

# CSS

## [2] Modelo box e variáveis globais

### Desenvolvimento WEB I

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

Prof. Felipe Scheidt – IFPR – Campus Foz do Iguaçu

2023

# Tópicos

- Display inline e block
- Herança e especificidade de regras
- Unidades de medidas
- Variáveis do CSS

# Propriedade display

Permite definir alterar o comportamento padrão que um elemento é renderizado na tela.

Basicamente, um elemento html quando não vinculado a um container flex ou grid, pode ser classificado em 3 tipos:

- inline
- inline-block
- block

# Propriedade display

A propriedade display possui muitas possibilidades de valores, alguns dos quais já foram usados: flex, grid, list-item,...

Podemos ocultar um elemento usando **display:none**

Para torná-lo visível, restauramos o valor original do display.

Exemplo: oculta os parágrafos quando ocorrer mouse-over

```
p:hover {  
    display:none;  
}
```

# Display Block

São elementos que iniciam uma nova linha ocupando todo espaço disponível nesta.

Por padrão as seguintes tags são block

- `<div>`
- `<p>`
- `<h1>`, `<h2>`
- `<form>`

# Display inline

Nesse caso, a tag não inicia uma nova linha, compartilhando o espaço disponível na linha com outras tags se estas forem também inline.

Por padrão as seguintes tags são inline:

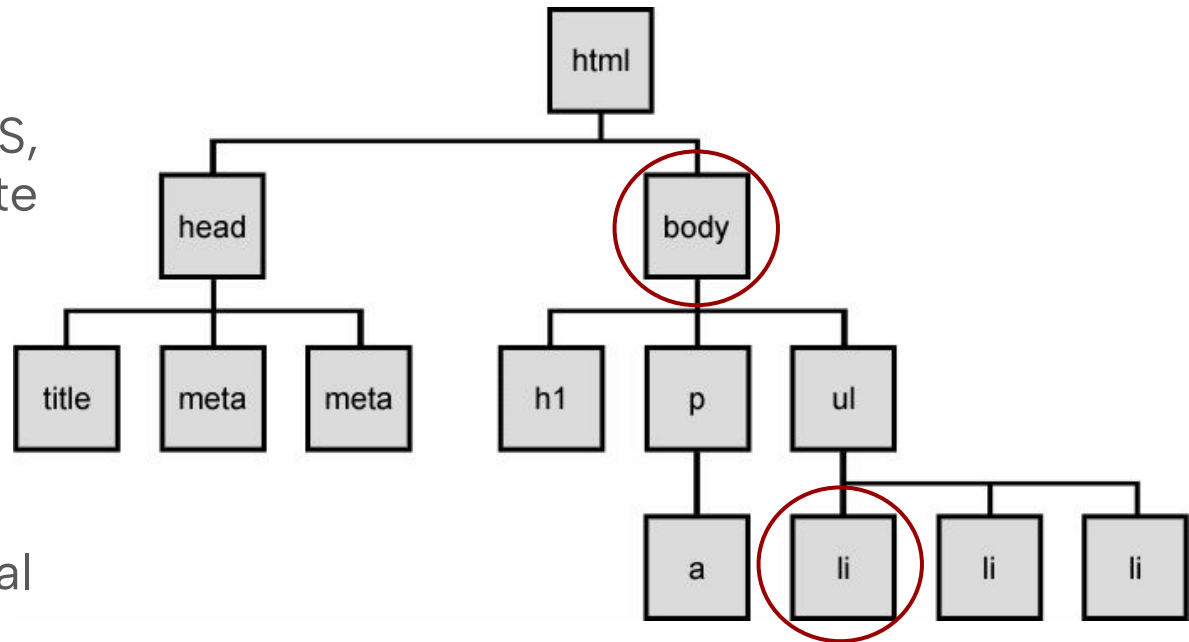
- `<span>`
- `<a>`
- `<img>`

# DOM – Document object model

Páginas HTML são representadas por um modelo chamado **DOM** (árvore de elementos).

Esse modelo vale para o CSS, HTML e JS, sendo importante compreender a estrutura hierarquica formada pelas tags.

Um regra CSS pode ser escrita considerando o caminho desde o nodo inicial até o nodo alvo da regra.



li  
body ul l  
body > ul > li

# Herança vs. Especificidade

Regras podem ser herdadas na hierarquia de tags.

Quando se define uma cor de fonte na tag body, todos os descendentes herdam essa propriedade.

Porém se uma regra mais específica é definida, a regra herdada do ancestral é sobrescrita.

Regras CSS também podem ser definidas para uma área específica do html, formando um caminho no DOM:

```
main .conteudo form.atualizar label{  
  color: ■ navy;  
}  
main .conteudo form.cadastro label{  
  color: ■ red;  
}
```



# Unidades de medida

Existem dois tipos de unidades de medida:

- **Relativa:** o tamanho do elemento é relativo (depende) ao tamanho da viewport ou do container (parent).
- **Absoluta:** o tamanho é fixo independente do tamanho da tela ou parent, preservando assim as suas dimensões.

# Exemplo unidades

nome	sigla	exemplo	métrica
centimeters	cm	1cm;	absoluta 1cm = 37.8px
pixel	px	14px;	absoluta
root element	rem	1rem;	relativa 1rem ≈ 16px
parent element	em	1em;	relativa font-size
percentage	%	50%;	relativa se width=200px(parent) 50%=100px
viewport width	vw	1vw;	relativa 1vw = 1% viewport
viewport height	vh	1vh;	relativa

Mais unidades: [https://developer.mozilla.org/en-US/docs/Learn/CSS/Building\\_blocks/Values\\_and\\_units](https://developer.mozilla.org/en-US/docs/Learn/CSS/Building_blocks/Values_and_units)

## Box model e box-sizing

No modelo box model usado no CSS a dimensão (tamanho) de um elemento html é definido pelo width e height, **somando-se** o padding e border caso existam.

Isso pode ser indesejado, pois não é claro qual será o tamanho final desse elemento.

Para determinar com precisão o tamanho que o elemento ocupará, usa-se a propriedade **box-sizing** que incorporar o espaço do padding e border ao tamanho final do elemento.

# Box model e box-sizing

## padrão: content-box

```
width: 300px;  
box-sizing: content-box  
width real: 340px  
height real: 180px
```

## com border-box

```
width: 300px;  
box-sizing: border-box  
width real: 300px  
height real: 140px
```

```
.box {  
  box-sizing: border-box;  
}
```

# Variáveis globais

Valores de propriedades do CSS tendem a se **repetir** tornando custosa a manutenção do código.

Por exemplo, uma propriedade como cor de fundo pode facilmente aparecer em várias regras css.

Uma boa prática é **atribuir/isolar** esse valor para uma variável global, e na regra CSS colocar uma referência para a variável.

A função **var()** permite obter o valor da variável.

```
:root {  
  --font: 18px;  
  --text-color:  #e6e6e7;  
  --bg-color:  indigo;  
}
```

```
p {  
  background: var(--bg-color);  
}
```

# Exercício 1

Considere o exercício 6 da lista de layout (imagem abaixo) e escreva o código CSS utilizando agora **variáveis globais**. Identifique os valores das propriedades e defina as respectivas variáveis. Propriedades nesse exemplo incluem ser a cor de fundo, família da fonte, cor da fonte, padding, margin, etc...



## Introdução

---

### VANTAGENS DO SOFTWARE

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.

## Desenvolvimento

---

### METODOLOGIA DE 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.