

TIPOS DE JUNÇÃO DE TABELAS

- **EQUI-JUNÇÃO - JOIN e INNER JOIN**

Ligação entre tabelas quando os valores das colunas responsáveis pela junção das duas tabelas são iguais. A equi-junção é possível apenas quando definimos de forma adequada a FK de uma tabela e sua referência à PK da tabela precedente.

```
SELECT v.Nome
FROM vendedor v, Departamento d
WHERE v.codDepartamento = d.codDepartamento AND d.nome = 'Marketing'
```

É equivalente a:

```
SELECT v.Nome
FROM vendedor v join Departamento d on v.codDepartamento = d.codDepartamento
WHERE d.nome = 'Marketing'
```

- **JUNÇÃO NATURAL – NATURAL JOIN**

Quando a junção das tabelas R e S é realizada através da combinação de pares de atributos de mesmo nome em R e S.

```
SELECT *
FROM cliente NATURAL INNER JOIN cliente_fisico
```

- **CROSS JOIN - UNJOINED**

Quando uma consulta utiliza dados de várias tabelas, se não especificar-se uma forma de ligar essas tabelas, teremos o produto cartesiano de todas as tabelas especificadas.

- **JUNÇÃO EXTERNA – OUTER JOIN**

É utilizada quando a consulta deve apresentar valores de uma das tabelas mesmo que não exista uma junção correspondente entre as FK e PK . Os atributos da linha “não existente” são preenchidos com nulos.

Variantes:

- LEFT
- RIGHT
- FULL

O LEFT OUTER JOIN retorna todas as linhas do produto cartesiano qualificado (i.e., todas as linhas combinadas que passam pela condição ON), mais uma cópia de cada linha da tabela à esquerda para a qual não há uma linha da tabela à direita que tenha passado pela

condição ON. Esta linha da tabela à esquerda é estendida por toda a largura da tabela combinada inserindo-se nulos para as colunas da tabela à direita. Observe que somente as condições ON do próprio JOIN são consideradas na hora de decidir quais linhas possuem correspondência. As condições WHERE externas são aplicadas depois.

De forma inversa, o RIGHT OUTER JOIN retorna todas as linhas da junção, mais uma linha para cada linha da tabela à direita sem correspondência (estendida com nulos na tabela à esquerda). Isto é apenas uma conveniência da notação, porque poderia ser convertida em um LEFT OUTER JOIN trocando-se a tabela à direita pela tabela à esquerda.

O FULL OUTER JOIN retorna todas as linhas da junção, mais uma linha para cada linha da tabela à esquerda sem correspondência (estendida com nulos na tabela à direita), mais uma linha da tabela à direita sem correspondência (estendida com nulos na tabela à esquerda).

Exemplo: Listar os departamentos e seus funcionários, apresentando também os departamentos que não possuem funcionários.

SELECT d.nome, v.nome

FROM departamento d **left join** vendedor v **on** v.cod_departamento = d.cod_departamento

EXERCÍCIOS:

Para verificar se as consultas estão corretas, utilize o banco de dados chamado codeacademy que foi disponibilizado.

Desenvolva consultas usando junção para:

1. Apresentar os pedidos realizados no ano de 2014. Campos: código do pedido, data de realização, data de entrega e nome fantasia do fornecedor. Usar junção natural.

```
SELECT pedido.codPedido, pedido.dataRealizacao, pedido.dataEntrega,
fornecedor.nomeFantasia
FROM pedido
NATURAL JOIN fornecedor
```

2. Listar os vendedores da cidade de 'Apucarana'. Campos: nome do vendedor, data de nascimento, nome da cidade. Usar equi-junção.

```
SELECT vendedor.nome, vendedor.dataNascimento, cidade.nome
FROM vendedor, cidade
WHERE vendedor.codCidade = cidade.codCidade AND cidade.nome = "Apucarana"
```

3. Listar as informações das vendas do produto 'Cal'. Campos: código da venda, data da venda, nome do vendedor, número do lote, descrição do produto. Usar junção natural.

```
SELECT venda.codVenda, venda.dataVenda, vendedor.nome, produtoLote.numeroLote,
produto.descricao
FROM venda
NATURAL JOIN vendedor
NATURAL JOIN produtoLote
NATURAL JOIN produto
WHERE produto.descricao = "Cal"
```

4. Listar as informações dos produtos da classe 'Acabamentos'. Campos: sigla da classe, nome da classe, código do produto, descrição, estoque mínimo.

```
SELECT classe.sigla, classe.nome, produto.codProduto, produto.descricao,
produto.estoqueMinimo
FROM classe
JOIN produto ON classe.codClasse = produto.codClasse
WHERE classe.nome = "Acabamentos"
```

5. Listar as informações dos pedidos do fornecedor 'Incepa'. Campos: Nome Fantasia do fornecedor, código do pedido, data de realização, data da entrega.

```
SELECT fornecedor.nomeFantasia, pedido.codPedido, pedido.dataRealizacao,
pedido.dataEntrega
FROM fornecedor
JOIN pedido ON fornecedor.codFornecedor = pedido.codFornecedor
WHERE fornecedor.razaoSocial = "Incepa"
```

6. Apresentar todos os produtos, e quando existirem, as vendas relativas ao produto. Campos: código do produto, descrição do produto, código da venda. USAR LEFT JOIN

```
SELECT produto.codProduto, produto.descricao, itemVenda.codVenda
FROM produto
LEFT JOIN itemVenda ON produto.codProduto = itemVenda.codProduto
```

7. Apresentar todos os fornecedores cadastrados, e quando existirem, os seus pedidos. Campos: nome fantasia do fornecedor, código do pedido, data de entrega do pedido. USAR LEFT JOIN

```
SELECT fornecedor.nomeFantasia, pedido.codPedido, pedido.dataEntrega
FROM fornecedor
LEFT JOIN pedido ON fornecedor.codFornecedor = pedido.codFornecedor
```

8. Apresentar todos os departamentos, e seus vendedores. Apresentar também os departamentos que não possuem vendedores. Campos: nome do departamento, localização, nome do funcionário, data de nascimento. USAR RIGHT JOIN

```
SELECT departamento.nome AS nome_departamento, departamento.localizacao,
vendedor.nome, vendedor.dataNascimento
FROM vendedor
RIGHT JOIN departamento ON vendedor.codDepartamento =
departamento.codDepartamento
```

Desafio Opcional:

9. Apresentar todos os clientes e suas respectivas compras, apresentar também os clientes que não fizeram compras. Campos: nome do cliente se o mesmo for pessoa física ou o nome fantasia se o mesmo for pessoa jurídica, código da venda. USAR RIGHT JOIN

Questões extras e opcionais:

10. Apresentar todas as classes, e quando existirem os produtos da classe. Campos: todos os campos da classe, descrição do produto, preço de custo. USAR LEFT JOIN
11. Apresentar todos os estados e, quando existirem suas respectivas cidades. Campos todos de estado e cidade. USAR RIGHT JOIN
12. Apresentar todos os produtos e seus respectivos lotes quando existirem. Campos: código do produto, descrição, número do lote, data de validade do lote. USAR LEFT JOIN
13. Apresentar todas as cidades, e quando existirem, os vendedores nela cadastrados. Campos: código da cidade, nome, sigla do estado, nome do vendedor, endereço, telefone. USAR RIGHT JOIN
14. Apresentar todos os vendedores, e quando existirem as vendas sob responsabilidade do vendedor. Campos: nome, data de contratação, código da venda, data da venda. USAR LEFT JOIN
15. Apresentar todas as cidades e os clientes nela cadastrados, quando existirem. Campos: nome da cidade, nome do cliente. USAR RIGHT JOIN