

# Microsoft SQL Server 2016 (T-SQL)

## Aula 09

Professor Anderson Henrique

## Assuntos tratados nessa aula:

01 – LIKE e NOT LIKE

02 – JOINS e INNER JOINS

03 – OUTER JOINS – LEFT e RIGHT

Professor Anderson Henrique



## 01 – LIKE e NOT LIKE

Determina se uma cadeia de caracteres específica corresponde a um padrão especificado. Um padrão pode incluir caracteres normais e curingas

NOT LIKE inverte a comparação, verificando se a cadeia de caracteres NÃO corresponde ao padrão especificado

Usado junto com WHERE:

WHERE coluna LIKE padrão

## Padrões Específicos

'%' - Qualquer cadeia de 0 ou mais caracteres

'\_' - Sublinhado: qualquer caracter único

'[]' - Qualquer caracter único no intervalo ou conjunto especificado ([a-h], [aeiou])

'[^]' - Qualquer caracter único que não esteja no intervalo ou conjunto especificado ([a-h], [aeiou])

```
SELECT Nome_Livro  
FROM tbl_Livro  
WHERE Nome_Livro LIKE 'S%';
```

```
SELECT Nome_Livro  
FROM tbl_Livro  
WHERE Nome_Livro LIKE '%g';
```

```
SELECT Nome_Livro  
FROM tbl_Livro  
WHERE Nome_Livro LIKE '_i%';
```

Professor Anderson Henrique

```
SELECT Nome_Livro  
FROM tbl_Livro  
WHERE Nome_Livro LIKE '[SL]%';
```

```
SELECT Nome_Livro  
FROM tbl_Livro  
WHERE Nome_Livro LIKE '%[gl]';
```

```
SELECT Nome_Livro  
FROM tbl_Livro  
WHERE Nome_Livro LIKE '_[is]%';
```

## 02 – JOINS e INNER JOINS

A palavra-chave JOIN é usada para obter dados provenientes de duas ou mais tabelas, baseado em um relacionamento entre colunas nestas tabelas

### JOINS – Tipos

- **INNER JOIN:** Retorna linhas quando houver pelo menos uma correspondência em ambas as tabelas
- **OUTER JOIN:** Retorna linhas mesmo quando não houver pelo menos uma correspondência em uma das tabelas (ou ambas). O OUTER JOINS divide-se em LEFT JOIN, RIGHT JOIN e FULL JOIN

Sintaxe:

```
SELECT colunas FROM tabela1  
INNER JOIN tabela2 ON tabela1.colunas = tabela2.coluna;
```

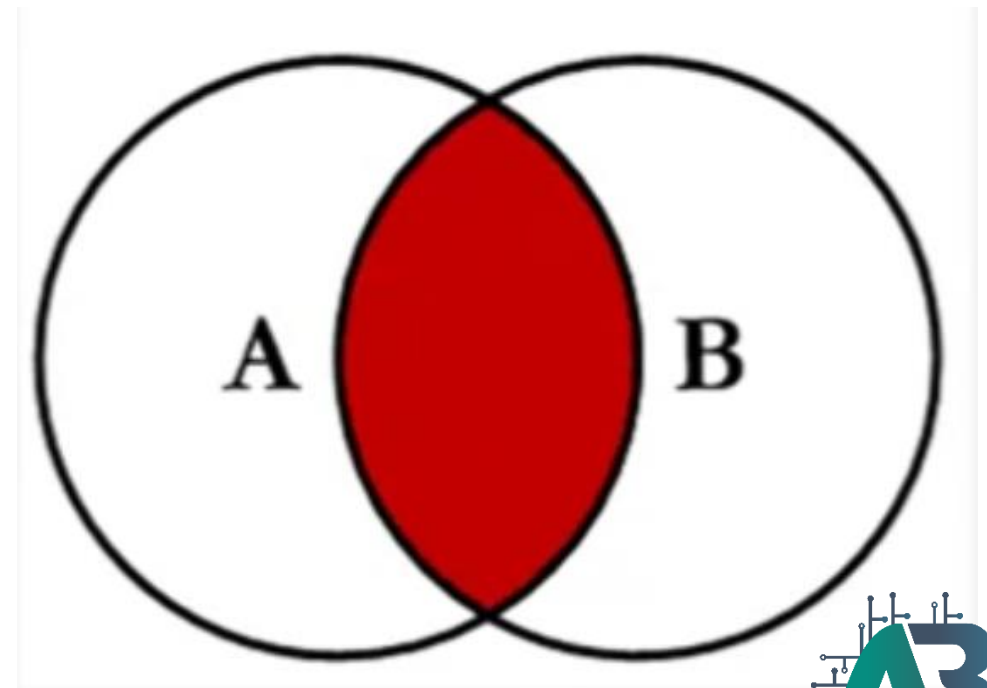
```
SELECT * FROM tbl_Livro  
INNER JOIN tbl_Autor ON tbl_Livro.ID_Autor = tbl_Autor.ID_Autor;
```

```
SELECT tbl_livro.Nome_Livro, tbl_Livro. ISBN_Livro, tbl_Autor.Nome_autor  
FROM  tbl_Livro  INNER  JOIN  tbl_Autor  ON  tbl_Livro.ID_Autor  =  
tbl_Autor.ID_Autor;
```



```
SELECT L.Nome_livro, E.Nome_Editora  
FROM tbl_Livro As L INNER JOIN tbl_Editora AS E  
ON L.ID_Editora = E.ID_Editora;
```

**INNER JOIN**



## 03 – OUTER JOINS

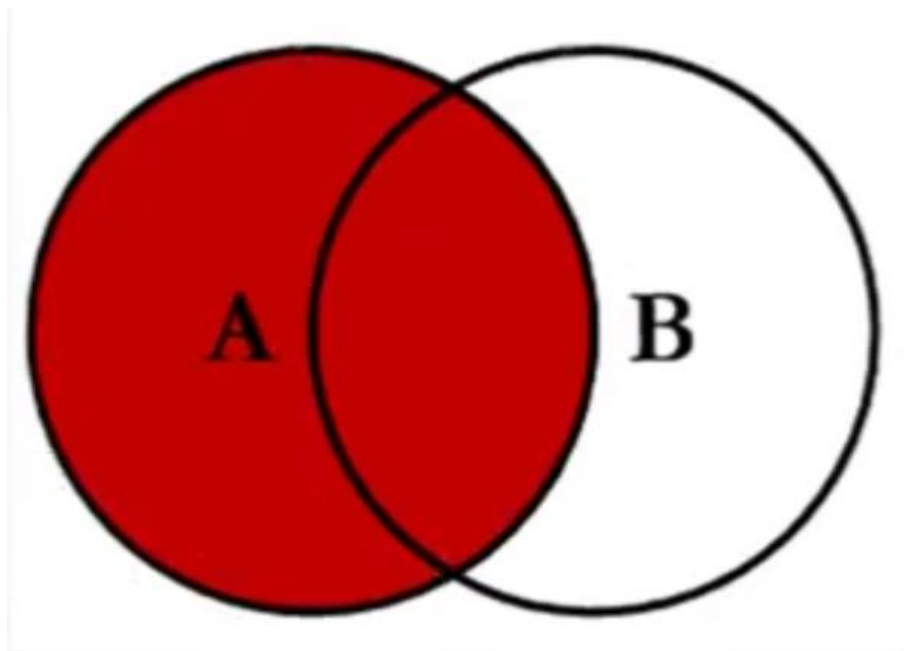
- **LEFT JOIN:** Retorna todas as linhas da tabela à esquerda, mesmo se não houver correspondência na tabela à direita
- **RIGHT JOIN:** Retorna todas as linhas da tabela à direita, mesmo se não houver nenhuma correspondência na tabela à esquerda
- **FULL JOIN:** Retorna linhas quando houver uma correspondência em qualquer uma das tabelas. É uma combinação de LEFT e RIGHT JOIN

Sintaxe:

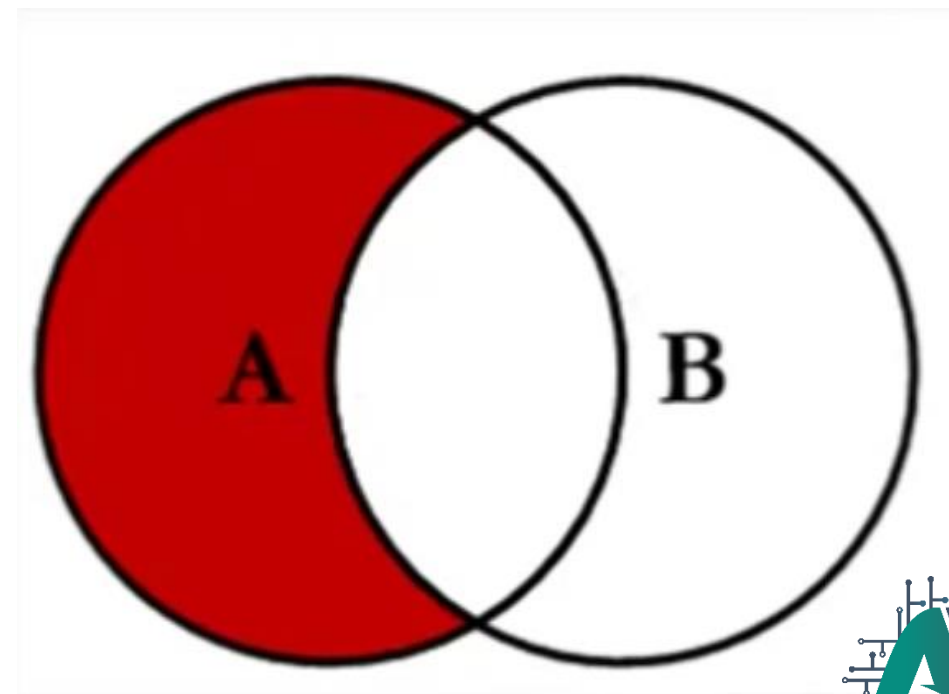
```
SELECT coluna  
FROM tabela_esq  
LEFT JOIN tabela_dir ON tabela_esq.coluna = tabela_dir.coluna;
```

```
SELECT * FROM tbl_Autor  
LEFT JOIN tbl_Livro  
ON tbl_Livro.ID_Autor = tbl_Autor.ID_Autor;
```

**LEFT JOIN**



**LEFT JOIN excluindo correspondências**

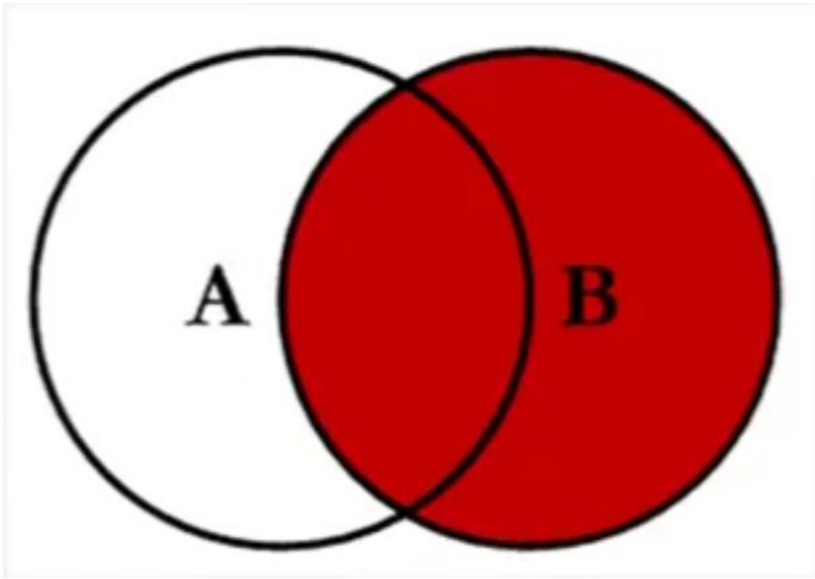


**TECH**  
soluções em ti

Sintaxe:

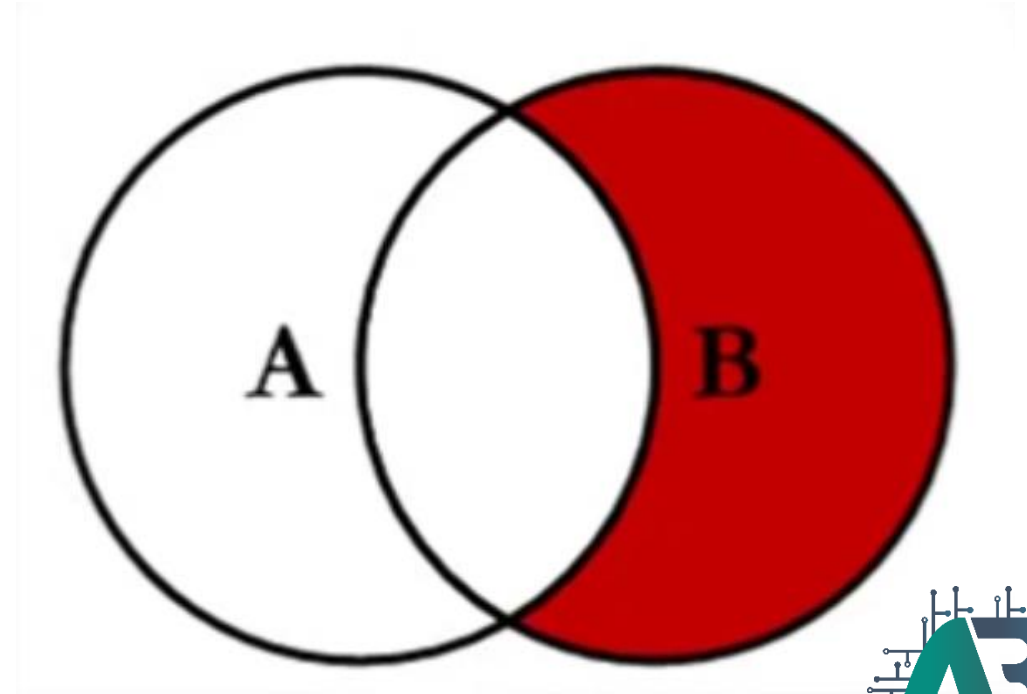
```
SELECT coluna  
FROM tabela_esq  
LEFT JOIN tabela_dir ON tabela_esq.coluna = tabela_dir.coluna  
WHERE tabela_dir.coluna IS NULL;
```

```
SELECT * FROM tbl_Autor  
LEFT JOIN tbl_Livro  
ON tbl_Livro.ID_Autor = tbl_Autor.ID_Autor  
WHERE tbl_Livro.ID_Autor IS NULL;
```



**RIGHT JOIN**

**RIGHT JOIN** excluindo correspondências



Sintaxe:

```
SELECT coluna  
FROM tabela_esq  
RIGHT JOIN tabela_dir ON tabela_esq.coluna = tabela_dir.coluna;
```

```
SELECT * FROM tbl_Livro AS Li  
RIGHT JOIN tbl_Editora AS Ed  
ON Li.ID_Editora = Ed.ID_Editora;
```

Sintaxe:

```
SELECT coluna  
FROM tabela_esq  
RIGHT JOIN tabela_dir ON tabela_esq.coluna = tabela_dir.coluna  
WHERE tabela_esq.coluna IS NULL;
```

```
SELECT * FROM tbl_Livro AS Li  
RIGHT JOIN tbl_Editora AS Ed  
ON Li.ID_Editora = Ed.ID_Editora  
WHERE Li.ID_Editora IS NULL;
```



**Dúvidas?**

**Professor Anderson Henrique**



## Para a próxima aula

01 – FULL JOIN (Selecionar dados de várias tabelas)

02 – IN e NOT IN (Filtros de Múltiplas Condições)

03 – OUTER JOINS – LEFT e RIGHT (Selecionar dados de várias tabelas)

Professor Anderson Henrique

