

# SQL

## consultas

---




Carina F. Dorneles  
dorneles@inf.ufsc.br

Parte I

# SQL

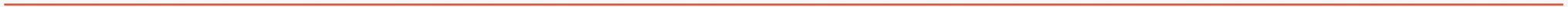
- **DDL:** possui comandos para *implementar* o modelo lógico
    - Comando para **criação** das tabelas
    - Comando para **alteração** das tabelas
    - Comando para **remoção** das tabelas
  - **DML:** possui comandos para *manipular* os dados armazenados no Banco de Dados
    - Comando para **inserção** de dados nas tabelas
    - Comando para **alteração** de dados nas tabelas
    - Comando para **remoção** de dados nas tabelas
    - **Comando para consulta** aos dados nas tabelas
-

# SQL

- **DDL:** possui comandos para *implementar* o modelo lógico
    - Comando para **criação** das tabelas
    - Comando para **alteração** das tabelas
    - Comando para **remoção** das tabelas
  - **DML:** possui comandos para *manipular* os dados armazenados no Banco de Dados
    - Comando para **inserção** de dados nas tabelas
    - Comando para **alteração** de dados nas tabelas
    - Comando para **remoção** de dados nas tabelas
    - **Comando para consulta** aos dados nas tabelas
-

## Voltando um pouco na álgebra.....

- Q1: Recuperar o título e ano dos livros com mais de 350 páginas



## Voltando um pouco na álgebra.....

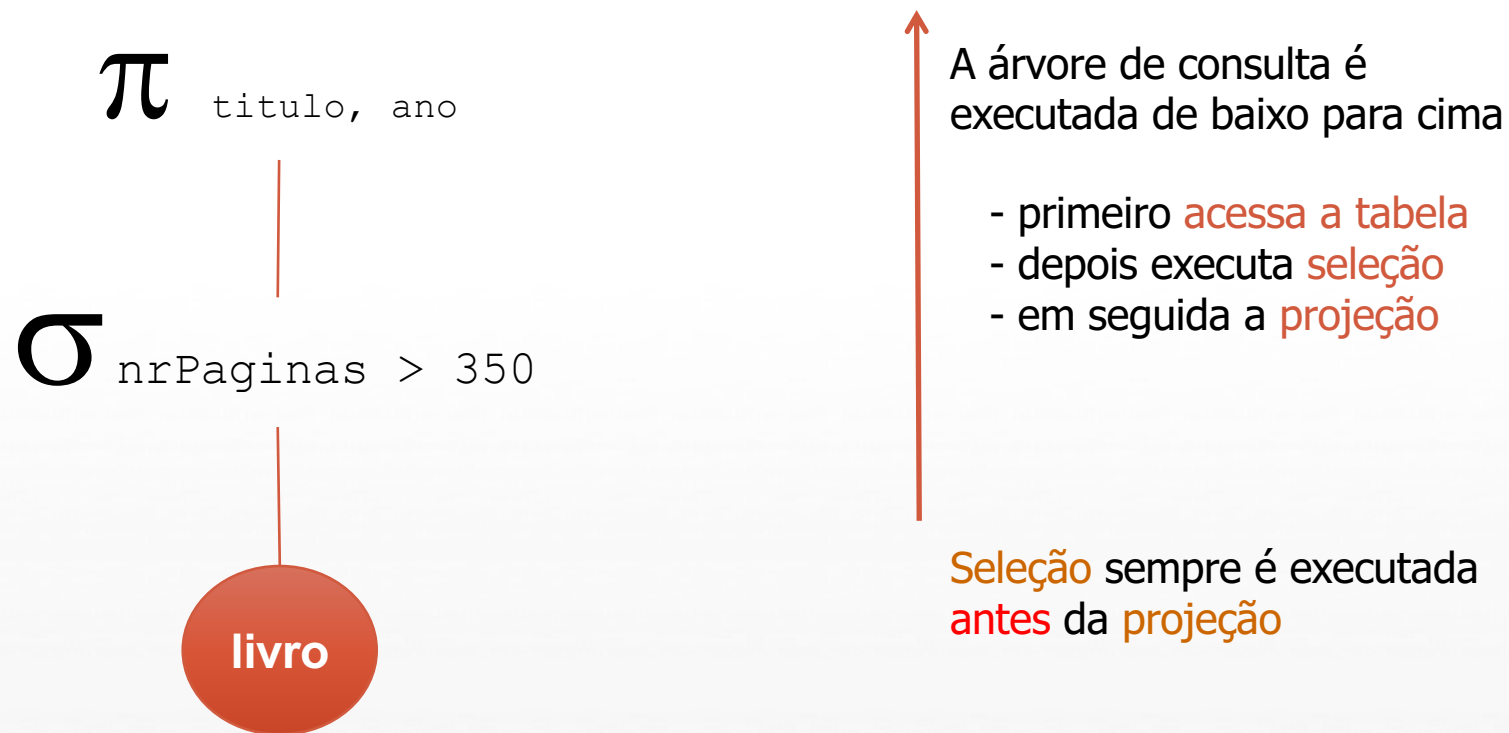
- Q1: Recuperar o título e ano dos livros com mais de 350 páginas

$\pi$  título, ano ( $\sigma$  nrPaginas > 350 (Livro))

| Livro  |                             |      |           |
|--------|-----------------------------|------|-----------|
| Codigo | Título                      | Ano  | NrPaginas |
| LI005  | Web e Banco de dados        | 2013 | 330       |
| LI670  | Introdução a Banco de Dados | 2000 | 500       |
| LI340  | Programação C               | 2012 | 250       |
| LI003  | Algoritmos e Lógica         | 2000 | 700       |



# Execução no Banco de dados





## No SQL

- Q1: Recuperar o título e ano dos livros com mais de 350 páginas

$\pi$  titulo, ano ( $\sigma$  nrPaginas > 350 (Livro))

```
SELECT titulo, ano  
FROM livro  
WHERE nrPaginas > 350
```

---



## No SQL

- Q1: Recuperar o título e ano dos livros com mais de 350 páginas

$\pi$  titulo, ano ( $\sigma$  nrPaginas > 350 (Livro))

```
SELECT titulo, ano  
FROM livro  
WHERE nrPaginas > 350
```

← SELECT representa a  $\pi$

---

## No SQL

- Q1: Recuperar o título e ano dos livros com mais de 350 páginas

$\pi$  titulo, ano ( $\sigma$  nrPaginas > 350 (Livro) )

```
SELECT titulo, ano  
FROM livro  
WHERE nrPaginas > 350
```

← SELECT representa a  $\pi$

← FROM indica a(s) tabela(s)

---

## No SQL

- Q1: Recuperar o título e ano dos livros com mais de 350 páginas

$\pi$  titulo, ano ( $\sigma$  nrPaginas > 350 (Livro) )

```
SELECT titulo, ano  
FROM livro  
WHERE nrPaginas > 350
```

← SELECT representa a  $\pi$   
← FROM indica a(s) tabela(s)  
← WHERE representa a  $\sigma$

---

# Exemplos

“Recuperar **todos os atributos** dos pacientes, com email **ap@a.d.br**”

```
SELECT *  
FROM paciente  
WHERE email = 'ap@a.d.br'
```

```
FROM paciente  
WHERE email = 'ap@a.d.br' AND profissao != 'engenhheiro'
```

“Recuperar nome, endereço e email dos pacientes cuja profissão seja diferente de ‘engenheiros’”

```
SELECT nome, endereco, email  
FROM paciente  
WHERE profissao <> 'engenheiro'
```

---

# Exemplos

“Recuperar **todos os atributos** dos pacientes, com email *ap@a.d.br*”

```
SELECT *  
FROM paciente  
WHERE email = 'ap@a.d.br'
```

“Recuperar o **nome** dos pacientes, com email *ap@abc.de.br* e idade maior que 18

```
SELECT nome  
FROM paciente  
WHERE email = 'ap@abc.de.br' AND idade > 18
```

“Recuperar nome, endereço e email dos pacientes cuja profissão seja diferente de ‘engenheiros’”

```
SELECT nome, endereco, email  
FROM paciente  
WHERE profissao <> 'engenheiro'
```

---

# Exemplos

“Recuperar **todos os atributos** dos pacientes, com email *ap@a.d.br*”

```
SELECT *  
FROM paciente  
WHERE email = 'ap@a.d.br'
```

“Recuperar o **nome** dos pacientes, com email *ap@abc.de.br* e idade maior que 18

```
SELECT nome  
FROM paciente  
WHERE email = 'ap@abc.de.br' AND idade > 18
```

“Recuperar **nome, endereço e email** dos pacientes cuja *profissão seja diferente de 'engenheiros'*”

```
SELECT nome, endereco, email  
FROM paciente  
WHERE profissao <> 'engenheiro'
```

---

# Uso da expressão WHERE

- Operadores de comparação:

=, <>, <, <=, >, >=

<coluna> **BETWEEN** 'valorInicial' **AND** 'valorFinal'

<coluna> **IN** ('valor1', ..., 'valorn')

<coluna> **LIKE** '%banco de %'

<coluna> **LIKE** '19 \_ \_'

<coluna> **IS** [NOT] **NULL**

"\_" = um caracter

"%" = sequência de caracteres

- Operadores lógicos: **AND** e **OR**

---

## Exemplos

- Recuperar todos os dados dos pacientes cujo nome **começa** com A

```
SELECT *  
FROM paciente  
WHERE nome LIKE 'A%'
```

- Recuperar dados dos paciente cujo nome **contenha** 'Silva'

```
SELECT *  
FROM paciente  
WHERE nome LIKE '%Silva%'
```

---



# Exemplos

- Pacientes cuja idade esteja entre 20 e 30

```
SELECT *
```

```
FROM paciente
```

```
WHERE idade BETWEEN 20 and 30
```

```
OU WHERE idade >= 20 AND idade <=30
```

```
nunca "WHERE idade >= 20 AND <=30          /*ERRADO*/"
```

- Pacientes cuja idade esteja entre 20 e 30 e cuja cidade seja diferente de "Blumenau"

```
SELECT *
```

```
FROM paciente
```

```
WHERE (idade BETWEEN 20 and 30) AND cidade <> 'Blumenau'
```

---

# Exemplos

- Pacientes cuja idade esteja entre 20 e 30

```
SELECT *  
FROM paciente  
WHERE idade BETWEEN 20 and 30
```

OU WHERE idade >= 20 AND idade <=30

**nunca** "WHERE idade >= 20 AND <=30 /\*ERRADO\*/"

- Pacientes cuja idade esteja entre 20 e 30 e cuja cidade seja diferente de "Blumenau"

```
SELECT *  
FROM paciente  
WHERE (idade BETWEEN 20 and 30) AND cidade <> 'Blumenau'
```

---

## Exemplos – uso do DISTINCT

- Selecionar a **idade** dos pacientes

```
SELECT idade  
FROM paciente
```

| <i>idade</i> |
|--------------|
| 10           |
| 10           |
| 10           |
| 70           |
| 70           |
| 70           |
| 70           |
| 20           |
| ...          |
| ...          |

- Selecionar a **idade** dos pacientes

```
SELECT DISTINCT idade  
FROM paciente
```

---

## Exemplos – uso do DISTINCT

- Selecione a **idade** dos pacientes

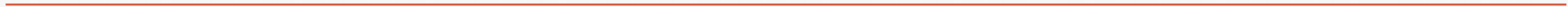
```
SELECT idade  
FROM paciente
```

| <i>idade</i> |
|--------------|
| 10           |
| 10           |
| 10           |
| 70           |
| 70           |
| 70           |
| 70           |
| 20           |
| ...          |
| ...          |

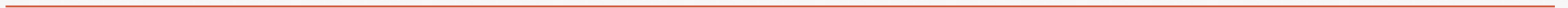
- Selecione a **idade** dos pacientes

```
SELECT DISTINCT idade  
FROM paciente
```


| <i>idade</i> |
|--------------|
| 10           |
| 70           |
| 20           |

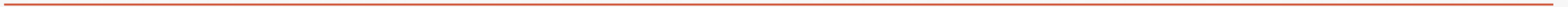


# Juntando tabelas



# Juntando tabelas - álgebra

- Produto cartesiano **X**
- Junção 



# Juntando tabelas - álgebra

Q2: Recuperar título dos livros emprestados

- Produto cartesiano **X**

$\pi$  `Titulo` ( $\sigma$  `codigo = codLivro` (`Livro` `Emprestimo`))

- Junção

$\pi$  `Titulo` (`Livro`  $\bowtie$  `codigo = codLivro` `Emprestimo`)

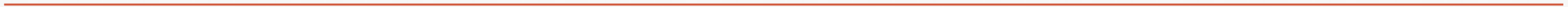
---

# SQL

- Produto Cartesiano

$\pi$  Titulo ( $\sigma$  codigo = codLivro (Livro Emprestimo))

```
SELECT titulo  
FROM livro, emprestimo  
WHERE codigo = codlivro
```





# Juntando tabelas - álgebra

- Junção

$\pi$  `Titulo` (`Livro`  $\bowtie$  `codigo = codLivro` `Emprestimo`)

`SELECT titulo`

`FROM livro JOIN emprestimo ON codigo=codlivro`

---

## Exemplos

- Recuperar título do livro e data de empréstimo, dos livros emprestados a partir de 2000

- Produto cartesiano

```
SELECT titulo, data  
FROM livro, emprestimo  
WHERE codigo = codlivro AND data > '01/01/2000'
```

- Junção

```
SELECT titulo, data  
FROM livro JOIN emprestimo ON codigo = codlivro  
WHERE data > '01/01/2000'
```

---

# Renomeação

- Exemplo usando JOIN
  - Recuperar nome das pessoas e nomes das cidades onde elas moram

```
SELECT p.nome, c.nome
```

```
FROM pessoa p JOIN cidade c ON p.codcid = c.codigo
```

---

## Exemplos

- Recuperar título do livro, data de empréstimo e nome dos usuários, dos livros emprestados a partir de 2000
  - Produto cartesiano

```
SELECT l.titulo, e.data, u.nome
FROM livro l, emprestimo e, usuario u
WHERE l.codigo = e.codlivro AND e.codUsuario = l.codigo
      AND data > '01/01/2000'
```

- Junção

```
SELECT titulo, data
FROM livro l JOIN emprestimo e ON l.codigo = e.codlivro
      JOIN usuario u ON u.codigo = e.codUsuario
WHERE e.data > '01/01/2000'
```

---

# Exercícios

