

PESQUISA EM MÚLTIPLAS TABELAS

Nos capítulos anteriores as pesquisas foram realizadas em apenas uma tabela. No entanto, observamos no modelo criado, assim como acontece sempre, que uma tabela está relacionada à outra, sendo necessário extrair informações de duas ou mais tabelas relacionadas ao mesmo tempo. Esta tarefa recebe o nome de união ou junção de tabelas.

Isso se dá por meio das chaves primárias e chaves estrangeiras, que são no modelo físico as colunas que as tabelas têm em comum.

Para os exemplos utilizados nesta aula é fundamental fazer uso do Modelo Lógico para melhor visualizar os relacionamentos.

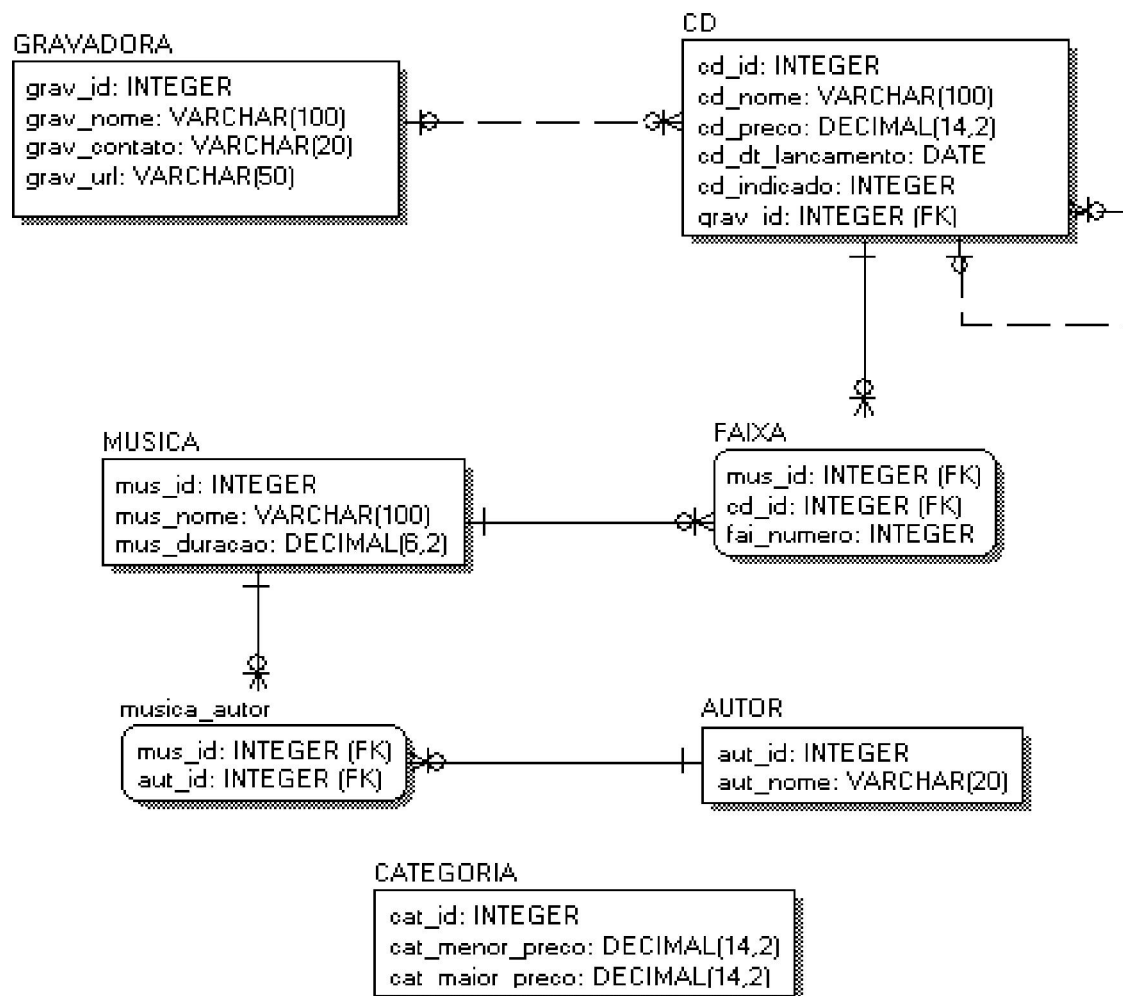


Figura 16 - Modelo Lógico de Dados

6.1. ALIAS

Ao realizar pesquisa em diversas tabelas são referenciadas colunas pertencentes às tabelas da relação. Dessa forma, é necessário distinguir a qual tabela pertence à dada coluna. Podemos fazer isso de duas formas:

- a) Colocando o nome da tabela na frente do nome da coluna.

Sintaxe:

```
SELECT tabela.coluna  
FROM tabela;
```

- b) Utilizando um **alias ou apelido** para a tabela e colocando-o na frente da coluna.

Sintaxe:

```
SELECT t.coluna  
FROM tabela t;
```

É ainda possível usar alias em nome de colunas, fazendo uso da cláusula AS;

Sintaxe:

```
SELECT coluna AS nome_desejado  
FROM tabela;
```

6.2. UNIÃO DE TABELAS

Para realizar a união de tabelas basta acrescentar após a cláusula FROM do comando SELECT as tabelas que queremos unir utilizando após a cláusula WHERE a condição de união entre elas (Chave Primária e Chave Estrangeira).

Sintaxe:

```
SELECT [tabela1.]coluna [, [tabela2.]coluna, ...]  
FROM tabela1, tabela2 [, ...]  
WHERE tabela1.PK = tabela2.FK
```

Uma prática recomendável é utilizar o mesmo nome nos campos de chave primária e chave estrangeira, pois dessa forma facilita a identificação dos campos ao realizar a união das tabelas.

É possível colocar diversas tabelas na cláusula FROM. Não devemos esquecer, porém, que é necessário especificar as chaves primárias e chaves estrangeiras de cada relacionamento.

Em caso de dúvidas, consulte o Modelo de Dados para estabelecer quais são as colunas comuns entre as tabelas. Daí a importância de se ter um bom **projeto de Banco de Dados claro e preciso**.

A cláusula WHERE deve conter todo canal de relacionamento entre as tabelas listadas na cláusula FROM sob pena de ocorrer o Produto Cartesiano.

6.3. PRODUTO CARTESIANO

Ocorrerá um produto cartesiano sempre que:

- A condição de união entre as tabelas for omitida (não houver cláusula WHERE);
 - Condição de união entre as tabelas for inválida (cláusula WHERE incorreta);
 - Todas as linhas da **primeira tabela** estiverem unidas a todas as linhas da **segunda tabela**.
- Nesta situação, as linhas da primeira tabela serão combinadas com as linhas da segunda, demonstrando um resultado na maior parte das vezes indesejado.
- Esse procedimento é muito utilizado para efetuar testes de **performance** no banco devido a grande quantidade de dados.

Exemplo:

```
SELECT cd.cd_id, cd.cd_nome, gravadora.grav_nome
FROM gravadora, cd
```

Ao invés de se mostrar todos os dados, deseja-se visualizar os CD's e suas respectivas gravadoras. Emprega-se então, a União entre tabelas.

Exemplo:

```
SELECT cd.cd_id, cd.cd_nome, gravadora.grav_nome
FROM gravadora, cd
WHERE cd.grav_id = gravadora.grav_id
```

Usando alias:

```
SELECT c.cd_id, c.cd_nome, g.grav_nome
FROM gravadora g, cd c
WHERE c.grav_id = g.grav_id
```

6.4. UNIÃO REGULAR (*INNER JOIN* OU *EQUI-JOIN*)

Denomina-se união regular as uniões que têm a cláusula WHERE unindo chave primária e chave estrangeira como visto anteriormente.

O padrão determina uma sintaxe alternativa para esse comando. Quando a **chave primária e a chave estrangeira têm o mesmo nome em ambas as tabelas** é possível simplificar o comando usando o comando **NATURAL JOIN**.

Exemplo:

```
SELECT cd.cd_id, cd.cd_nome, gravadora.grav_nome
FROM cd
NATURAL JOIN gravadora;
```

Outras duas maneiras definidas no padrão SQL é determinar qual(is) coluna(s) utilizar na união usando a cláusula **USING** e caso o nome das colunas não sejam iguais, determinar quais são as colunas com a cláusula **ON**. Ambas produzindo o mesmo efeito.

Exemplo:

```
SELECT cd.cd_id, cd.cd_nome, gravadora.grav_nome
FROM cd JOIN gravadora USING (grav_id);
```

```
SELECT cd.cd_id, cd.cd_nome, gravadora.grav_nome
FROM cd JOIN gravadora ON gravadora.grav_id = cd.grav_id;
```

6.5. UNIÃO DE MAIS DE DUAS TABELAS

Freqüentemente é necessário unir mais de duas tabelas para obter uma informação consistente.

Suponha que queira saber o nome da música, a faixa e o nome do CD em que está a música. No modelo de dados observamos a necessidade de se consultar três tabelas no Banco de Dados.

Exemplo:

```
SELECT c.cd_nome, f.fai_numero, m.mus_nome
FROM cd c, faixa f, musica m
WHERE c.cd_id = f.cd_id AND
      m.mus_id = f.mus_id
```

O código abaixo produziria o mesmo efeito:

```
SELECT c.cd_nome, f.fai_numero, m.mus_nome
FROM faixa f
INNER JOIN cd c ON c.cd_id = f.cd_id
INNER JOIN musica m ON m.mus_id = f.mus_id
```

Observe que é possível realizar a união de tantas tabelas quanto forem necessárias.

```
SELECT a.aut_nome, g.grav_nome
FROM autor a, cd c, gravadora g, faixa f, musica_autor ma, musica
m
WHERE a.aut_id = ma.aut_id AND
      m.mus_id = ma.mus_id AND
      m.mus_id = f.mus_id AND
      c.cd_id = f.mus_id AND
      g.grav_id = c.grav_id
```

6.6. UNIÕES EXTERNAS (OUTER-JOIN)

Observa-se na união entre tabelas que quando uma linha não satisfaz a condição de união entre as tabelas, ela não será mostrada no resultado da busca. Isto acontece porque o banco de dados, não podendo estabelecer a relação entre as colunas que estão sendo unidas na busca, coloca NULL onde o dado não existe.

Exemplo:

```
SELECT cd.cd_id, cd.cd_nome, gravadora.grav_nome
FROM gravadora, cd
WHERE cd.grav_id = gravadora.grav_id
```

No resultado observamos que alguns dados de gravadora são omitidos.

Define-se União Externa como aquela que inclui linhas no resultado da busca mesmo que não haja relação entre as duas tabelas.

Para que os dados omitidos sejam mostrados utilizaremos o comando SELECT utilizando União Externa.

6.6.1. União Externa à Esquerda (*Left Outer-Join*)

Como o nome diz, a união pela esquerda incluirá todas as linhas da primeira tabela na expressão, ou seja a tabela da esquerda será a tabela base para a união.

Observe esse comando e depois inverta a posição das tabelas **GRAVADORA** e **CD**.

Exemplo:

```
SELECT cd.cd_id, cd.cd_nome, gravadora.grav_nome
FROM gravadora
LEFT OUTER JOIN cd ON (gravadora.grav_id = cd.grav_id)
```

Neste caso, todas as gravadoras são mostradas independente de haver ou não dados relacionados na tabela CD.

6.6.2. União Externa à Direita (*Right Outer-join*)

Ao contrário da anterior, a união é realizada pela direita, pois incluirá todas as linhas da tabela da direita na expressão da união, ou seja a tabela da direita será a tabela base para a união.

Exemplo:

```
SELECT cd.cd_id, cd.cd_nome, gravadora.grav_nome
FROM gravadora
RIGHT OUTER JOIN cd ON (gravadora.grav_id = cd.grav_id)
```

NOTA:

Se a tabela da referência (direita ou esquerda) for a que contém a chave estrangeira, ocorrerá uma união regular entre as tabelas.

6.6.3. União Externa Total (*Full Outer-join*)

Realiza uma união independente da coluna opcional estar à direita ou à esquerda. Neste caso, são geradas linhas que não existem em uma ou em ambas as tabelas.

Exemplo:

```
SELECT cd.cd_id, cd.cd_nome, gravadora.grav_nome  
FROM gravadora  
FULL OUTER JOIN cd ON (gravadora.grav_id = cd.grav_id)
```

Neste caso a consulta age como uma união externa a direita ou a esquerda por não haver campos nulos na coluna de chave estrangeira de CD.

6.7. EXERCÍCIOS

- a. Faça uma busca que mostre todos os CD's e o nome de suas respectivas gravadoras.
- b. Faça uma pesquisa que mostre o produto cartesiano das tabelas MUSICA e CD.
- c. Realize uma consulta se utilizando da sintaxe JOIN que mostre todas MUSICAS e o nome de seus respectivos CD's.
- d. Crie uma pesquisa que retorne como resultado todas as gravadoras e seus respectivos CD's mesmo que estes não possuam CD's associados.
- e. Crie uma pesquisa que mostre o AUTOR e suas MUSICAS ordenadas por AUTOR.
- f. Acrescente a pesquisa anterior o nome do CD que foi gravado a música.
- g. Crie uma pesquisa que mostre em quais gravadoras os autores tem realizado suas gravações.