

# A17. DQL: ASSOCIAÇÃO DE TABELAS (INNER JOIN E OUTER JOIN)

PROF. WILLIAM C. AUGUSTONELLI (BILLY)

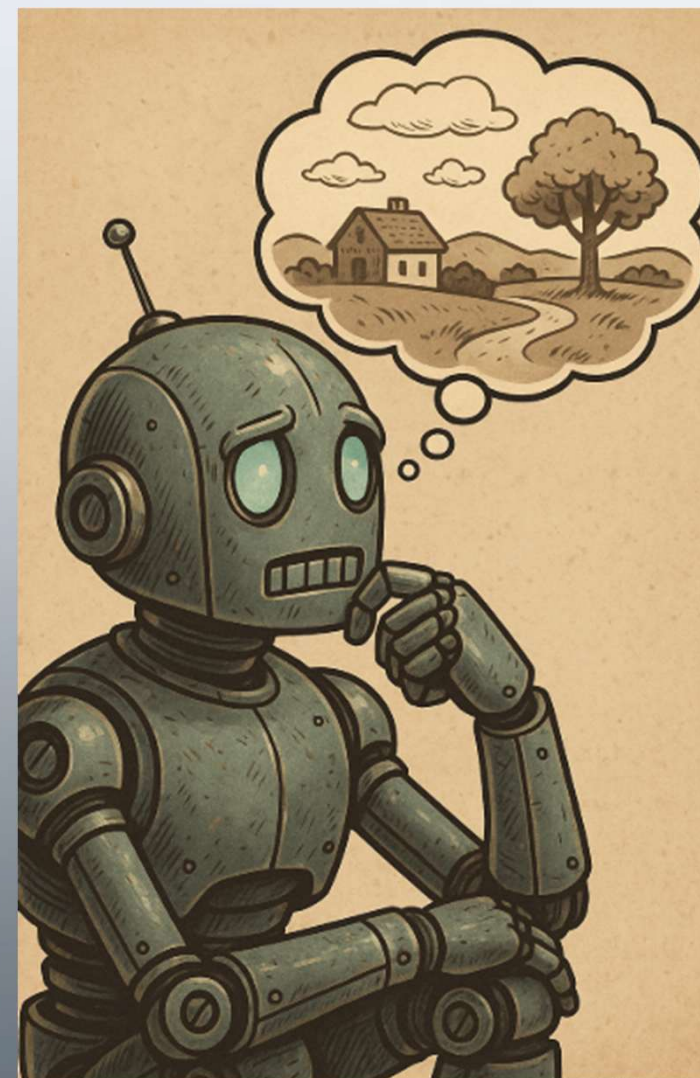
[WILLIAM.AUGUSTONELLI@DOCENTE.SENAI.BR](mailto:WILLIAM.AUGUSTONELLI@DOCENTE.SENAI.BR) – 2S2025

## OBJETIVO

- Compreender os conceitos de **junção entre tabelas** no modelo relacional
- Aplicar **INNER JOIN** e **OUTER JOIN** em consultas SQL
- Relacionar tabelas de forma eficiente utilizando chaves primárias e estrangeiras
- Interpretar os resultados e identificar quando usar cada tipo de JOIN

## NA ÚLTIMA AULA...

- União de Dados
  - Sintaxe do UNION e UNION ALL
  - Diferença entre UNION e UNION ALL
  - Regras para usar UNION
- Associação de Tabelas
  - Associação implícita com WHERE
  - CROSS JOIN
  - Diferença entre CROSS JOIN e INNER JOIN



# NOSSA AULA DE HOJE...

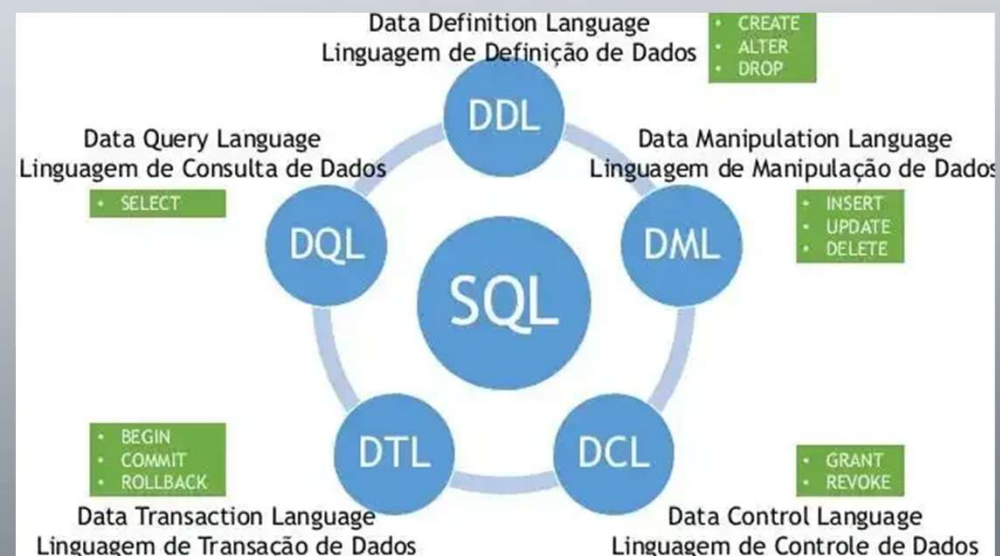
- INNER JOIN

- Conceito e funcionamento
- Sintaxe padrão SQL
- Exemplo prático com duas e três tabelas
- Comparação entre junção via WHERE

- OUTER JOIN

- Conceito geral
- Tipos: LEFT OUTER JOIN, RIGHT OUTER JOIN
- Diferenças e exemplos práticos
- Situações em que o OUTER JOIN é mais adequado

- Demonstração prática



# CONCEITO DE JOIN

- **JOIN** serve para **combinar dados** de duas ou mais tabelas
- Baseia-se em **valores relacionados** (geralmente por **chaves**)
- Tipos principais
  - INNER JOIN
  - LEFT (OUTER) JOIN
  - RIGHT (OUTER) JOIN
  - Existem outros: FULL, CROSS, SELF

# INNER JOIN

- Retorna apenas os registros que têm correspondência nas duas tabelas

```
SELECT aluno.nome, curso.nome  
FROM aluno  
INNER JOIN curso  
ON aluno.id_curso = curso.id;
```

- É o tipo de JOIN mais usado. Filtra os registros comuns.

# INNER JOIN COM TRÊS TABELAS

```
SELECT aluno.nome AS Aluno,  
       curso.nome AS Curso,  
       professor.nome AS Professor  
FROM aluno  
INNER JOIN curso ON aluno.id_curso = curso.id  
INNER JOIN professor ON curso.id_professor = professor.id;
```

# INNER JOIN COM TRÊS TABELAS

```
SELECT aluno.nome AS Aluno,  
       curso.nome AS Curso,  
       professor.nome AS Professor  
FROM aluno  
INNER JOIN curso ON aluno.id_curso = curso.id  
INNER JOIN professor ON curso.id_professor = professor.id;
```



## LEFT OUTER JOIN

- Retorna todos os registros da **tabela da esquerda**, mesmo que não exista correspondência na direita

```
SELECT aluno.nome, curso.nome  
FROM aluno  
LEFT JOIN curso  
ON aluno.id_curso = curso.id;
```

- **Exemplo:** Mostra todos os alunos, inclusive os que **ainda não tem curso cadastrado**

## RIGHT OUTER JOIN

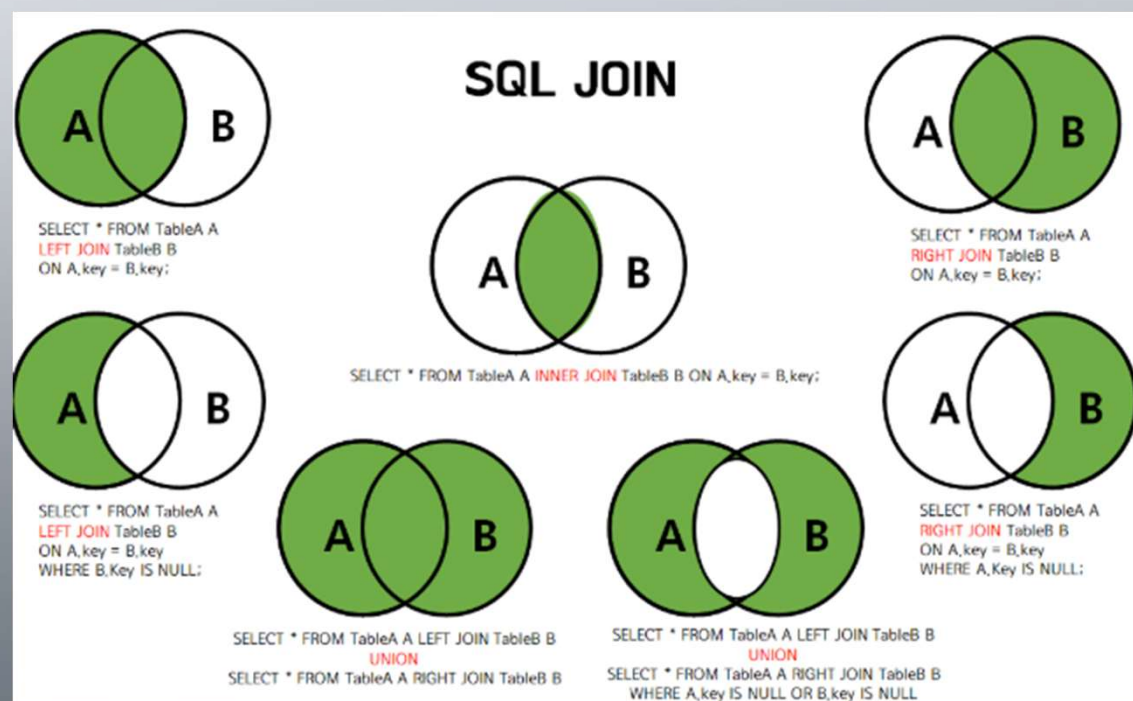
- Retorna todos os registros da **tabela da direita**, mesmo que não exista correspondência na esquerda

```
SELECT aluno.nome, curso.nome  
FROM aluno  
RIGHT JOIN curso  
ON aluno.id_curso = curso.id;
```

- **Exemplo:** Mostra todos os cursos, mesmo sem alunos matriculados

# COMPARATIVO

Tipo de JOIN	Tabela A	Tabela B	Resultado
INNER JOIN	Somente correspondentes	Somente correspondentes	Interseção
LEFT JOIN	Todos	Apenas correspondentes	Todos da esquerda
RIGHT JOIN	Apenas correspondentes	Todos	Todos da direita





Perguntas?! Dúvidas?!?

