

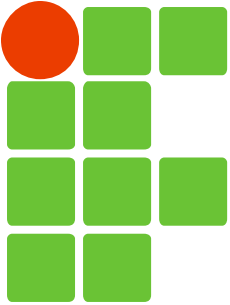
INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Banco de Dados

Consulta SQL - SELECT

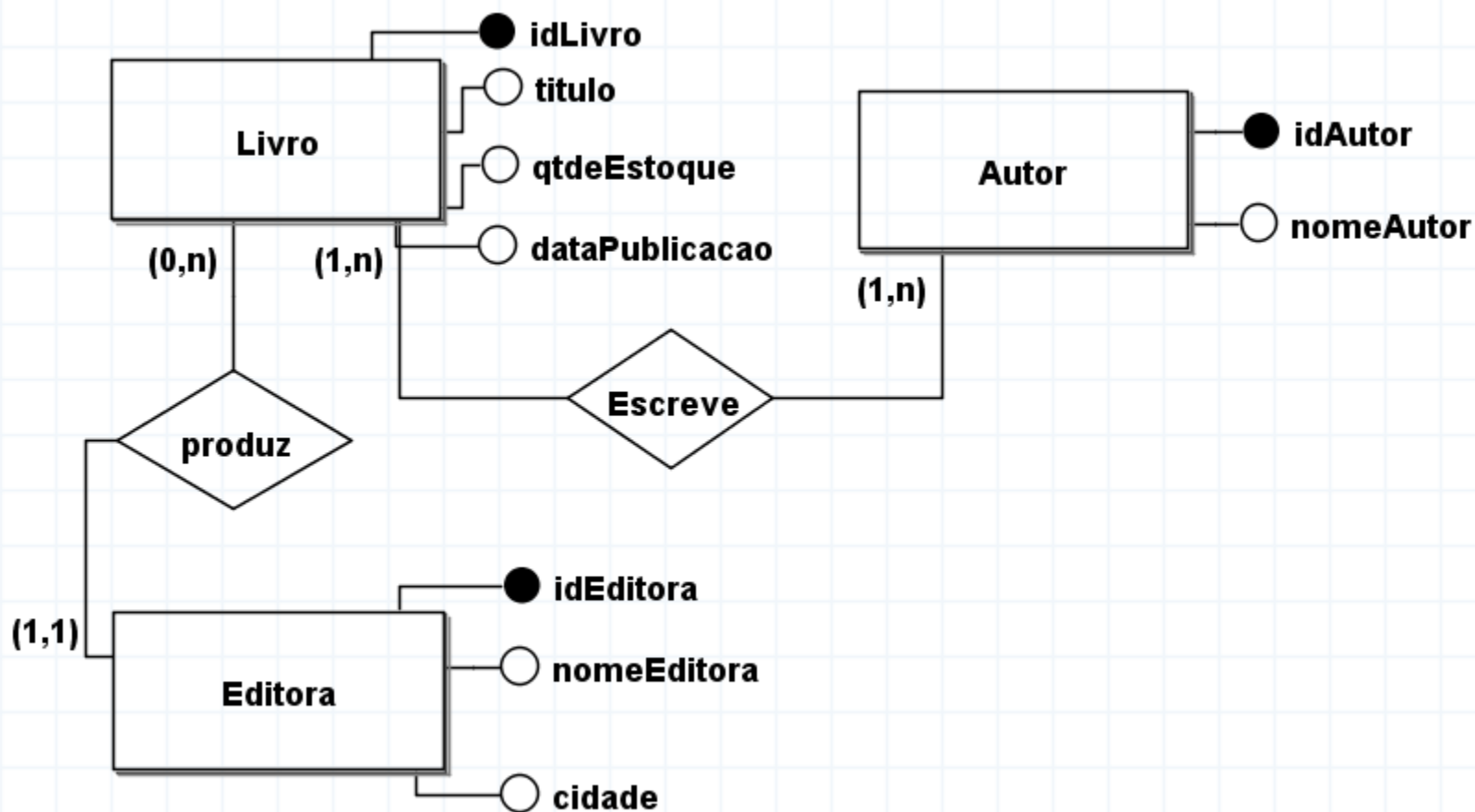
Professor
Igor Carlos Pulini

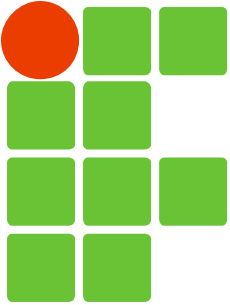


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Exemplo

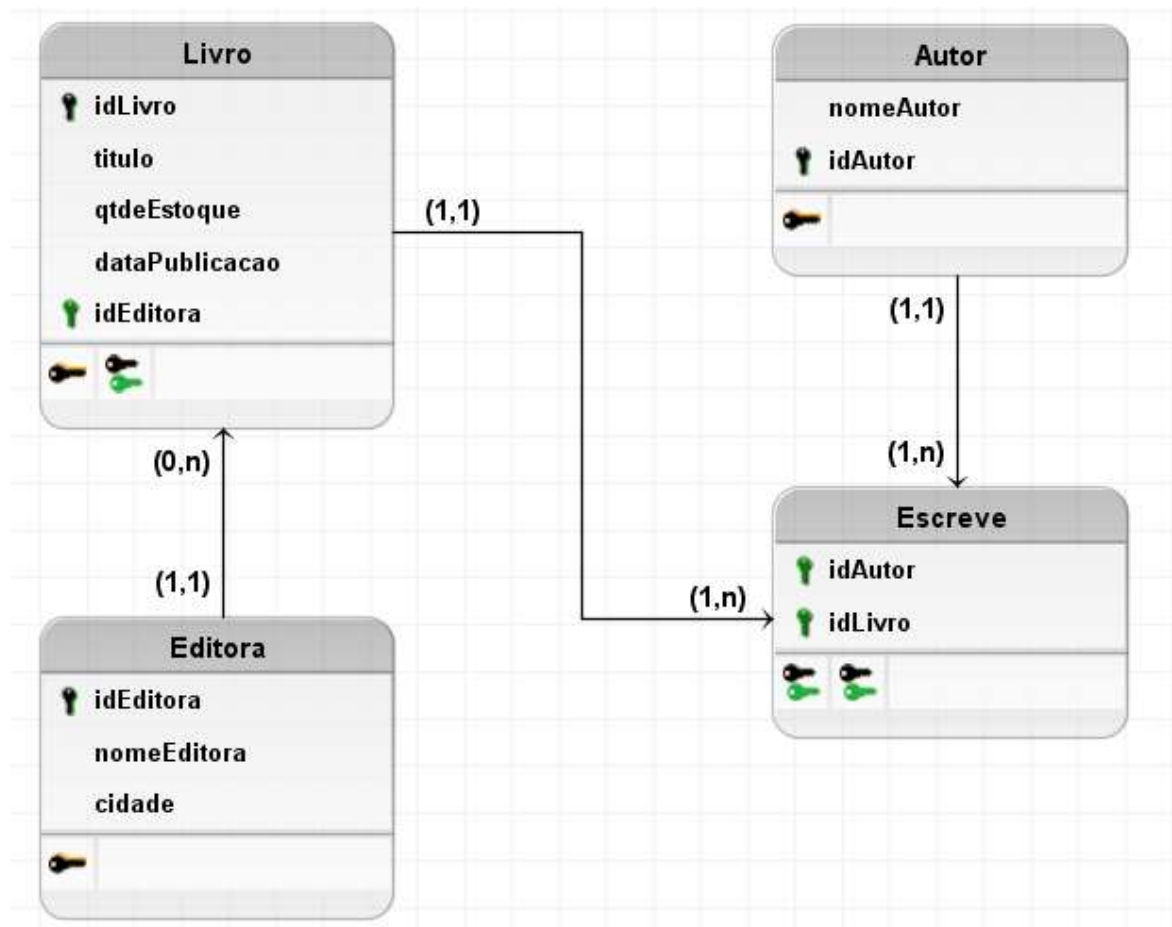


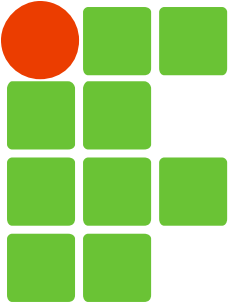


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Exemplo



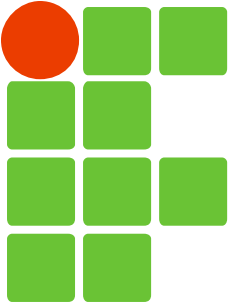


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Select

SELECT	<lista-de-colunas>
FROM	<lista-de-tabelas>
WHERE	<condições>
ORDER BY	<lista-de-colunas>
GROUP BY	<lista-de-colunas>
HAVING	<condições>

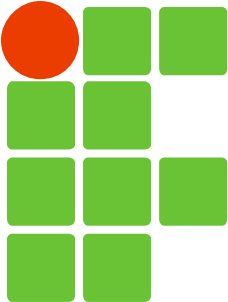


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Select

- [DISTINCT](#) – Seleciona linhas únicas.
- [ORDER BY](#) – Ordena as linhas ([DESC](#), [ASC](#))
- [WHERE](#) – Filtra as Linhas
- [LIMIT/FETCH](#) – Seleciona um subconjunto de Linhas
- [GROUP BY](#) – Faz agrupamentos de determinadas colunas
- [HAVING](#) – Faz filtros pelo agrupamento
- [JOINS](#) – Junta Tabelas
- [UNION](#) – Une o resultado do select de 2 tabelas ([ALL](#))
- [IS NULL](#) – Seleciona valores Nulos
- [AND, OR](#) – E, OU
- [IN](#) – Seleciona de uma lista Ex: where nome in (“Joao”, “Manoel”, “Joana”)
- [IN](#) – Seleciona de uma lista Ex: where nome in (SELECT NOME FROM TABELA)
- [EXCEPT](#) – Retira as linhas retornadas no except



**INSTITUTO FEDERAL
ESPIRITO SANTO**

CAMPUS COLATINA

Select

- JOIN – Une 2 ou mais tabelas

```
select Tabela1.Coluna, Tabela1.Coluna, ..., Tabela2.Coluna from Tabela1 cláusula join Tabela2 on Condição
```

- INNER – Só serão mostrados os registros com referências nas tabelas da junção

```
select a.Codigo, a.Nome, b.Setor from Funcion a inner join Setor b on (b.Codigo = a.Setor)
```

```
select a.Codigo, a.Nome, b.Setor from Funcion a join Setor b on (b.Codigo = a.Setor)
```

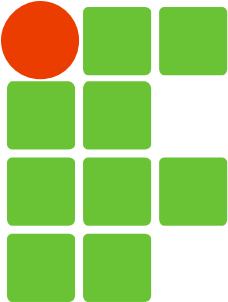
- OUTER – Mostrados todos os registros com ou sem referências nas tabelas da junção

```
select a.Codigo, a.Nome, b.Setor from Funcion a left outer join Setor b on (b.Codigo = a.Setor)
```

```
select a.Codigo, a.Nome, b.Setor from Funcion a right outer join Setor b on (b.Codigo = a.Setor)
```

- FULL – A palavra reservada full indica que todos os registros das duas tabelas serão mostrados, independentemente deles terem ou não referência

```
select a.Codigo, a.Nome, b.Setor from Funcion a full outer join Setor b on (b.Codigo = a.Setor)
```

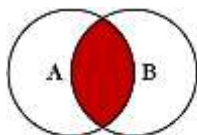


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

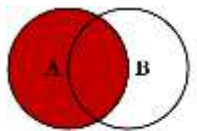
Joins

Inner Join



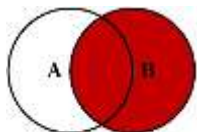
c, d

Left Join



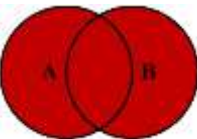
a, b, c, d

Right Join



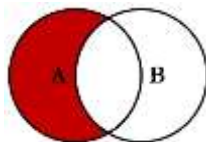
e, f, c, d

Outer Join



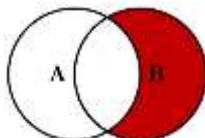
a, b, c, d, e, f

Left Excluding Join



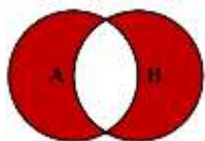
a, b

Right Excluding Join



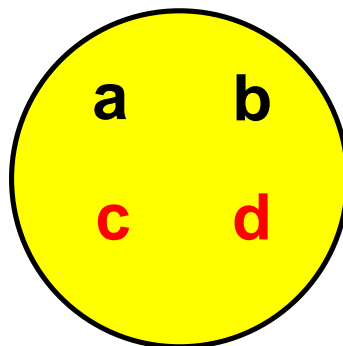
e, f

Outer Excluding Join

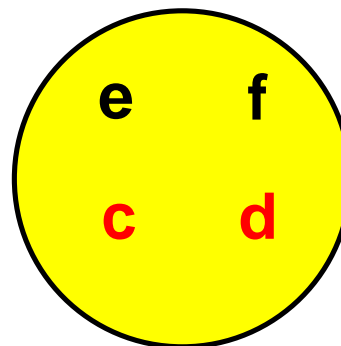


a, b, e, f

TabelaA



TabelaB



```
Drop table TabelaA;
Drop table TabelaB;

CREATE TABLE TabelaA(
  Nome varchar(50) NULL,
  Id integer(5) NULL);
CREATE TABLE TabelaB(
  Nome varchar(50) NULL,
  Id integer(5) NULL);

INSERT INTO TabelaA VALUES('c');
INSERT INTO TabelaA VALUES('d');
INSERT INTO TabelaA VALUES('a');
INSERT INTO TabelaA VALUES('b');
INSERT INTO TabelaA VALUES('e');
INSERT INTO TabelaA VALUES('f');
INSERT INTO TabelaA VALUES('c');
INSERT INTO TabelaA VALUES('d');

--Inner join - Retorna os registros que são comuns de duas tabelas (Implicito e Explícito)
--Explicito
SELECT
FROM TabelaA, TabelaB
WHERE TabelaA.Id = TabelaB.Id and TabelaA.Nome = TabelaB.Nome
--Implicito
SELECT
FROM TabelaA, TabelaB
WHERE TabelaA.Id = TabelaB.Id and TabelaA.Nome = TabelaB.Nome

--Left join - Retorna os registros que estão em A (mesmo que não estejam em B) e os registros de B que não possuem A
SELECT
FROM TabelaA, TabelaB
LEFT JOIN TabelaB ON TabelaA.Id = TabelaB.Id and TabelaA.Nome = TabelaB.Nome
--SQL
SELECT
FROM TabelaA, TabelaB
LEFT OUTER JOIN TabelaB ON TabelaA.Id = TabelaB.Id and TabelaA.Nome = TabelaB.Nome

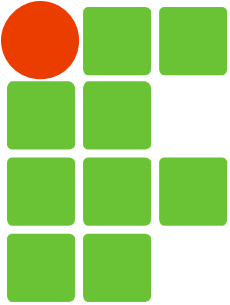
--Right join - Retorna os registros que estão em B (mesmo que não estejam em A) e os registros de A que não possuem B
SELECT
FROM TabelaA, TabelaB
RIGHT JOIN TabelaA ON TabelaA.Id = TabelaB.Id and TabelaA.Nome = TabelaB.Nome
--SQL
SELECT
FROM TabelaA, TabelaB
RIGHT OUTER JOIN TabelaB ON TabelaA.Id = TabelaB.Id and TabelaA.Nome = TabelaB.Nome

--Full join - Retorna os registros de A e B, também conhecido por Full Outer join ou Full join
SELECT
FROM TabelaA, TabelaB
FULL JOIN TabelaB ON TabelaA.Id = TabelaB.Id and TabelaA.Nome = TabelaB.Nome
--SQL
SELECT
FROM TabelaA, TabelaB
FULL OUTER JOIN TabelaB ON TabelaA.Id = TabelaB.Id and TabelaA.Nome = TabelaB.Nome
WHERE TabelaA.Nome is null

--Right Excluding join - Retorna os registros que estão em B e que não estão em A, e os registros de A que não estão em B
SELECT
FROM TabelaA, TabelaB
RIGHT JOIN TabelaB ON TabelaA.Id = TabelaB.Id and TabelaA.Nome = TabelaB.Nome
WHERE TabelaA.Nome is null

--Outer Excluding join - Retorna os registros que estão em B, que não estão em A, e os registros de A que não estão em B
SELECT
FROM TabelaA, TabelaB
FULL JOIN TabelaB ON TabelaA.Id = TabelaB.Id and TabelaA.Nome = TabelaB.Nome
WHERE TabelaA.Nome is null

--CROSS JOIN - Retorna todas as linhas das tabelas por cruzamento, ou seja, para cada linha da tabela A quanteiras todas as linhas da B
SELECT
FROM
CROSS JOIN TabelaB
```



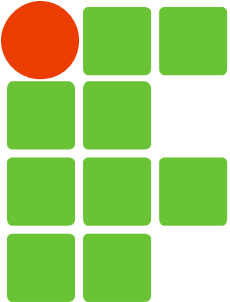
INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

FUNÇÕES DE AGRUPAMENTO

- COUNT – GROUP BY – AS
- MAX
- MIN
- SUM
- AVG



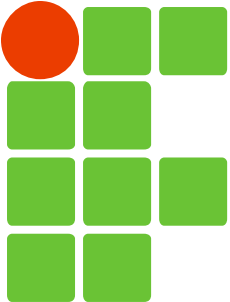


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

O que é uma SubConsulta?

Subconsulta é uma instrução do tipo
SELECT dentro de outra instrução SQL



INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

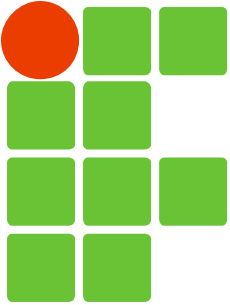
Vantagens e Desvantagens

Vantagens

- Estruturam a consulta para isolar cada parte da instrução
- Executam a mesma operação que normalmente exigiria associações e uniões complexas
- São mais fáceis de ler.

Desvantagem

- Uma desvantagem é que ela leva mais tempo para executar do que uma junção, devido à maneira como o otimizador de banco de dados as processa.



**INSTITUTO FEDERAL
ESPIRITO SANTO**

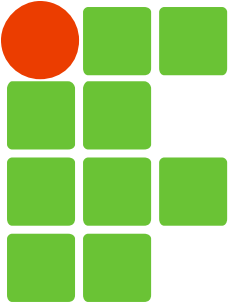
CAMPUS COLATINA

SubConsultas

```
select Codigo, Nome from Cliente where Codigo in ( select Cliente from Pedido )
```

```
select Codigo, Nome from Cliente c where Exists(select 1 from Pedido from pedido  
where c.codigo = pedido.codigo)
```

```
SELECT nome, CASE WHEN nome IN (SELECT nome FROM frutas)  
                THEN 'sim'  
                ELSE 'não'  
                END AS fruta  
FROM alimentos;
```

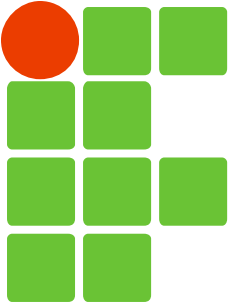


**INSTITUTO FEDERAL
ESPIRITO SANTO**

CAMPUS COLATINA

Concatenar Colunas

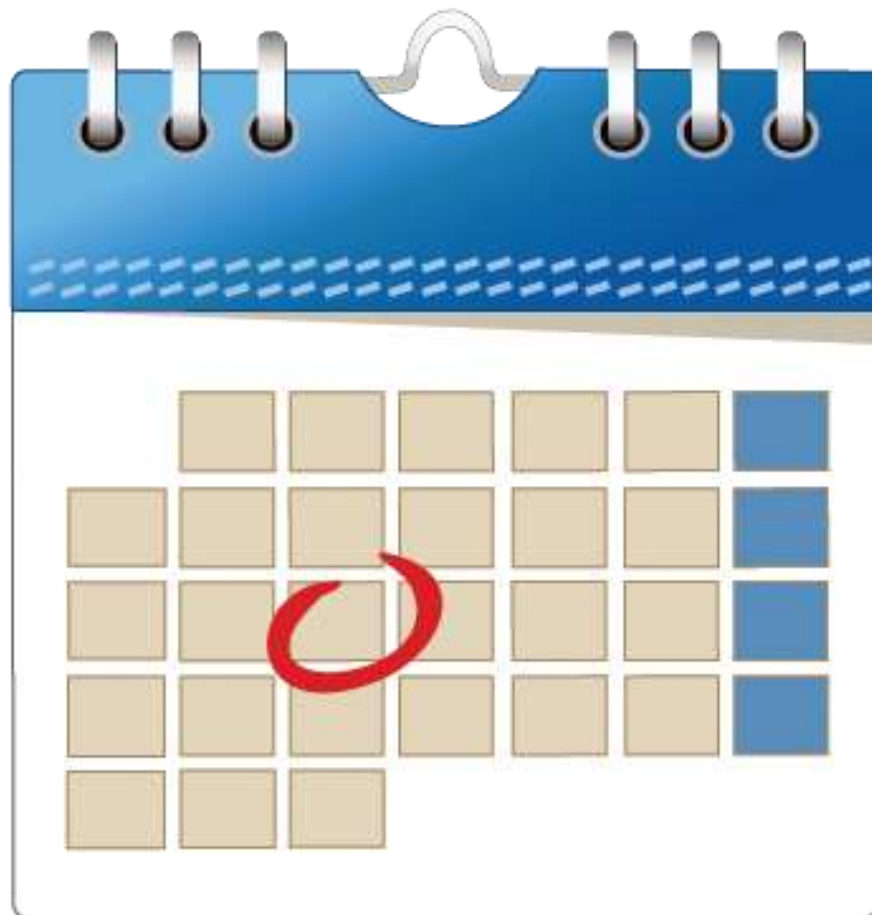
```
select Nome || ' ' || Sobrenome as Nome_Completo from Cliente
```

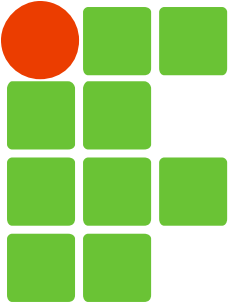


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

FUNÇÕES DATA





INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Fim
