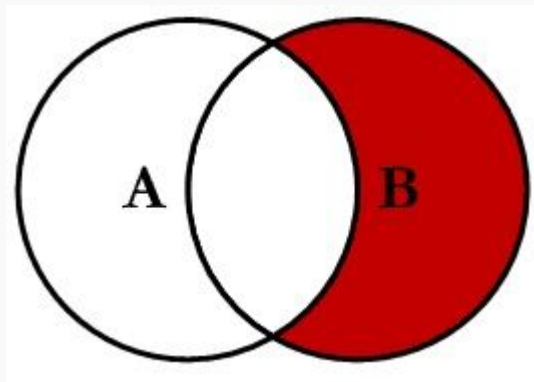


SQL

Right Excluding Join

- Retorna todos registros que estão na tabela B (endereco) e que não estejam na tabela A (cliente).

```
SELECT * FROM cliente as A  
RIGHT JOIN endereco as B ON cliente.codigo = endereco.codigo  
WHERE A.codigo is null
```

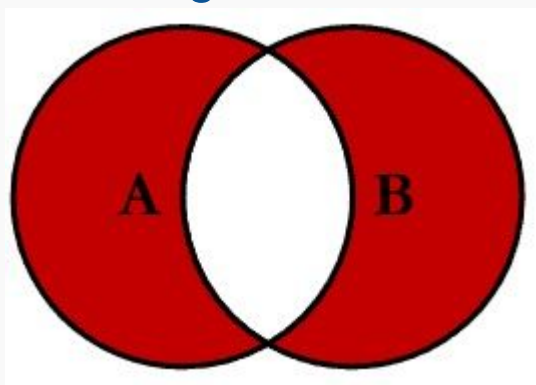


SQL

Outer Excluding Join

- Retorna todos registros que estão na tabela A (cliente) mas não estejam em B e todos os registros na tabela B (endereco) mas que não estejam em A.

```
SELECT * FROM cliente as A  
FULL OUTER JOIN endereco as B ON A.codigo = B.codigo  
WHERE A.codigo is null OR B.codigo is null
```



Relacionando com Subconsultas

- Tem vezes que precisamos relacionar uma tabela com uma consulta que gera dados intermediários. É possível usar todos os JOIN aprendidos com outras queries:

```
SELECT f.name country, c.name capital_city
FROM facts f
INNER JOIN (
    SELECT * FROM cities
    WHERE capital = 1
) c ON c.facts_id = f.id
LIMIT 10;
```


SQL

Funções de Agregação

SQL

Funções de Agregação

- Agrupam as linhas encontradas pela consulta e retorna apenas uma linha como resultado.

Vamos aprender as seguintes funções de agregação:

- 1) **AVG**: Calcula média aritmética;
- 2) **COUNT**: retorna o total de linha de acordo com a QUERY realizada;
- 3) **MAX**: retorna o registro de maior valor de acordo com a QUERY realizada;
- 4) **MIN**: retorna o registro de menor valor de acordo com a QUERY realizada;
- 5) **SUM**: retorna a soma de valores de acordo com a QUERY realizada.

SQL

Funções de Agregação AVG

- Argumento numérico;
- Retorno numérico decimal;
- Calcula a média aritmética sobre o conjunto de linhas;

```
SELECT AVG(nota) FROM Nota
```

SQL

Funções de Agregação COUNT

- Argumento qualquer;
- Retorno numérico inteiro;
- Retorna o número de linhas de entrada para o qual a expressão é não nula ou simplesmente o total caso seja especificado "*";

```
SELECT COUNT(*) FROM Cliente
```

SQL

Funções de Agregação MAX

- Argumento numérico, texto ou data/hora;
- Retorno mesmo tipo de dado do argumento;
- Retorna o maior valor entre todos os fornecidos;

```
SELECT MAX(nascimento) FROM Cliente
```

SQL

Funções de Agregação MIN

- Argumento numérico, texto ou data/hora;
- Retorno mesmo tipo de dado do argumento;
- Retorna o menor valor entre todos os fornecidos;

```
SELECT MIN(nascimento) FROM Cliente
```

SQL

Funções de Agregação SUM

- Argumento numérico;
- Retorno numérico inteiro ou decimal;
- Retorna o soma da expressão sobre as linhas de entrada;

```
SELECT SUM(valor) FROM Venda
```

AVALIAÇÃO DOCENTE

O que você está achando das minhas aulas neste conteúdo?

[Clique aqui](#) ou escaneie o QRCode ao lado para avaliar minha aula.

Sinta-se à vontade para fornecer uma avaliação sempre que achar necessário.





DEVinHouse

Parcerias para desenvolver a sua carreira

OBRIGADO!



<LAB365>