



# CSS

## GRID Layout e TABLE

### Desenvolvimento WEB I

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas  
Prof. Felipe Scheidt – IFPR – Campus Foz do Iguaçu  
2024

# Sumário

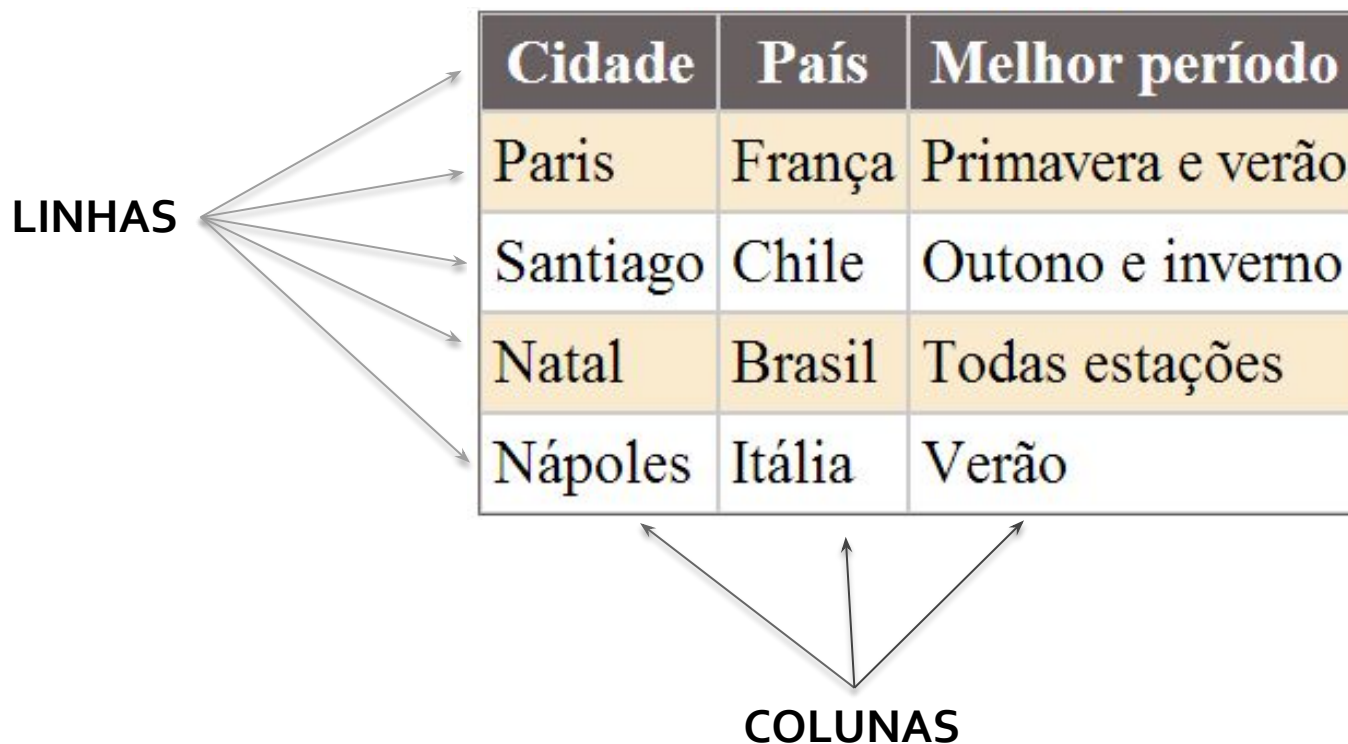
- Representação de dados tabulares
- Grid-layout
- Comparação entre grid e table

<table>

# 1 table

Tabela é uma estrutura muito usada para apresentar **dados**, melhorando a legibilidade da informação.

Os itens da tabela são posicionados em linhas (tr) e colunas (td)



The diagram shows a table with 4 rows and 3 columns. Arrows point from the word 'LINHAS' to each of the four rows. Arrows point from the word 'COLUNAS' to each of the three columns. The table content is as follows:

Cidade	País	Melhor período
Paris	França	Primavera e verão
Santiago	Chile	Outono e inverno
Natal	Brasil	Todas estações
Nápoles	Itália	Verão

# tag <table>

```
<table>
  <tr>
    <td>Linha 1, Coluna 1</td>
    <td>Linha 1, Coluna 2</td>
  </tr>
  <tr>
    <td>Linha 2, Coluna 1</td>
    <td>Linha 2, Coluna 2</td>
  </tr>
</table>
```

<tr> inicia uma nova linha

<td> cria uma coluna

</tr> fecha a linha

Linha 1, Coluna 1	Linha 1, Coluna 2
Linha 2, Coluna 1	Linha 2, Coluna 2

# Tags usadas em tabelas

**<table>** : inicia uma nova tabela

**<tr>** : define uma nova linha da tabela (*table row*)

**<th>** : nome da coluna (*table headings*)

**<td>** : define uma célula da tabela (*table data*)

**<thead>** : área do cabeçalho

**<tbody>** : área do conteúdo principal da tabela

**<tfoot>** : rodapé da tabela

**<caption>** : atribui um título ou legenda.

# Cabeçalho da tabela

A tag **<th>** define uma coluna do cabeçalho. É boa prática colocar um cabeçalho para melhorar a

legibilidade dos dados. É comum ter apenas uma linha com **<th>**

**<td>** Define uma célula na coluna x, linha y

**<tr>** inicia uma nova linha

```
<table>
  <tr>
    <th>Modelo</th>
    <th>Fabricante</th>
  </tr>
  <tr>
    <td>Palio</td>
    <td>Fiat</td>
  </tr>
  <tr>
    <td>Corsa</td>
    <td>Chevrolet</td>
  </tr>
</table>
```

# Definindo CSS da tabela

Background das células

```
td{  
    background-color: yellow;  
}
```

background da tabela

```
table{  
    background-color: #CCC;  
}
```

Bordas das células

```
td{  
    border: 1px solid black;  
}
```

Espaçamento das células

```
table {  
    border-spacing: 2px;  
}
```



**GRID**

## ② Grid layout

Técnica de estruturação de layout tabular orientado a linhas e colunas semelhante a tabela, com as seguintes características:

- É responsivo
- Prático para alterar a estrutura (quantidade de linhas e colunas).
- É uma propriedade, não sendo restrito à uma tag
- É uma forma mais “natural” de visualizar do que o flex

# 1) Estrutura HTML do Grid

Para desenhar o layout com grid começamos pela **quantidade de colunas**.

Definir a tag que será o grid container (agrupa os itens).

Cada item do grid, é chamado de grid items

```
<div class="grid-container">  
  <div class="item">01</div>  
  <div class="item">02</div>  
  <div class="item">03</div>  
</div>
```

# display: grid

Para tornar um elemento num grid-container, define-se duas propriedades:

- **display: grid**

*o elemento passa a ser um grid*

- **grid-template-columns: 200px 300px;**

*define a quantidade de colunas e largura*

# Propriedades do grid-container

- **align-items:center**

*define o alinhamento vertical (padrão)*

- **gap:1rem**

*define o espaço entre as colunas e linhas*

- **justify-items:center**

*define o alinhamento horizontal*

# Propriedades do `grid-item`

Propriedades para ajuste dos items da `grid-container`:

- **`grid-column:1/3`**

*posição/coluna que o item será colocado.*

- **`grid-row:1`**

*item deve ser colocado na 1ª linha*

# grid-template-columns

Propriedade que define as dimensões do elemento grid.

Exemplos:

- **grid-template-columns: 100px 200px 130px;**

*Define 3 colunas com tamanhos fixos (100, 200 e 130)*

- **grid-template-columns: repeat(4, 300px);**

*Cria 4 colunas, usando repeat e aplicando a mesma largura (300)*

- **grid-template-columns: 1fr 1fr;**

*Nesse caso a coluna é responsiva – espaço disponível (fração da tela)*

- **grid-template-columns: max-content min-content;**

*max-content – usa o espaço máximo ocupado pelos items*

# Exemplo de tabela com grid

Criar a tabela abaixo com **grid-layout**

Análise inicial: qual a quantidade de colunas?

Nome	Marca	Consumo
Fox	VW	14 km/l
Gol	VW	15 km/l
Golf	VW	11 km/l



# Grid area

Também podemos usar grid para criar o layout da página. Com a propriedade **grid-template-areas**, é possível definir um nome para cada área da página. No exemplo ao lado temos 3 áreas que podem ser representadas com uma grid-area:

```
body {
```

```
    grid-template-areas: "header" "main" "footer";
```

```
}
```

O elemento <footer> é inserido na grid-area “footer” com a regra CSS:

```
footer {
```

```
    grid-area: footer;
```

```
}
```



# Exercício 1

Faça a tabela a seguir com **grid** layout.

Modelo	Consumo
Clio	15,8 km/l
Palio Fire	15,4 km/l
Novo Uno	15,2 km/l
Up	14,3 km/l

**Cores:**  
#98FF98  
#CCFFFF  
#6E6E6E

## Exercício 2

Faça a tabela abaixo usando **<table>**

Vendas do 1º Semestre	
Modelo	Unidades
Palio	585.240
Gol	529.894
fonte: agencia Reuters	

**Cores:**

rgb(243, 213, 124);

#E6e6e6;

#97CBFA;

## Exercício 3

Faça a tabela ao lado usando <table>.

Cores:  
#6E6E6E

Imagem	Modelo	Consumo
	Clio	15,8 km/l
	Palio Fire	15,4 km/l
	Novo Uno	15,2 km/l
	Up	14,3 km/l

## Exercício 4

Use **grid** para criar a tabela:

15ª Rodada				
Goiás	0	x	1	Inter
Palmeiras	1	x	0	São Paulo
Cruzeiro	3	x	0	Santos
Botafogo	2	x	0	Fluminense
Série A		2014		

# Exercício 5

Crie a estrutura da tabela abaixo usando grid e unidades de medida relativas como o fr (fraction) para aproveitar o espaço de tela disponível.

## Casas no centro



## Casas na vila A



**Cores :**

#CCC

#5EFB6E

# Exercício 6

Use Grid para fazer o layout da página abaixo. As divisões (topo, menu, menu lateral, conteúdo e rodapé) devem posicionadas com **grid-area**

## Desenvolvimento de sistemas

[google](#) | [ifpr](#) | [uol](#)

- produtos
- clientes
- help-desk
- informações

### Introdução

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

### Desenvolvimento

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Site desenvolvido por tectronica S.A.