

Cláusula JOIN...

A cláusula JOIN é usada para combinar dados provenientes de duas ou mais tabelas do banco de dados, baseado em um relacionamento entre colunas destas tabelas. há duas categorias principais de joins:

INNER JOIN: Retorna linhas (registros) quando houver pelo menos uma correspondência em ambas as tabelas.

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

Inner Join

Como dito, um INNER JOIN permite obter registros com dados provenientes de duas ou mais tabelas relacionadas do banco de dados no MySQL. A sintaxe básica de um INNER JOIN em uma consulta é:

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

Onde tabela1.coluna é o nome da primeira tabela concatenado com um ponto e com o nome da coluna chave primária ou estrangeira da tabela, e tabela2.coluna é o nome da segunda tabela concatenado com um ponto e com a chave estrangeira ou primária dessa tabela que se relaciona com a chave da primeira tabela.

Inner Join

Vejamos um exemplo. Vamos consultar as tabelas de livros e autores (tbl_livro e tbl_autores) e retornar os dados relativos aos livros e aos autores ao mesmo tempo:

```
SELECT * FROM tbl_livro  
INNER JOIN tbl_autores  
ON tbl_livro.ID_Autor = tbl_autores.ID_Autor;
```

Inner Join

Vejamos outro exemplo. Faremos a consulta nas tabelas de livros e autores, porém retornando apenas os nomes dos livros, seus ISBNs e os nomes dos autores desses livros. Note que as duas primeiras informações estão na tabela de livros, e o nome do autor está somente na tabela de autores. O que as liga é o relacionamento entre as colunas ID_Autor existentes nas tabelas - é chave primária na tabela de autores e estrangeira na de livros:

```
SELECT tbl_Livro.Nome_Livro, tbl_Livro.ISBN, tbl_autores.Nome_Autor  
FROM tbl_Livro  
INNER JOIN tbl_autores  
ON tbl_Livro.ID_Autor = tbl_autores.ID_Autor;
```

Inner Join

Mais um exemplo, agora com filtros. Vamos retornar os nomes dos livros e nomes das editoras, mas somente das editoras cujo nome se inicia com a letra M. Note o uso de aliases nestas declarações, a fim de simplificar o código:

```
SELECT L.Nome_Livro AS Livros, E.Nome_editora AS Editoras
FROM tbl_Livro AS L
INNER JOIN tbl_editoras AS E
ON L.ID_editora = E.ID_editora
WHERE E.Nome_Editora LIKE 'M%'
```

Inner Join

Mais um exemplo para terminar. Agora vamos fazer um INNER JOIN com as três tabelas do banco de dados simultaneamente. Queremos os nomes e preços dos livros, nomes de seus autores e editoras, mas somente das editoras cujo nome se inicia com a letra O, tudo isso ordenado em ordem decrescente de preço dos livros:

```
SELECT L.Nome_Livro AS Livro,  
       A.Nome_autor AS Autor,  
       E.Nome_Editora AS Editora,  
       L.Preco_Livro AS 'Preço do Livro'  
FROM tbl_Livro AS L  
INNER JOIN tbl_autores AS A  
  ON L.ID_autor = A.ID_autor  
INNER JOIN tbl_editoras AS E  
  ON L.ID_editora = E.ID_editora  
WHERE E.Nome_Editora LIKE 'O%'  
ORDER BY L.Preco_Livro DESC;
```

Inner Join

Mais um exemplo para terminar. Agora vamos fazer um INNER JOIN com as três tabelas do banco de dados simultaneamente. Queremos os nomes e preços dos livros, nomes de seus autores e editoras, mas somente das editoras cujo nome se inicia com a letra O, tudo isso ordenado em ordem decrescente de preço dos livros:

```
SELECT L.Nome_Livro AS Livro,  
       A.Nome_autor AS Autor,  
       E.Nome_Editora AS Editora,  
       L.Preco_Livro AS 'Preço do Livro'  
FROM tbl_Livro AS L  
INNER JOIN tbl_autores AS A  
  ON L.ID_autor = A.ID_autor  
INNER JOIN tbl_editoras AS E  
  ON L.ID_editora = E.ID_editora  
WHERE E.Nome_Editora LIKE 'O%'  
ORDER BY L.Preco_Livro DESC;
```