

Banco de dados II

Aula 4

Outubro de 2024

Segurança em Bancos de dados

- INNER
- OUTER
- CROSS
- SELF



01

Join (Relacionando tabelas)

Banco de dados

JOIN (relacionando tabelas)

Um join é uma operação que permite você consultar duas ou mais tabelas para produzir um result set que incorpora linhas e colunas de cada tabela. Seu join ocorre em colunas das tabelas que são comuns em ambas às tabelas.

TIPOS DE JOINS

- Inner Join
- Outer Join
- Cross Join
- Self Join

Utilizando o JOIN

Vamos imaginar que para o relacionamento existente entre as tabelas Pedidos e Clientes possa acontecer os seguintes cenários:

- 1) Existem clientes que ainda não fizeram nenhum pedido;
- 2) Existem pedidos que foram efetuados por clientes não cadastrados no sistema;
- 3) Existem pedidos que foram realizados por clientes cadastrados.

Utilizando o JOIN

Para os possíveis cenários o comando join seria basicamente o mesmo:

```
SELECT campos  
FROM Clientes c TIPO_DE_JOIN JOIN Pedidos p  
ON c.codigodocliente = p.codigodocliente;
```

Mudando o tipo do Join de acordo com o cenário. Vamos deixar o claro o seguinte, este comando é bastante utilizado quando não existe a obrigatoriedade do relacionamento. Isto é, pode haver clientes sem pedidos e pedidos sem cliente.



02

INNER JOIN



Utilizando o JOIN - INNER

O Inner Join é utilizado nas situações em que você quer selecionar os registros das duas tabelas, desde que as mesmas possuam informações cruzadas (relacionadas). No nosso exemplo, é o JOIN que resolve a situação 3: a consulta vai retornar os clientes cadastrados que efetuaram pedidos.

```
SELECT campos
```

```
FROM Clientes c INNER JOIN Pedidos p
```

```
ON c.codigodocliente = p.codigodocliente;
```




03

OUTHER JOIN

Utilizando o JOIN - OUTER

O Outer join possui o funcionamento um pouco diferente do anterior. Ao usar o Outer join, além de podermos retornar os registros das duas tabelas seguindo alguma relação, ainda podemos retornar registros que não entram nesta relação.

- Existem dois tipos de Outer Join
- Left Outer Join
- Right Outer Join

Utilizando o JOIN - OUTER

Left Outer Join ou Left Join

Aplica o conceito de Outer Join na tabela que se encontrar à esquerda da relação (ou seja, o resultado vem da tabela à esquerda). No nosso exemplo, é o JOIN que resolve a situação 1, a consulta retorna apenas os clientes que ainda não efetuaram nenhum pedido.

```
SELECT *  
FROM Clientes c LEFT OUTER JOIN Pedidos p  
ON c.codigodocliente = p.codigodocliente;
```

Utilizando o JOIN - OUTER

Right Outer Join ou Right Join

Aplica o conceito de Outer Join na tabela que se encontrar à esquerda da relação (ou seja, o resultado vem da tabela à esquerda). No nosso exemplo, é o JOIN que resolve a situação 2, a consulta retorna apenas os pedidos que foram realizados por clientes não cadastrados.

```
SELECT *  
FROM Clientes c RIGTH OUTER JOIN Pedidos p  
ON c.codigodocliente = p.codigodocliente;
```



04

CROSS JOIN

Utilizando o JOIN - CROSS

Cross

Cross join mostra combinação de todas as linhas das tabelas unidas. Nenhuma coluna comum será necessária para fazer um cross join. Quando você usa cross joins, SQL Server produz um produto cartesiano no qual o número de linhas no resultado seja igual ao número de segunda tabela. Por exemplo, se são 8 linhas na primeira tabela e 9 linhas na outra tabela SQL Server retorna um total de 72 linhas.

Utilizando o JOIN - CROSS

Cross

No nosso exemplo, deseja-se gerar um conjunto de dados que contém todas as combinações possíveis de valores entre as duas tabelas.

```
SELECT *  
FROM clientes  
CROSS JOIN produtos;
```



05

SELF JOIN

Utilizando o JOIN - SELF

SELF JOIN é uma técnica empregada para conseguir o produto cartesiano de uma tabela consigo mesma. Sua utilização não é muito frequente, mas colocaremos algum exemplo de sua utilização. Suponhamos a seguinte tabela (O campo autor é numérico, embora para ilustrar o exemplo utilize o nome):



.....

Obrigado!

Alguma dúvida?
jheymesso.Cavalcanti@unicap.br