



Instituto Federal
Campus Goiânia

Bacharelado em Sistemas de Informação

Banco de Dados II



Prof. Dory Gonzaga Rodrigues





Agenda

- Subqueries
 - De uma Linha
 - Múltiplas Linhas
 - Subqueries na cláusula FROM

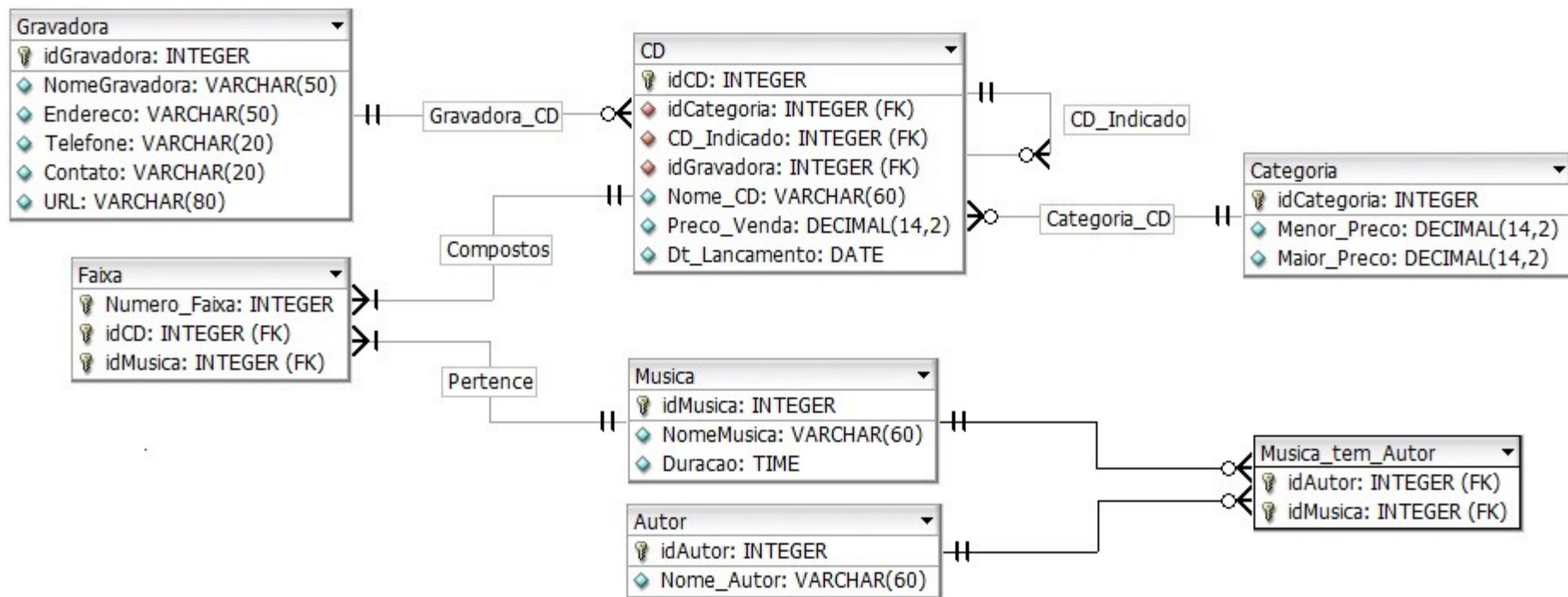




DML BANCO DE DADOS

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 SUBQUERIES

SQL – DML – Subqueries

SubQueries

- A Subqueries: é o uso de consultas aninhadas no comando SELECT;
- Como esta funcionalidade faz parte do padrão SQL, praticamente todos os bancos de dados devem permitir sua utilização;
- Existem 3 tipos de subquery:
 - o retorno do SELECT interno será de uma única linha;
 - o retorno do SELECT interno conterá mais de uma linha;
 - o retorno do SELECT interno conterá mais de uma linha e coluna;





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQueries

Atenção:

- As Subquery devem vir entre parênteses;
- As Subquery devem estar à direita do operador;
- Não coloque a cláusula ORDER BY em uma Subquery;
- Utilize operadores de linha apenas em buscas que retornem uma única linha;
- Utilize operadores de grupo apenas em buscas que potencialmente retornem mais de uma linha;





SQL – DML – Subqueries

SubQuery de uma linha

- Quando o retorno do SELECT interno será de uma única linha

EXEMPLO

Retorne o nome e o preço dos CD's que estão acima da média de preço de venda dos CD's.





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery de uma linha

- Quando o retorno do SELECT interno será de uma única linha

EXEMPLO

Retorne o nome e o preço dos CD's que estão acima da média de preço de venda dos CD's.

Resp: **SELECT** nome_CD, preco_venda
 FROM cd
 WHERE preco_Venda > (
 SELECT **AVG**(preco_venda)
 FROM cd
);





SQL – DML – Subqueries

SubQuery em cláusula HAVING

- Utilizamos a subquery da mesma forma que na cláusula WHERE;
- EXEMPLO
Retorne o código da Gravadora e o maior preço de venda dos CD's que estão acima da média de preço de venda da Gravadora.





SQL – DML – Subqueries

SubQuery em cláusula HAVING

- Utilizamos a subquery da mesma forma que na cláusula WHERE;
- EXEMPLO
Retorne o código da Gravadora e o maior preço de venda dos CD's que estão acima da média de preço de venda da Gravadora.

Resp: **SELECT** idgravadora, **MAX**(preco_venda)
FROM cd a
GROUP BY idgravadora
HAVING **MAX**(preco_venda) > (
SELECT **AVG**(preco_venda)
FROM cd
WHERE idgravadora = a.idgravadora
);






DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery em cláusula HAVING

- Observe que neste caso, estamos utilizando dados do primeiro SELECT (a.idgravadora) para realizar a média no segundo SELECT;

Resp: **SELECT** idgravadora, **MAX**(preco_venda)
FROM cd a
GROUP BY idgravadora
HAVING **MAX**(preco_venda) > (
 SELECT **AVG**(preco_venda)
 FROM cd
 WHERE idgravadora = a.idgravadora
);





SQL – DML – Subqueries

SubQuery com uso do EXISTS

- Na subqueries, a cláusula EXISTS verifica o número de linhas retornadas. Caso seja retornada uma ou mais linhas, então o resultado da primeira SELECT será apresentado;
- EXEMPLO
Retorne o código e o nome das gravadoras que possuem CD's cadastrados.





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery com uso do EXISTS

- Na subqueries, a cláusula EXISTS verifica o número de linhas retornadas. Caso seja retornada uma ou mais linhas, então o resultado da primeira SELECT será apresentado;
- EXEMPLO
Retorne o código e o nome das gravadoras que possuem CD's cadastrados.

Resp: **SELECT** idGravadora, NomeGravadora

FROM gravadora

WHERE EXISTS (

SELECT *

FROM cd

WHERE cd.idGravadora = gravadora.idGravadora

);





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery com uso do EXISTS

- Os exemplos abaixo são equivalentes:

```
SELECT DISTINCT gravadora.idGravadora, gravadora.NomeGravadora  
FROM gravadora, cd  
WHERE gravadora.idGravadora = cd.idGravadora;
```

```
SELECT idGravadora, NomeGravadora  
FROM gravadora  
WHERE EXISTS (  
    SELECT *  
    FROM cd  
    WHERE cd.idGravadora = gravadora.idGravadora  
);
```





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery de múltiplas linhas

- O SELECT INTERNO retorna mais de uma linha (tupla) e sempre uma coluna;
- O SELECT EXTERNO recebe valores que funcionam como uma lista, sendo assim:
 - utilizamos os operadores de grupo: IN, ALL, ANY
 - neste caso, os operadores simples não irão funcionar: = , > , < ou !=





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery de múltiplas linhas

- EXEMPLO

Retorne o código, o nome e o preço de venda dos CD's que têm o preço igual ao menor preço de cada gravadora:

Resp:

1 – Encontramos o menor preço de venda de cada gravadora (lista de valores);

```
SELECT MIN(Preco_Venda) FROM cd GROUP BY idCategoria
```

2 – Depois encontramos quais são os CDs com preços iguais a um dos valores contidos na lista.

```
SELECT idCD, Nome_CD, Preco_Venda  
FROM cd  
WHERE Preco_Venda IN (9,10.5,13);
```





SQL – DML – Subqueries

SubQuery de múltiplas linhas

- EXEMPLO

Retorne o código, o nome e o preço de venda dos CD's que têm o preço igual ao menor preço de cada gravadora:

Resp:

```
SELECT idCD, Nome_CD, Preço_Venda
FROM cd
WHERE Preço_Venda IN (
    SELECT MIN(Preço_Venda)
    FROM cd
    GROUP BY idCategoria
);
```





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery de múltiplas linhas

- EXEMPLO

Retorne o nome do CD, a faixa e o nome das músicas que tenham a palavra “EU” no nome da música:





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery de múltiplas linhas

- EXEMPLO

Retorne o nome do CD, a faixa e o nome das músicas que tenham a palavra “EU” no nome da música:

Resp:

```
SELECT idMusica  
FROM musica  
WHERE UPPER(nomeMusica) LIKE "%EU %";
```

```
SELECT idCD, Numero_Faixa  
FROM faixa  
WHERE idMusica IN (17,18,36,45,62);
```





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery de múltiplas linhas

- EXEMPLO

Retorne o nome do CD, a faixa e o nome das músicas que tenham a palavra “EU” no nome da música:

Resp:

```
SELECT cd.idCD, cd.Nome_CD, faixa.Numero_Faixa  
FROM cd, faixa  
WHERE faixa.idCD = cd.idCD  
AND faixa.idMusica IN (  
    SELECT idMusica  
    FROM musica  
    WHERE UPPER(nomeMusica) LIKE "%EU %"  
);
```





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery de múltiplas linhas

- EXEMPLO

Retorne o nome do CD, a faixa e o nome das músicas que tenham a palavra “EU” no nome da música:

Resp:

```
SELECT cd.Nome_CD, faixa.Numero_Faixa, musica.NomeMusica
FROM cd, faixa, musica
WHERE faixa.idCD      = cd.idCD
      AND faixa.idMusica = musica.idMusica
      AND faixa.idMusica IN (
                                SELECT idMusica
                                FROM musica
                                WHERE UPPER(nomeMusica) LIKE "%EU %"
                                );
```





SQL – DML – Subqueries

SubQuery com uso da cláusula ANY

- O operador ANY permite o uso dos operadores simples (=, >, <, !=) com uma lista de valores;
- O operador ANY realiza a comparação para cada valor do SELECT EXTERNO com cada valor apresentado na lista retornada pelo SELECT INTERNO;





SQL – DML – Subqueries

SubQuery com uso da cláusula ANY

- EXEMPLO

Retorne o nome do CD e o preço de venda dos CD's que tenham preço de venda inferior a qualquer outro da gravadora com código 2:

Resp:





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery com uso da cláusula ANY

- EXEMPLO

Retorne o nome do CD e o preço de venda dos CD's que tenham preço de venda inferior a qualquer outro da gravadora com código 2:

Resp:

```
SELECT cd.Preco_Venda FROM cd WHERE cd.idGravadora = 2;
```

```
SELECT idCD, cd.Nome_CD, cd.Preco_Venda, cd.idGravadora  
FROM cd
```

```
WHERE idGravadora != 2
```

```
AND Preco_Venda < ANY ( SELECT Preco_Venda  
                        FROM cd  
                        WHERE idGravadora = 2  
                        );
```





SQL – DML – Subqueries

SubQuery com uso da cláusula ALL

- O operador ALL é utilizado em conjunto com os operadores simples: $>$ e $<$;
- O operador ALL realiza a comparação para cada valor do SELECT EXTERNO com cada valor apresentado na lista retornada pelo SELECT INTERNO;
- O resultado do SELECT EXTERNO são as linhas que atendam à condição definida para TODOS os valores da lista retornada pelo SELECT INTERNO;





SQL – DML – Subqueries

SubQuery com uso da cláusula ALL

- EXEMPLO

Retorne o nome do CD e o preço de venda dos CD's que tenham preço de venda inferior à média de preço de venda de TODAS as gravadoras.

Resp:





SQL – DML – Subqueries

SubQuery com uso da cláusula ALL

- EXEMPLO

Retorne o nome do CD e o preço de venda dos CD's que tenham preço de venda inferior à média de preço de venda de TODAS as gravadoras.

Resp:

```
SELECT AVG(Preco_Venda) FROM cd GROUP BY idGravadora;

SELECT Nome_CD, Preco_Venda
FROM cd
WHERE Preco_Venda < ALL (
    SELECT AVG(Preco_Venda)
    FROM cd
    GROUP BY idGravadora
);
```





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery na cláusula FROM

- Podemos utilizar uma subquery na cláusula FROM do comando SELECT;
- Uma subquery na cláusula FROM, funciona como uma tabela. Muito parecido com as Visões (tabelas Temporárias Materializadas)
- EXEMPLO
Retorne o nome do CD, o preço de venda e o preço médio da gravadora.





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery na cláusula FROM

- Podemos utilizar uma subquery na cláusula FROM do comando SELECT;
- Uma subquery na cláusula FROM, funciona como uma tabela. Muito parecido com as Visões (tabelas Temporárias Materializadas)
- EXEMPLO
Retorne o nome do CD, o preço de venda e o preço médio da gravadora.

```
SELECT idGravadora, AVG(Preco_Venda) AS Preco_Medio  
FROM cd  
GROUP BY idGravadora
```





DML SELECT SUBQUERIES

SQL – DML – Subqueries

SubQuery na cláusula FROM

- Podemos utilizar uma subquery na cláusula FROM do comando SELECT;
- Uma subquery na cláusula FROM, funciona como uma tabela. Muito parecido com as Visões (tabelas Temporárias Materializadas)

- EXEMPLO

Retorne o nome do CD, o preço de venda e o preço médio da gravadora.

```
SELECT cd.Nome_CD, cd.Preco_Venda, b.Preco_Medio  
FROM cd, (SELECT idGravadora, AVG(Preco_Venda) AS Preco_Medio  
          FROM cd GROUP BY idGravadora) b  
WHERE cd.idGravadora = b.idGravadora  
      AND cd.Preco_Venda > b.Preco_Medio
```

