



**Instituto Federal**  
Campus Goiânia

**Bacharelado em Sistemas de Informação**

# **Banco de Dados II**



**Prof. Dory Gonzaga Rodrigues**





## Agenda

- Pesquisa em Múltiplas Tabelas
  - Produto Cartesiano
  - União de Tabelas
  - Alias
  - Junção
    - Interna
    - Externa
    - Auto-Relacionamento

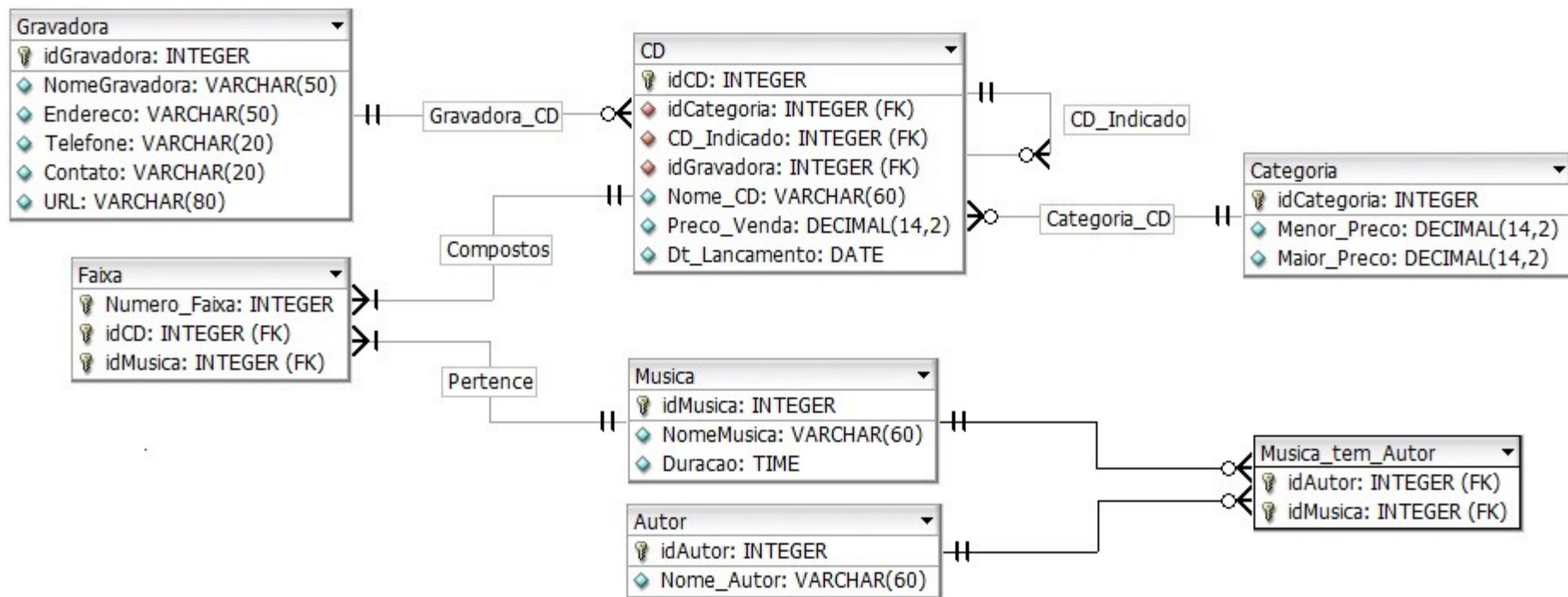




## DML BANCO DE DADOS

### BANCO DE DADOS REFERÊNCIA

- Utilizaremos nesta aula o banco de dados: Catálogo de CDs



Arquivo enviado por e-mail: [DDL\\_Catalogo\\_CDs.sql](#)  
[DML\\_Catalogo\\_CDs.sql](#)





## DML SELECT

### SQL – DML

#### - Manipulação dos dados em Tabela: **SELECT**

O comando **SELECT** é sem sombra de dúvidas o comando mais utilizado e por isso o mais importante da linguagem SQL. Este comando permite selecionar os dados armazenados no Banco de Dados.

##### A sintaxe:

```
SELECT      [ DISTINCT | ALL ]      { * , coluna1, coluna1, ..., colunaN }  
FROM       nome da tabela  
[ ORDER BY <campo>      ]  
[ WHERE     <condição>   ]  
[ GROUP BY <campo>      ]  
[ HAVING    <condição>   ]
```

[ ] Opcional

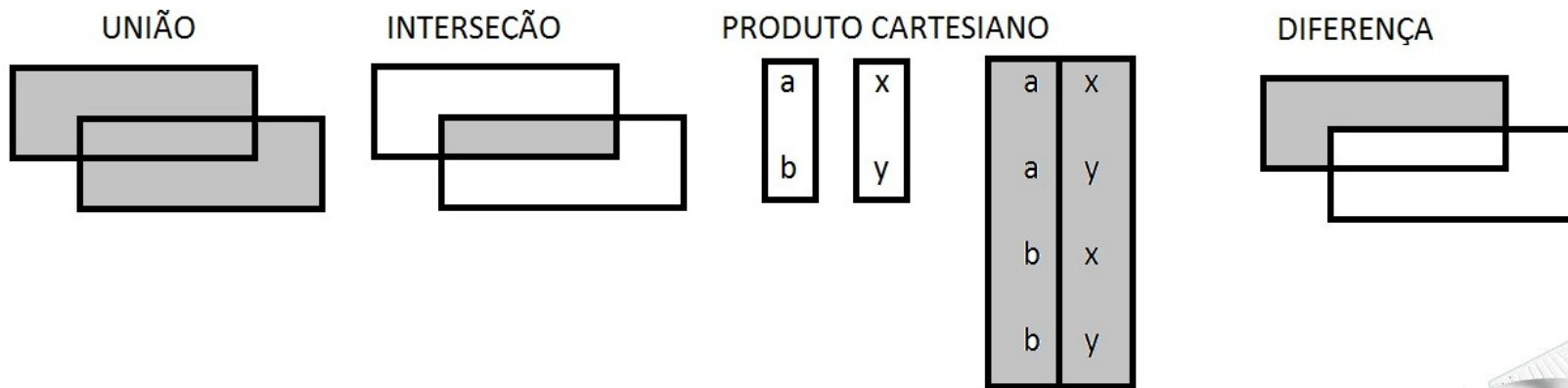




## DML SELECT PESQUISA – Múltiplas Tabelas

### SQL – DML – Pesquisa em Múltiplas Tabelas

- Na Montagem do Modelo de Dados utilizamos várias tabelas relacionadas. Desta forma, vamos aprender como vincular a informação dessas tabelas de forma correta.
- Quando trabalhamos com múltiplas tabelas no SELECT, utilizamos os conceitos de PRODUTO CARTESIANO, UNIÃO, INTERSEÇÃO.





DML SELECT PESQUISA – Múltiplas Tabelas

## SQL – DML – Pesquisa em Múltiplas Tabelas

### PRODUTO CARTESIANO

Para realizar o **PRODUTO CARTESIANO** entre duas tabelas, basta acrescentarmos os nomes das tabelas na cláusula FROM. Abaixo temos o comando SELECT realizando o produto cartesiano entre as tabelas CD e GRAVADORA

Resp: **SELECT** idCD, Nome\_CD, NomeGravadora  
**FROM** cd, gravadora;

- Ocorrerá um produto cartesiano sempre que:
  - Não utilizar a cláusula WHERE;
  - Quando a condição da cláusula WHERE for inválida;
  - Todas as linhas da 1ª Tabela estiverem unidas a todas as linhas da 2ª Tabela.





DML SELECT PESQUISA – Múltiplas Tabelas

## SQL – DML – Pesquisa em Múltiplas Tabelas

### UNIÃO REGULAR (INNER JOIN OU EQUI-JOIN)

Para realizar a **UNIÃO REGULAR (INNER JOIN OU EQUI-JOIN)** entre duas tabelas, basta acrescentarmos na cláusula **WHERE** a união da chave primária à chave estrangeira das tabelas que participam na cláusula **FROM** do comando **SELECT**.

Resp: **SELECT** idCD, Nome\_CD, NomeGravadora  
FROM cd, gravadora  
WHERE cd.idGravadora = gravadora.idGravadora;





DML SELECT PESQUISA – Múltiplas Tabelas

## SQL – DML – Pesquisa em Múltiplas Tabelas

### UNIÃO REGULAR (INNER JOIN OU EQUI-JOIN)

A **UNIÃO REGULAR (INNER JOIN OU EQUI-JOIN)** entre duas tabelas, também pode ser feita através das seguintes sintaxes alternativas:

- Somente quando a chave primária e estrangeira têm o mesmo nome.

Resp: **SELECT** idCD, Nome\_CD, NomeGravadora  
**FROM** cd **NATURAL JOIN** gravadora;

Resp: **SELECT** idCD, Nome\_CD, NomeGravadora  
**FROM** cd **JOIN** gravadora **USING**(idGravadora);







DML SELECT PESQUISA – Múltiplas Tabelas

## SQL – DML – Pesquisa em Múltiplas Tabelas

### UNIÃO REGULAR (INNER JOIN OU EQUI-JOIN)

A **UNIÃO REGULAR (INNER JOIN OU EQUI-JOIN)** entre duas tabelas, também pode ser feita através das seguintes sintaxes alternativas:

- Quando a chave primária e estrangeira têm nomes diferentes.

Resp: **SELECT** idCD, Nome\_CD, NomeGravadora  
**FROM** cd **JOIN** gravadora  
**ON** cd.idGravadora = cd.idGrav;

Atenção: No banco Catálogo de CD's, as chaves possuem nomes iguais!





DML SELECT PESQUISA – Múltiplas Tabelas

## SQL – DML – Pesquisa em Múltiplas Tabelas

### APELIDOS EM TABELAS

- Da mesma forma que utilizamos apelidos para as colunas que participam da projeção no comando **SELECT**, podemos criar apelidos para as tabelas;
- O objetivo é evitar que o comando fique extremamente extenso;

Resp: **SELECT** a.idCD, Nome\_CD, b.NomeGravadora  
**FROM** cd a, gravadora b  
**WHERE** a.idGravadora = b.idGravadora;





DML SELECT PESQUISA – Múltiplas Tabelas

## SQL – DML – Pesquisa em Múltiplas Tabelas

### UNIÃO REGULAR (INNER JOIN OU EQUI-JOIN)

#### EXEMPLO

- Selecione o nome dos CD's, suas faixas e o nome das músicas que pertencem aos CD's com código 1 ou 2:





## DML SELECT PESQUISA – Múltiplas Tabelas

### SQL – DML – Pesquisa em Múltiplas Tabelas

#### UNIÃO REGULAR (INNER JOIN OU EQUI-JOIN)

##### EXEMPLO

- Selecione o nome dos CD's, suas faixas e o nome das músicas que pertencem aos CD's com código 1 ou 2:

Resp: **SELECT** c.nome\_cd, f.numero\_faixa, m.nomemusica  
      **FROM** CD c, FAIXA f, MUSICA m  
      **WHERE** c.idCD IN (1,2)  
            **AND** c.idCD = f.idCD  
            **AND** f.idMusica = m.idMusica;

Resp: **SELECT** c.nome\_cd, f.numero\_faixa, m.nomemusica  
      **FROM** CD c **NATURAL JOIN** FAIXA f **NATURAL JOIN** MUSICA m  
      **WHERE** c.idCD IN (1,2);





DML SELECT PESQUISA – Múltiplas Tabelas

## SQL – DML – Pesquisa em Múltiplas Tabelas

### UNIÃO SEM COLUNAS CHAVE EM COMUM (NON-JOIN)

- Em situações em que não existe um relacionamento explícito entre colunas de tabelas, mas podemos criar relacionamentos de uma coluna de uma tabela com o intervalo de outras colunas de outra tabela.

Resp: **SELECT** c.Nome\_CD, c.Preco\_Venda, ca.idCategoria  
**FROM** CD c, cd\_categoria ca  
**WHERE** c.Preco\_Venda **BETWEEN** ca.Menor\_Preco **AND** ca.Maior\_Preco;





DML SELECT PESQUISA – Múltiplas Tabelas

## SQL – DML – Pesquisa em Múltiplas Tabelas

### UNIÃO EXTERNAS (OUTER-JOIN)

- Quando desejamos que as tuplas que não satisfazem a condição de união entre as tabelas seja apresentada devemos utilizar a UNIÃO EXTERNA.
- Existem 3 formas de realizar a União Externa:
  - à Esquerda (Left Outer Join)
  - à Direita (Right Outer Join)
  - Total (Full Outer Join)





DML SELECT PESQUISA – Múltiplas Tabelas

## SQL – DML – Pesquisa em Múltiplas Tabelas

### UNIÃO EXTERNAS (LEFT OUTER JOIN)

- Incluirá as linhas da primeira tabela que aparece na expressão de união.

```
SELECT c.idCD, c.Nome_CD, c.idGravadora, g.NomeGravadora  
FROM CD c LEFT OUTER JOIN gravadora g;
```

```
SELECT c.idCD, c.Nome_CD, idGravadora, g.NomeGravadora  
FROM CD c LEFT OUTER JOIN gravadora g USING(idGravadora) ;
```

- Se o nome dos campos fossem diferentes nas tabelas utilizamos a sintaxe abaixo:

```
SELECT c.idCD, c.Nome_CD, c.idGravadora, g.NomeGravadora  
FROM CD c LEFT OUTER JOIN gravadora g  
ON (c.idGravadora = g.idGrav) ;
```





DML SELECT PESQUISA – Múltiplas Tabelas

## SQL – DML – Pesquisa em Múltiplas Tabelas

### UNIÃO EXTERNAS (RIGHT OUTER JOIN)

- Incluirá as linhas da segunda tabela que aparece na expressão de união.

```
SELECT c.idCD, c.Nome_CD, g.idGravadora, g.NomeGravadora  
FROM CD c RIGHT OUTER JOIN gravadora g;
```

```
SELECT c.idCD, c.Nome_CD, idGravadora, g.NomeGravadora  
FROM CD c RIGHT OUTER JOIN gravadora g USING(idGravadora) ;
```

- Se o nome dos campos fossem diferentes nas tabelas utilizamos a sintaxe abaixo:

```
SELECT c.idCD, c.Nome_CD, g.idGravadora, g.NomeGravadora  
FROM CD c RIGHT OUTER JOIN gravadora g  
ON (c.idGravadora = g.idGrav) ;
```







DML SELECT PESQUISA – Múltiplas Tabelas

## SQL – DML – Pesquisa em Múltiplas Tabelas

### UNIÃO EXTERNAS (FULL OUTER JOIN)

- Incluirá as linhas de ambas as tabelas que aparecem na expressão de união.

```
SELECT c.idCD, c.Nome_CD, c.idGravadora, g.NomeGravadora  
FROM CD c FULL OUTER JOIN gravadora g;
```

```
SELECT c.idCD, c.Nome_CD, idGravadora, g.NomeGravadora  
FROM CD c FULL OUTER JOIN gravadora g USING(idGravadora) ;
```

- Se o nome dos campos fossem diferentes nas tabelas utilizamos a sintaxe abaixo:

```
SELECT c.idCD, c.Nome_CD, g.idGravadora, g.NomeGravadora  
FROM CD c FULL OUTER JOIN gravadora g  
ON (c.idGravadora = g.idGrav) ;
```





DML SELECT PESQUISA – Múltiplas Tabelas

## SQL – DML – Pesquisa em Múltiplas Tabelas

### UNIÃO EXTERNAS (LEFT OUTER JOIN)

- No Oracle:

(+) deve ficar do lado da coluna  
que pode ter o conteúdo NULL

```
SELECT c.idCD, c.Nome_CD, g.idGravadora, g.NomeGravadora  
FROM CD c FULL OUTER JOIN gravadora g  
WHERE c.idGravadora = (+) g.idGravadora;
```

### UNIÃO EXTERNAS (RIGHT OUTER JOIN)

- No Oracle:

```
SELECT c.idCD, c.Nome_CD, g.idGravadora, g.NomeGravadora  
FROM CD c FULL OUTER JOIN gravadora g  
WHERE c.idGravadora (+) = g.idGravadora;
```





DML SELECT PESQUISA – Múltiplas Tabelas

## SQL – DML – Pesquisa em Múltiplas Tabelas

### UNIÃO COM UMA TABELA (SELF JOIN)

- A União com uma única tabela ocorre quando temos auto-relacionamentos;

Exemplo: Mostrar o código e o nome dos CD's e dos seus CD's indicados;

```
SELECT a.idCD, a.Nome_CD, a.cd_indicado, b.Nome_CD  
FROM CD a, CD b  
WHERE a.cd_indicado = b.idCD;
```

