



# **PROGRAMAÇÃO PARA INTERNET RICA**

## **FUNDAMENTOS DO CSS – PARTE II**

Prof. Dr. Daniel Caetano

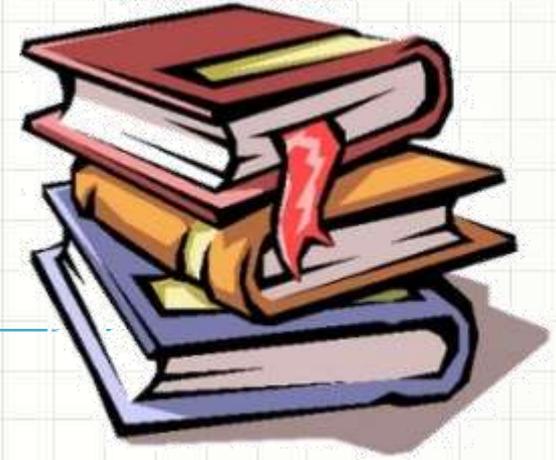
2012 - 2

# Objetivos

- Apresentar pseudo-elementos e pseudo-classes do CSS
- Apresentar o CSS como ferramenta de layout de página
- Capacitar para o posicionamento de elementos com o uso de CSS
- **TRABALHO A ONLINE!**



# Material de Estudo



## Material

## Acesso ao Material

Notas de Aula

<http://www.caetano.eng.br/>  
(Aula 5)

Apresentação

<http://www.caetano.eng.br/>  
(Aula 5)

Material Didático

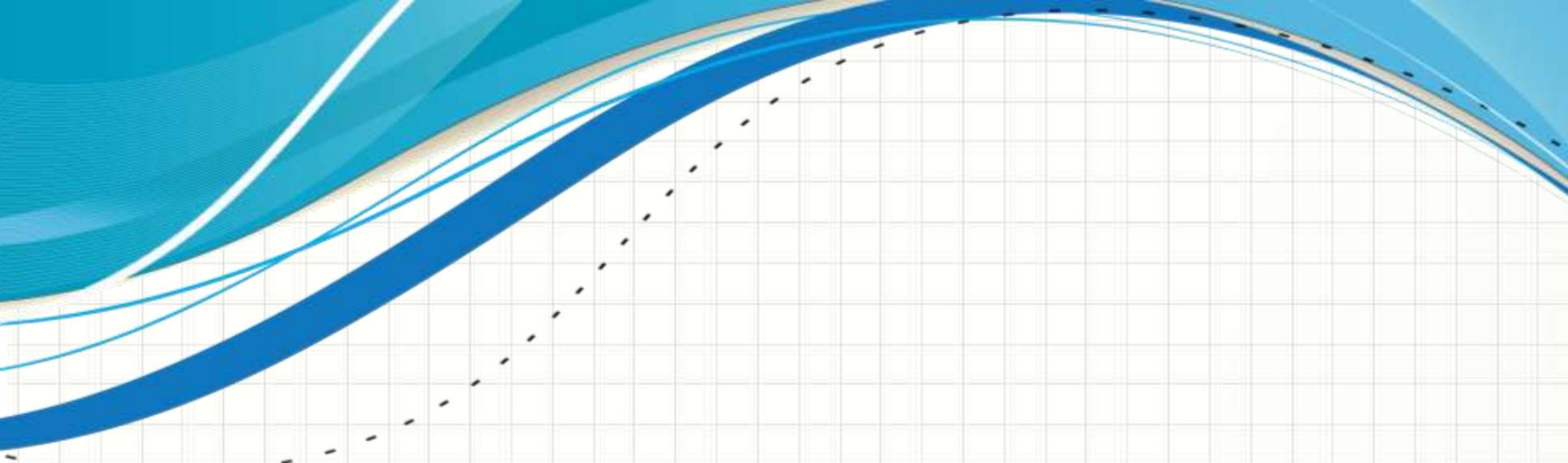
-

Google

+“CSS” +tableless +tutorial

Web Sites

<http://www.w3.org/>



# **ESTILOS AVANÇADOS**

# Estilos Avançados

- Alguns tipos de estilos não foram discutidos
  - Estilos associados a ações e à ordem dos elementos
  - Estilos de posicionamento
- Iremos estudá-los nesta aula

# Pseudo-Classes

- Formas de definir estilos para **estados** dos elementos
- Como faço para definir estilo para um elemento que ganhou foco?
- Como posso definir o estilo para um elemento cujo mouse esteja em cima?
- seletor:estado {  
    propriedade: valor;  
}

# Pseudo-Classes

- Pseudo-Classes mais comuns

|            |                                    |
|------------|------------------------------------|
| – :link    | Links não visitados                |
| – :visited | Links já visitados                 |
| – :hover   | Mouse sobre o elemento             |
| – :active  | Elemento ativo (botão pressionado) |
| – :focus   | Elemento com foco ativo            |

Links

Tudo

# Pseudo-Classes

- Exemplos de Pseudo-Classes

```
a:link {  
    color: green;  
    text-decoration: none;  
}
```

```
a:hover {  
    color: red;  
    text-decoration: underline;  
}
```

# Pseudo-Classes

- Exemplos de Pseudo-Classes

```
input:focus {  
    background-color: yellow;  
}
```

```
#menu:hover {  
    background-color: blue;  
}
```

# Pseudo-Elementos

- Definir estilos para elementos que apareçam em determinada **ordem** (posição)
- E se eu quiser definir um estilo para o primeiro elemento de uma lista?
- Ou definir algo diferente para o último?
- seletor: **descritor** {  
    propriedade: valor;  
}

# Pseudo-Elementos

- Pseudo-Elementos mais comuns
  - :first-child                      Primeiro elemento
  - :last-child                      Último elemento
  - :first-letter                      Primeira letra
  - :first-line                      Primeira linha
  - :nth-child(regra)              “Enésimo” elemento

# Pseudo-Elementos

- Exemplos

```
p:first-letter {  
    font-size: 1.2em;  
}
```

```
p:first-line {  
    font-variant: small-caps;  
}
```

```
li.menu:last-child {  
    border-right: none;  
}
```

# Pseudo-Elementos

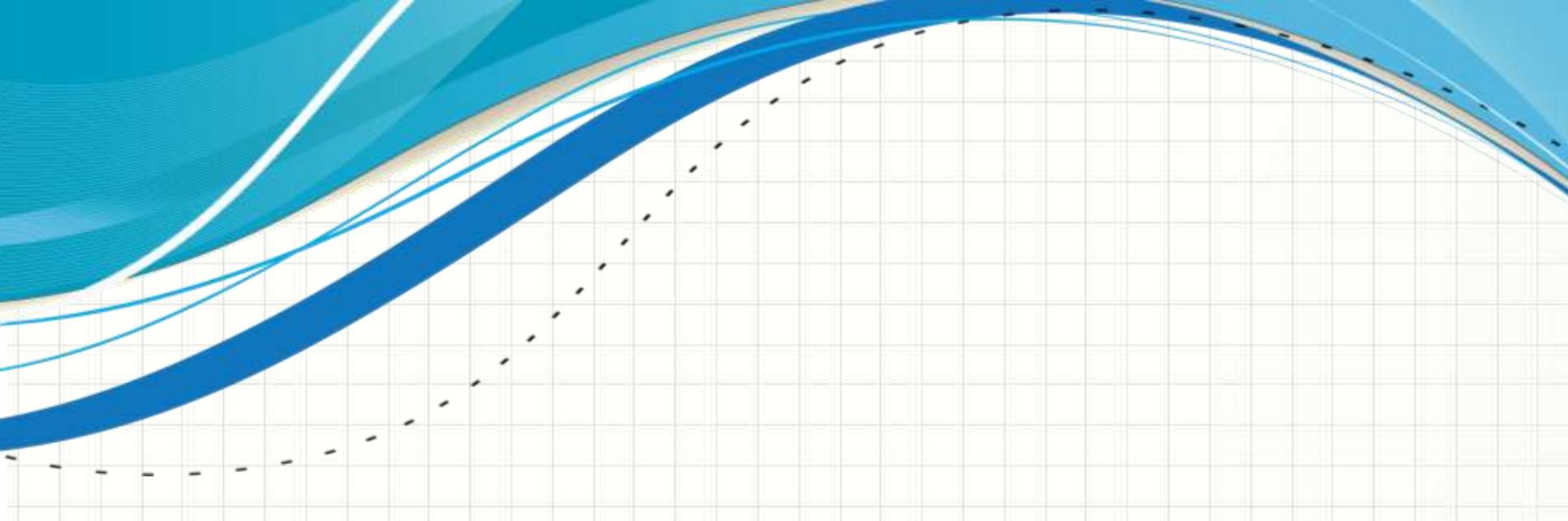
- Exemplos

```
tr:nth-child(2n+1) td {  
    background-color: white;  
}
```

*(A partir da posição 1, de dois em dois)*

```
tr:nth-child(2n+2) td {  
    background-color: gray;  
}
```

*(A partir da posição 2, de dois em dois)*



# **POSICIONAMENTO COM O CSS: MOTIVAÇÃO**

# Exemplo de Mudança de Layout

Minha Página Pessoal HTML com CSS

**TITULO**

## INDICE

1. [INFORMAÇÕES PESSOAIS](#)
2. [NOTAS DE PROGRAMAÇÃO WEB](#)
3. [CONTATO](#)

**MENU**

## 1. INFORMAÇÕES PESSOAIS



**CONTEUDO**

Meu nome é Daniel Caetano e sou sócio-fundador da empresa Amusement Factory Software, sendo também professor das seguintes disciplinas:

- Programação Web
- Segurança e Auditoria de Sistemas
- Pesquisa Operacional
- Arquitetura de Computadores
- Sistemas Operacionais

# Exemplo de Mudança de Layout

Minha Página Pessoal com CSS Posicional

ÍNDICE

**MENU**

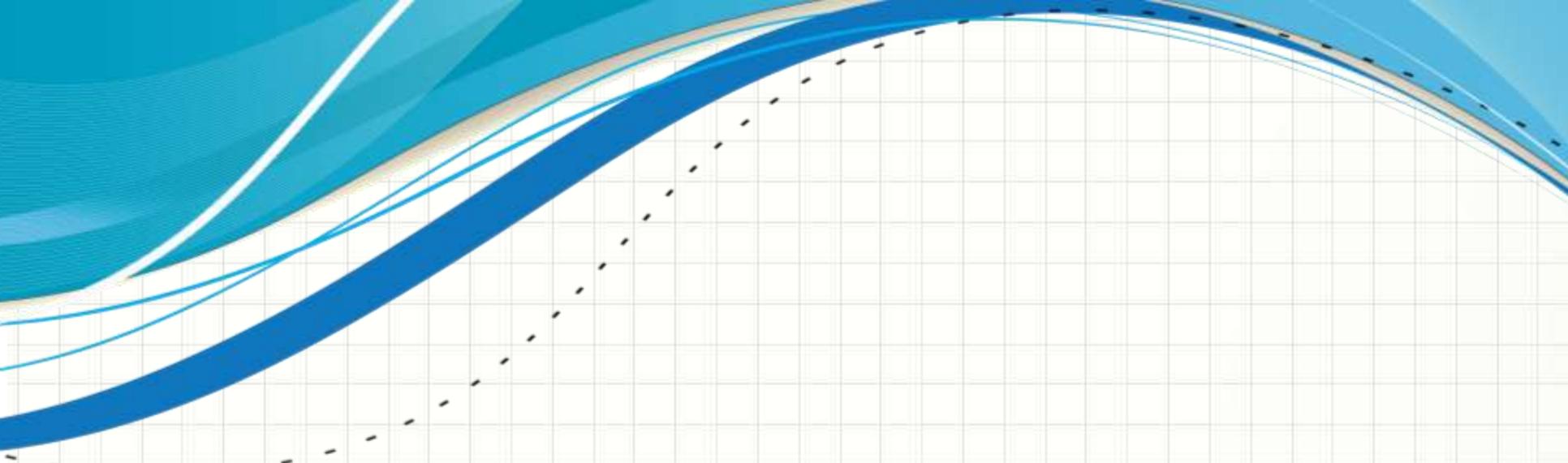
1. [INFORMAÇÕES PESSOAIS](#)
2. [NOTAS DE PROGRAMAÇÃO WEB](#)
3. [CONTATO](#)

## 1. INFORMAÇÕES PESSOAIS



Meu nome é Daniel Castano e sou sócio-fundador da empresa Amusement Factory. Sou também professor das seguintes disciplinas:

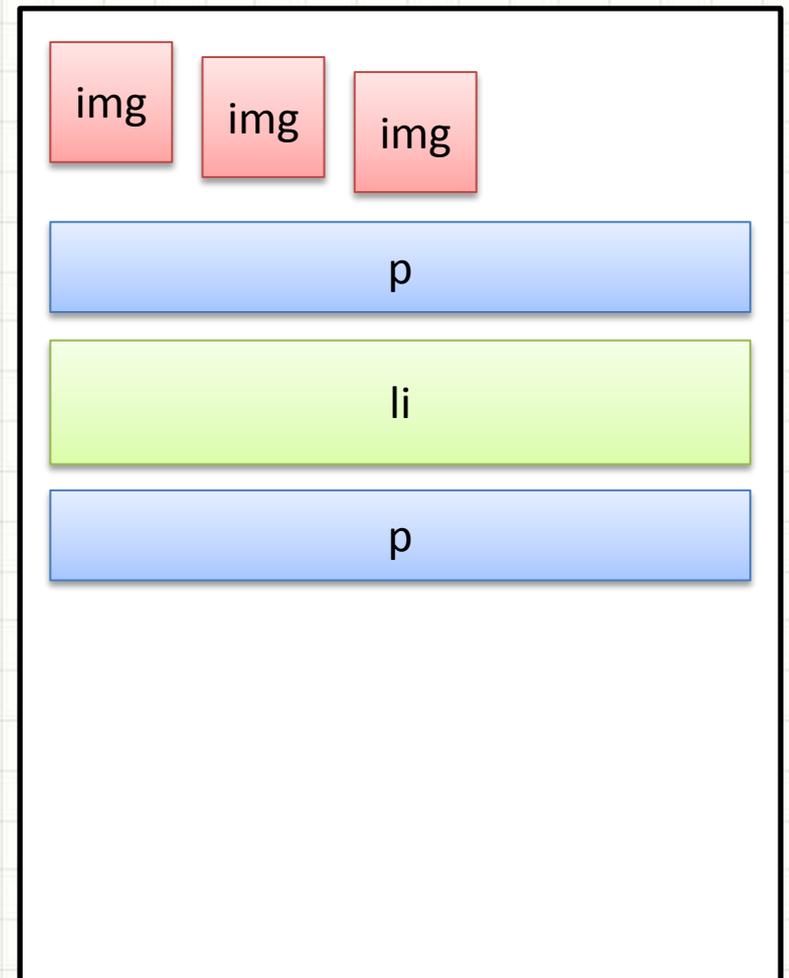
- Programação Web
- Segurança e Auditoria de Sistemas
- Pesquisa Operacional
- Arquitetura de Computadores
- Sistemas Operacionais



# **POSICIONAMENTO COM O CSS: TIPOS DE POSICIONAMENTO**

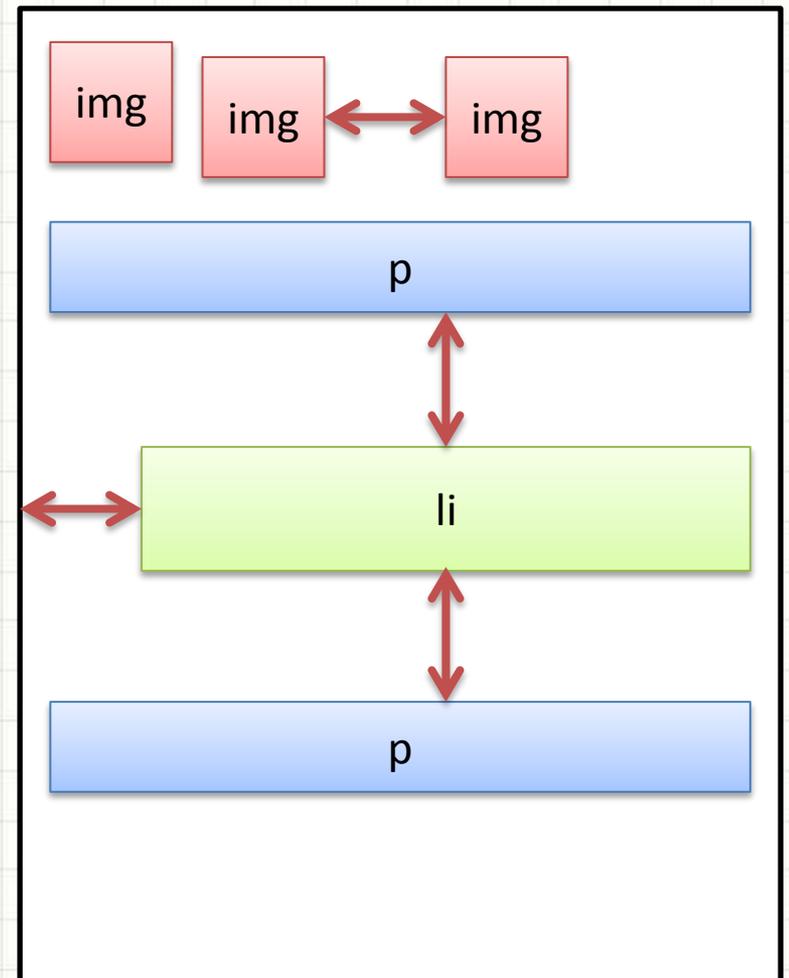
# Posicionamento + CSS: **position**

- Atributo **position: static**
- Base: outros elementos
- Posicionamento padrão
  - Elementos pequenos
    - Lado a lado
  - Elementos grandes
    - Um em baixo do outro



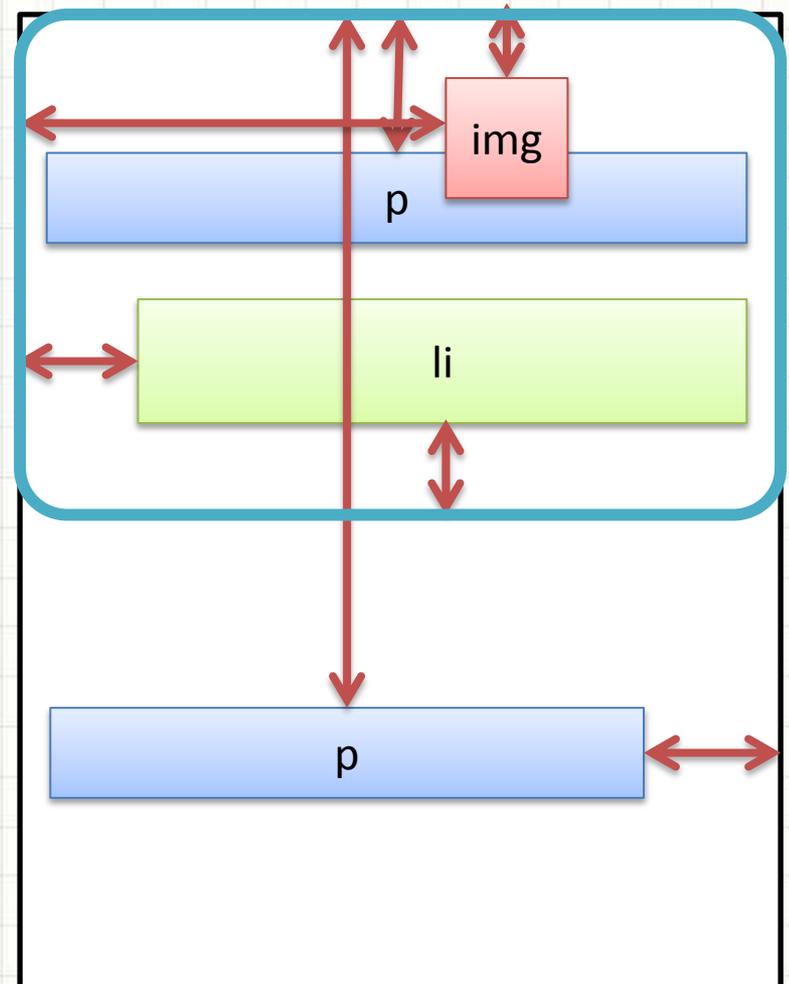
# Posicionamento + CSS: **position**

- Atributo **position: static**
- Espaçamento
  - margin
- Centralizar
  - margin-left: auto;
  - margin-right: auto;
- Tamanho
  - Conteúdo, padding, border, width, height



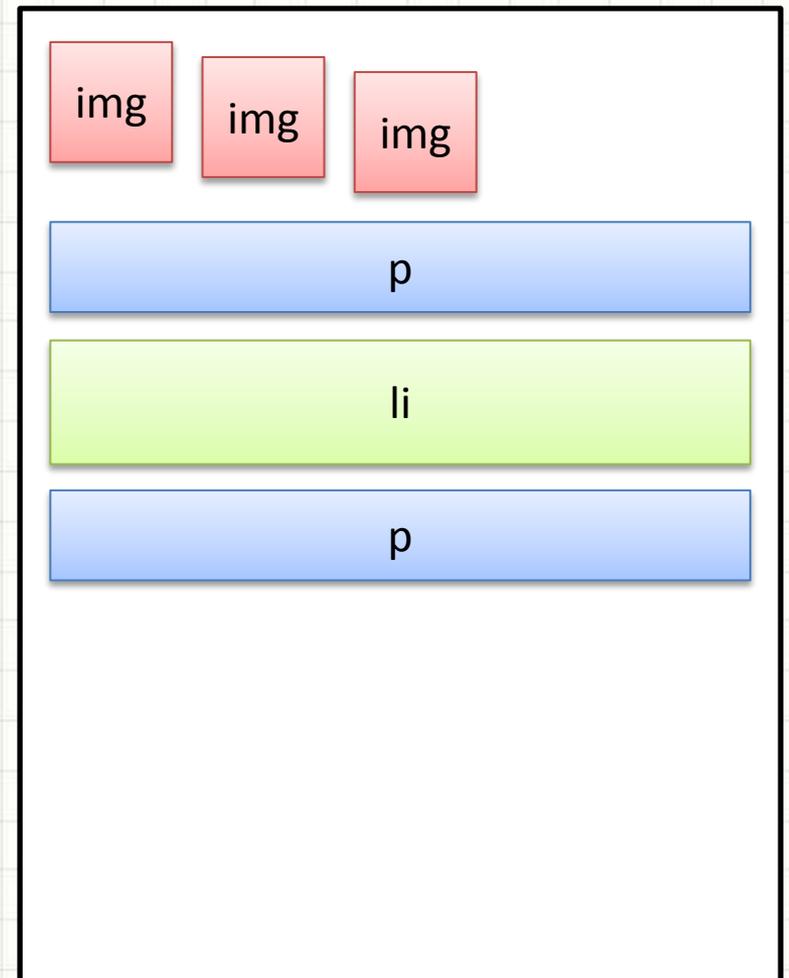
# Posicionamento + CSS: **position**

- Atributo **position: absolute**
- Base: janela
  - Rola com documento
- Espaçamento
  - margin, left, right, top  
bottom
- Tamanho
  - Conteúdo, padding,  
border, width, height



# Posicionamento + CSS: **position**

- Atributo **position: relative**
- Elemento relativo...
  - Posiciona-se como static
- Para que serve?
  - Muda comportamento dos absoletes internos!



# Posicionamento + CSS: **position**

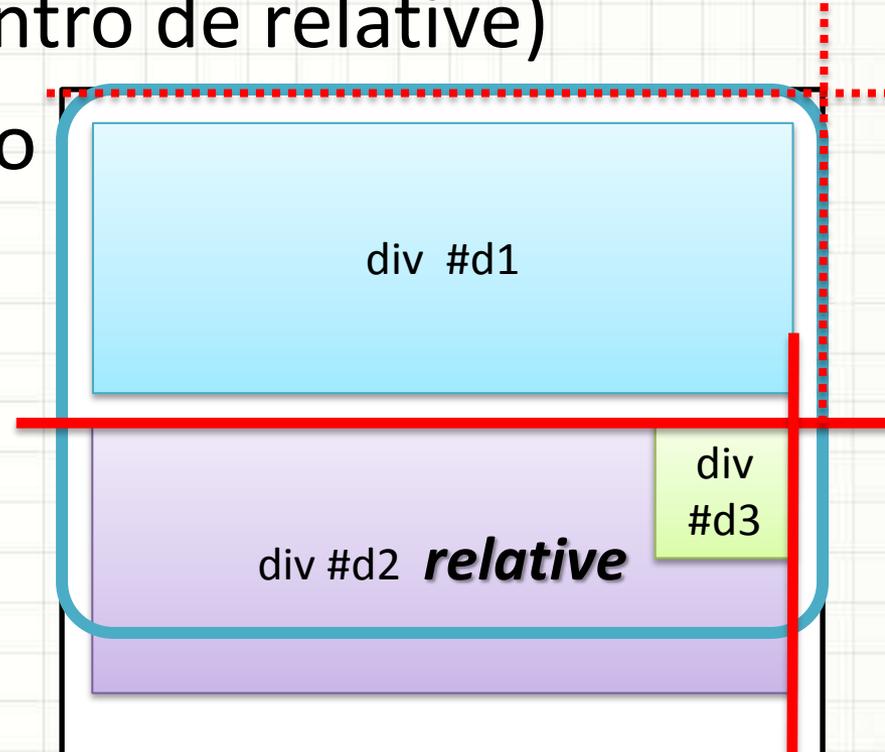
- **position: absolute** (dentro de relative)
- Base: elemento relativo
- Muda a referência!
  - margin, left, right, top  
bottom

```
<div id="d1"></div>
```

```
<div id="d2">
```

```
  <div id="d3">
```

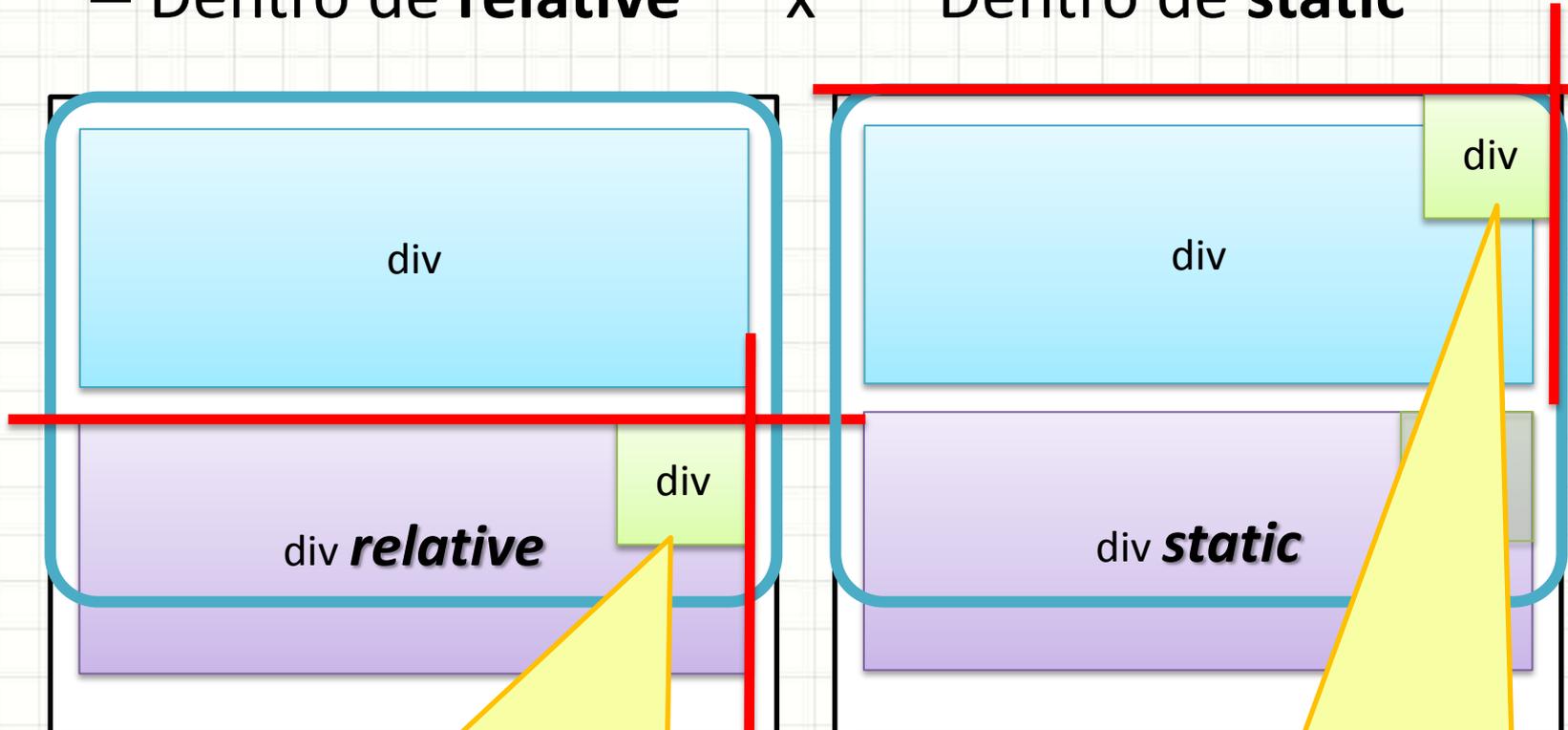
```
</div>
```



**#d3** absolute com  
top=0px e right=0px

# Posicionamento + CSS: **position**

- **position: absolute** – top 0x; right: 0px
  - Dentro de **relative**      x      Dentro de **static**

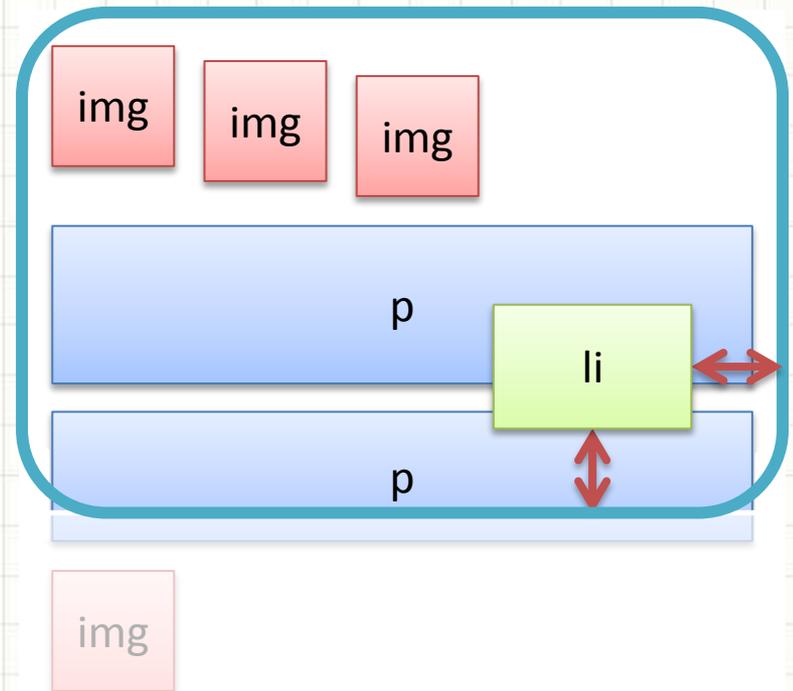


**div absolute** com top=0px e right=0px

**div absolute** com top=0px e right=0px

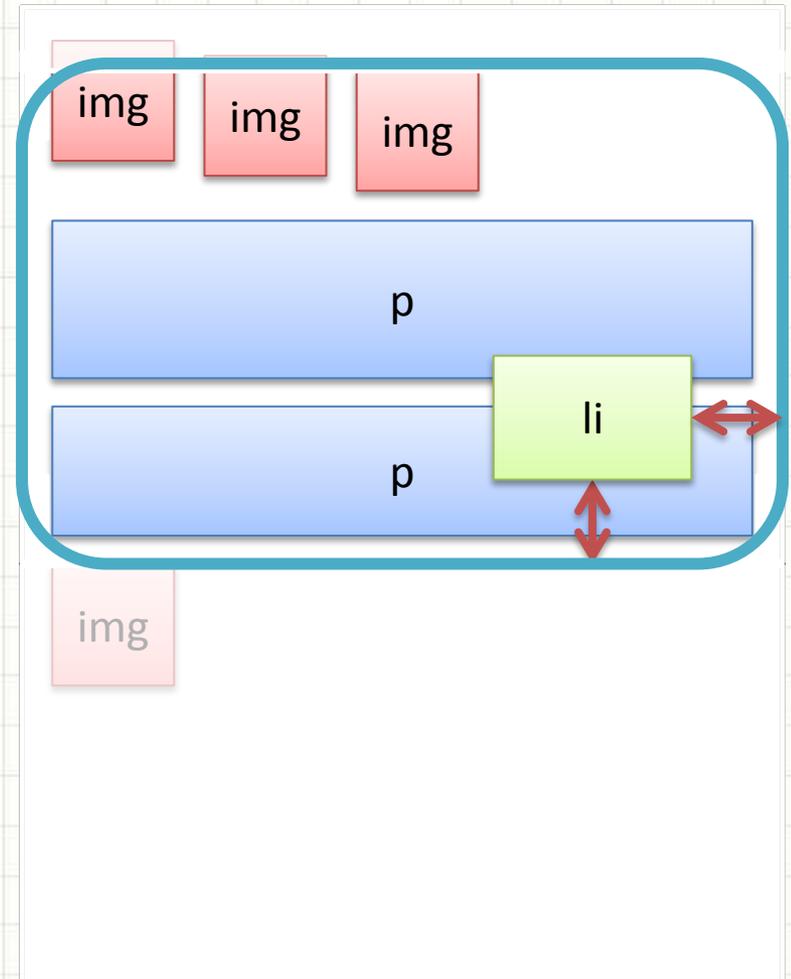
# Posicionamento + CSS: **position**

- Atributo **position: fixed**
- Base: **janela** (navegador)
- Espaçamento
  - margin, left, right, top  
bottom
- Tamanho
  - Conteúdo, padding,  
border, width, height



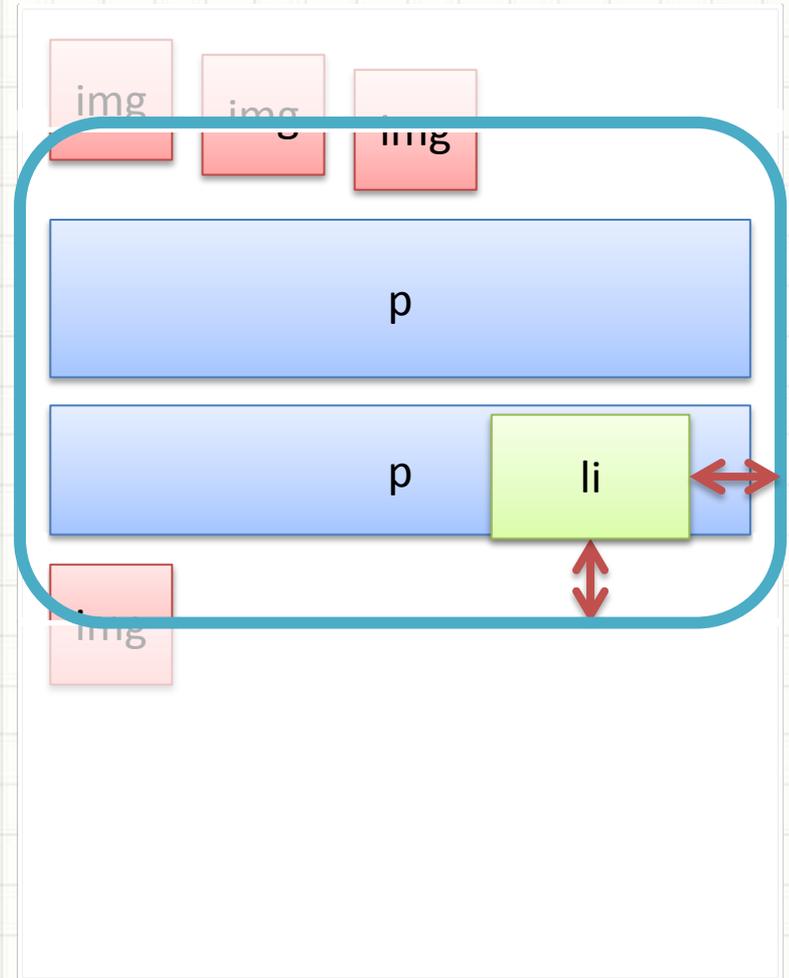
# Posicionamento + CSS: **position**

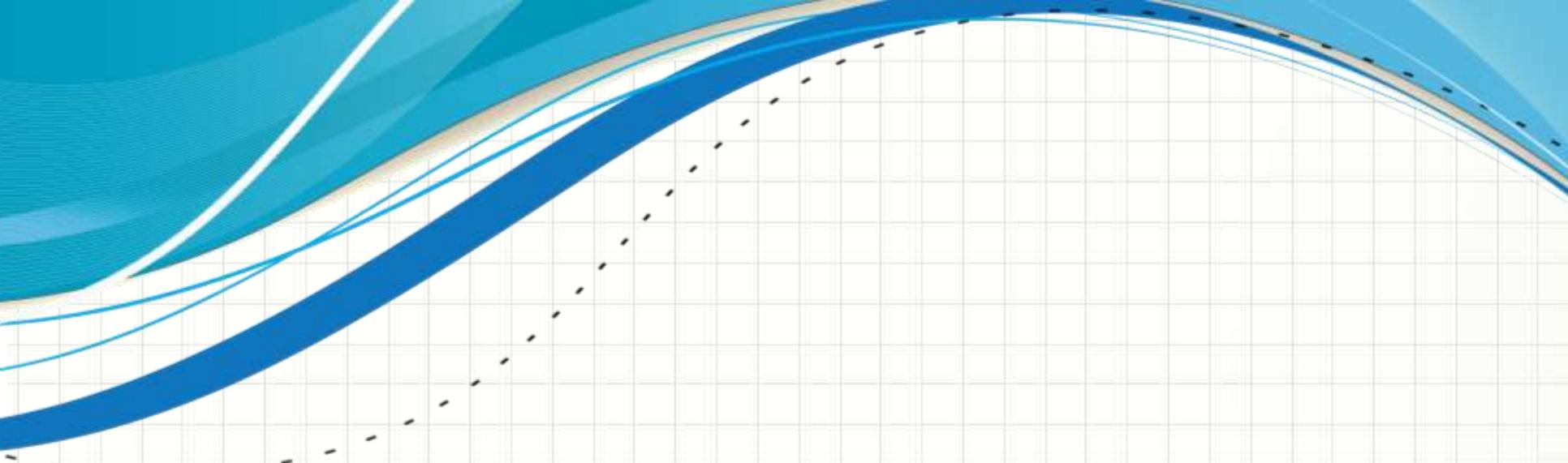
- Atributo **position: fixed**
- Base: **janela** (navegador)
- Espaçamento
  - margin, left, right, top  
bottom
- Tamanho
  - Conteúdo, padding,  
border, width, height



# Posicionamento + CSS: **position**

- Atributo **position: fixed**
- Base: **janela** (navegador)
- Espaçamento
  - margin, left, right, top  
bottom
- Tamanho
  - Conteúdo, padding,  
border, width, height

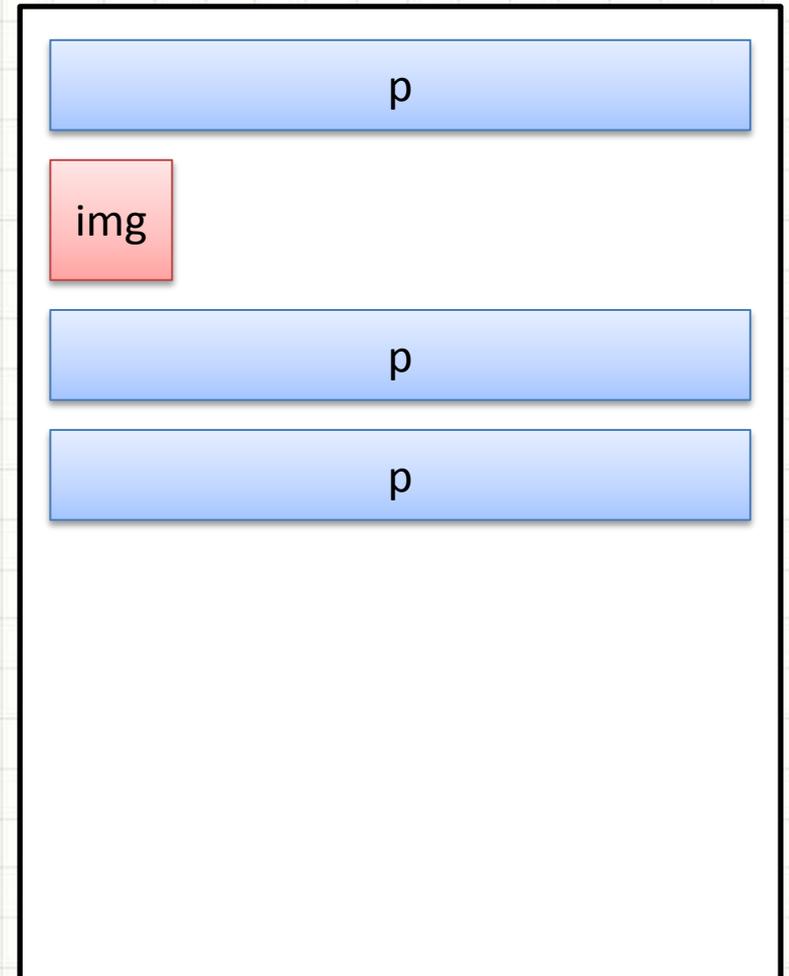




# **POSICIONAMENTO COM O CSS: ELEMENTOS “PRESOS” ÀS LATERAIS**

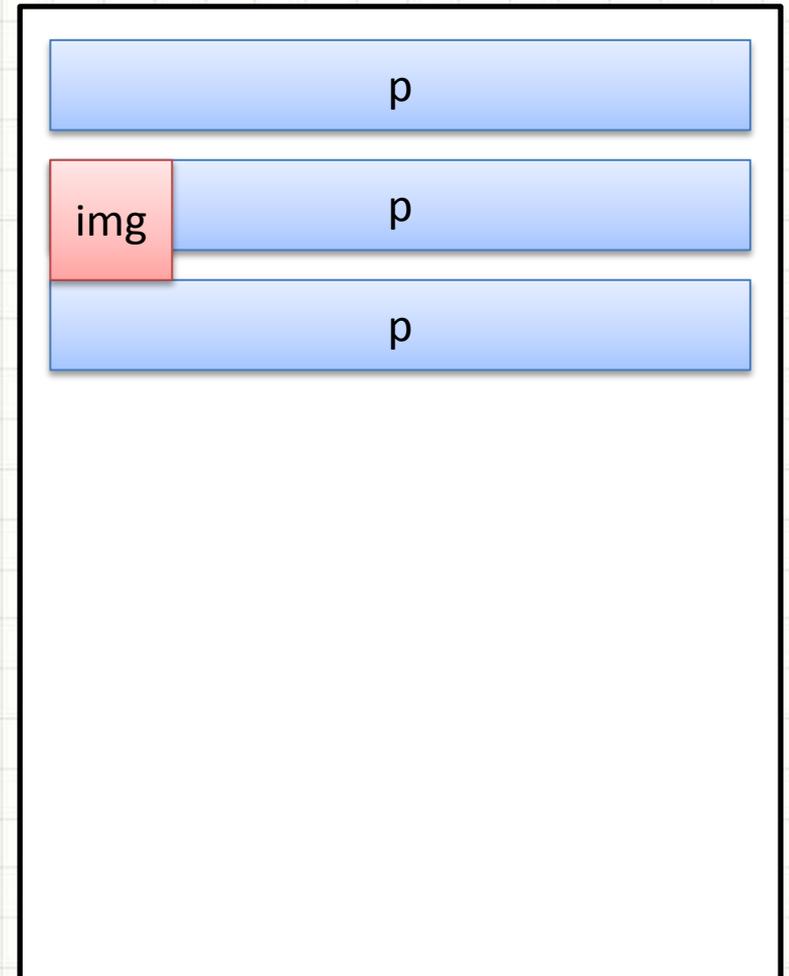
# Posicionamento + CSS: float

- Atributo **float: none**
- Seguida a regra normal do posicionamento



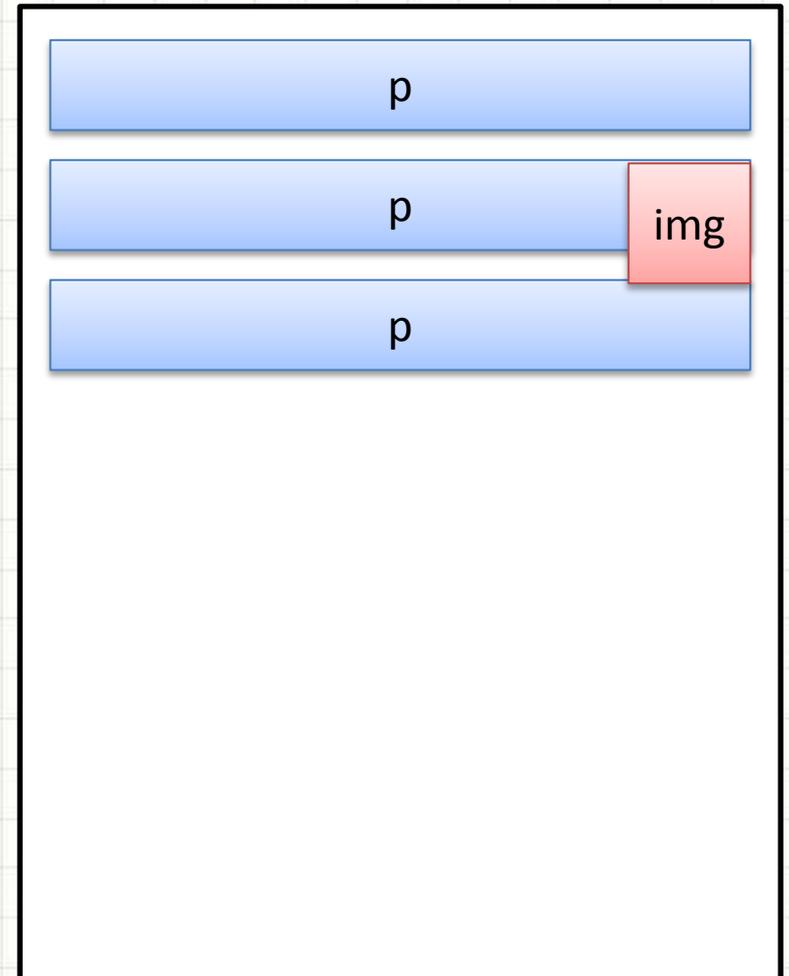
# Posicionamento + CSS: **float**

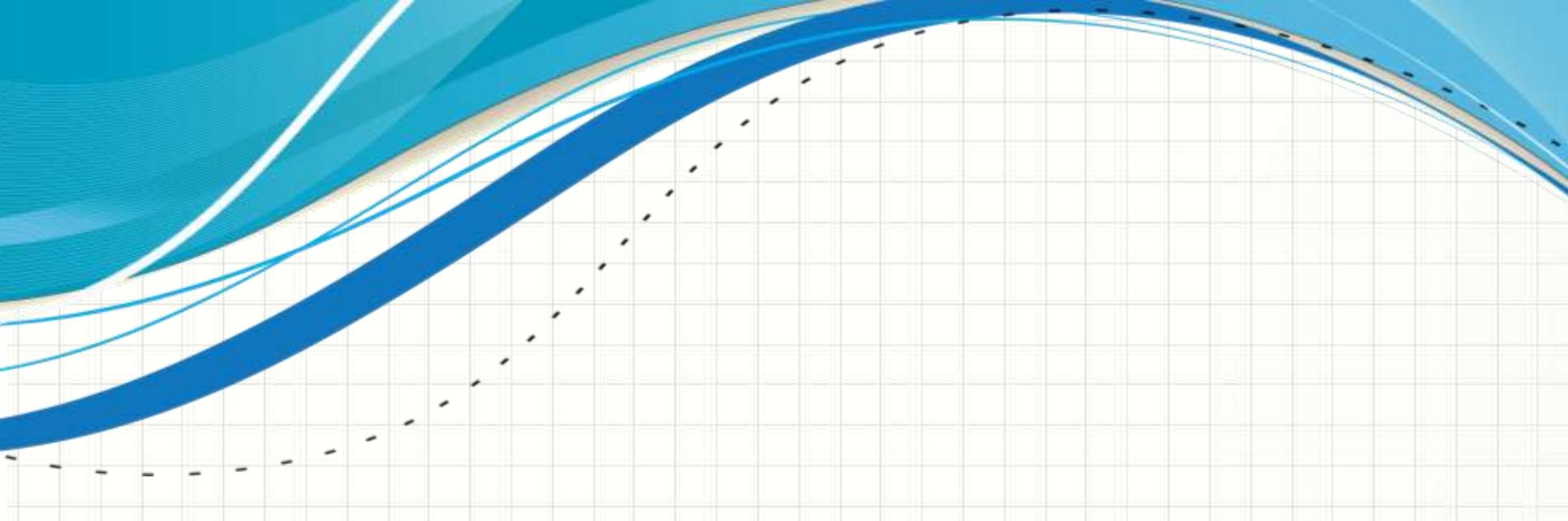
- Atributo **float: left**
- Elemento “gruda” na lateral... E os outros elementos “escorrem” à sua volta



# Posicionamento + CSS: **float**

- Atributo **float: right**
- Elemento “gruda” na lateral... E os outros elementos “escorrem” à sua volta

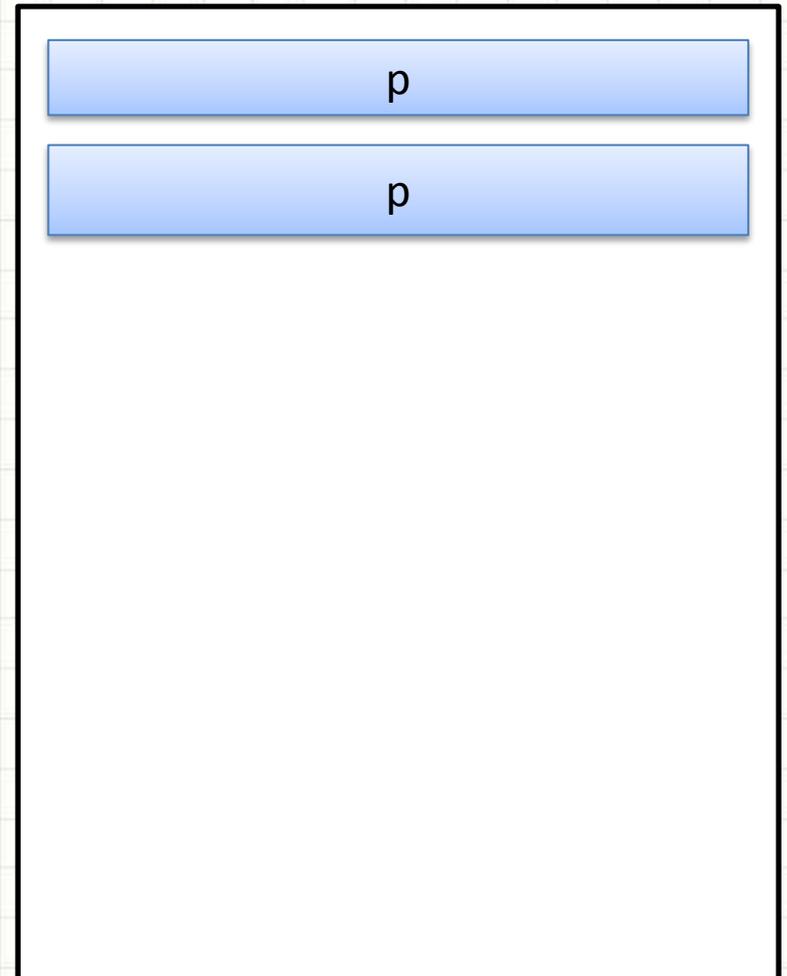




**POSICIONAMENTO COM O  
CSS: VISUALIZANDO  
CONTEÚDO DOS ELEMENTOS**

# Posicionamento + CSS: **overflow**

- Atributo **overflow: visible**
- O elemento desrespeita as medidas indicadas para que o conteúdo seja totalmente visível



# Posicionamento + CSS: **overflow**

- Atributo **overflow: visible**
- O elemento desrespeita as medidas indicadas para que o conteúdo seja totalmente visível

Aqui há um texto com overflow: visible ativado, permitindo todo o texto e desrespeitando a definição original

p

# Posicionamento + CSS: **overflow**

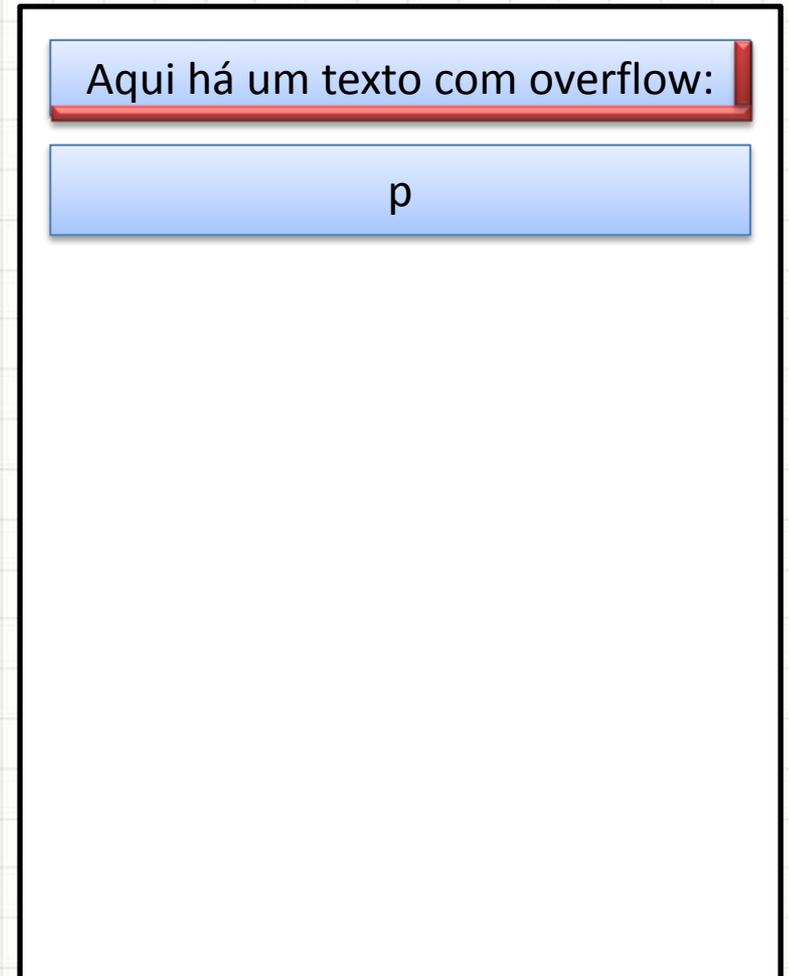
- Atributo **overflow: hidden**
- O elemento respeita as medidas indicadas, ainda que isso signifique cortar o conteúdo

Aqui há um texto com overflow:

p

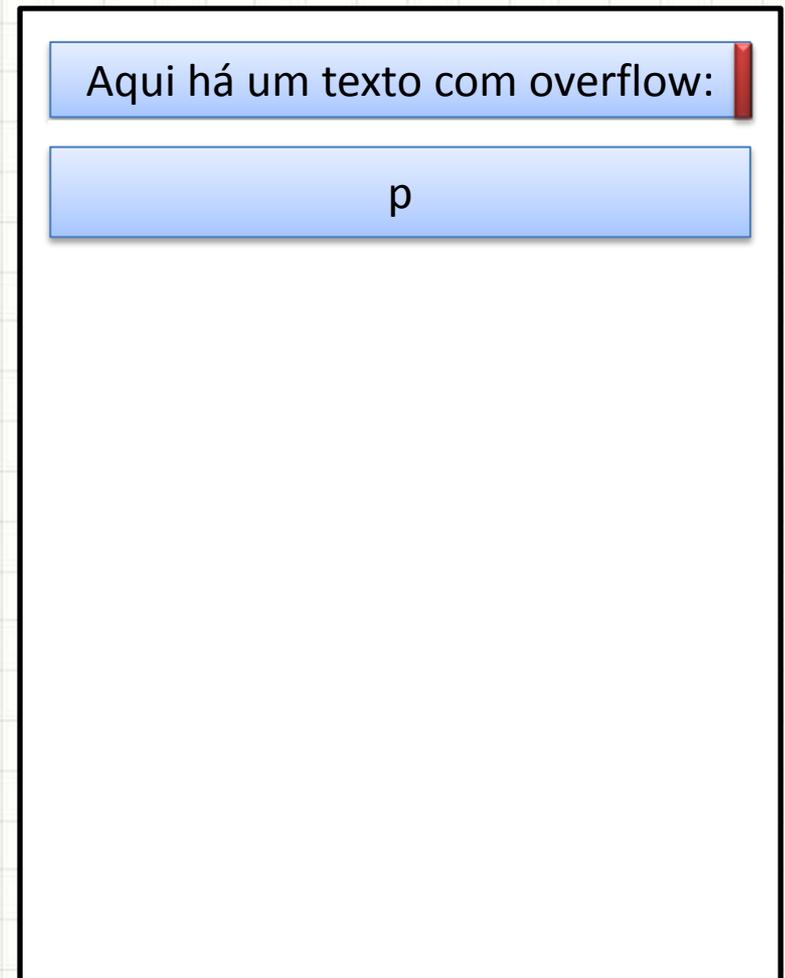
# Posicionamento + CSS: **overflow**

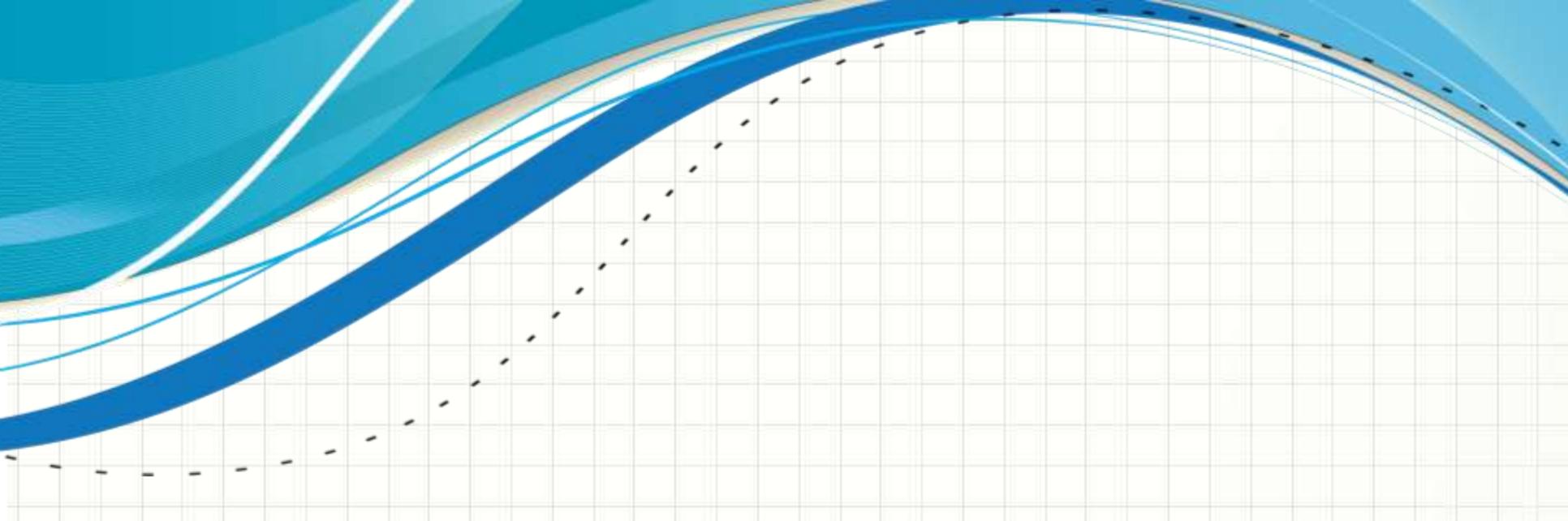
- Atributo **overflow: scrollbar**
- O elemento respeita as medidas indicadas, mas sempre serão mostradas barras de rolagem



# Posicionamento + CSS: **overflow**

- Atributo **overflow: auto**
- O elemento respeita as medidas indicadas, mas se o conteúdo não couber, mostra a barra de rolagem necessária

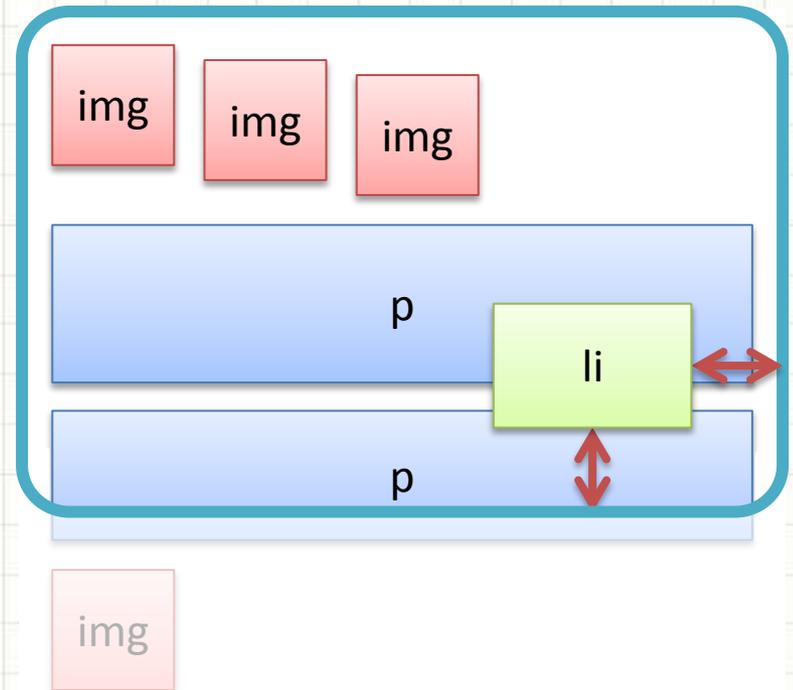




**POSICIONAMENTO COM O  
CSS: CONTROLANDO QUEM  
ESTÁ NA FRENTE**

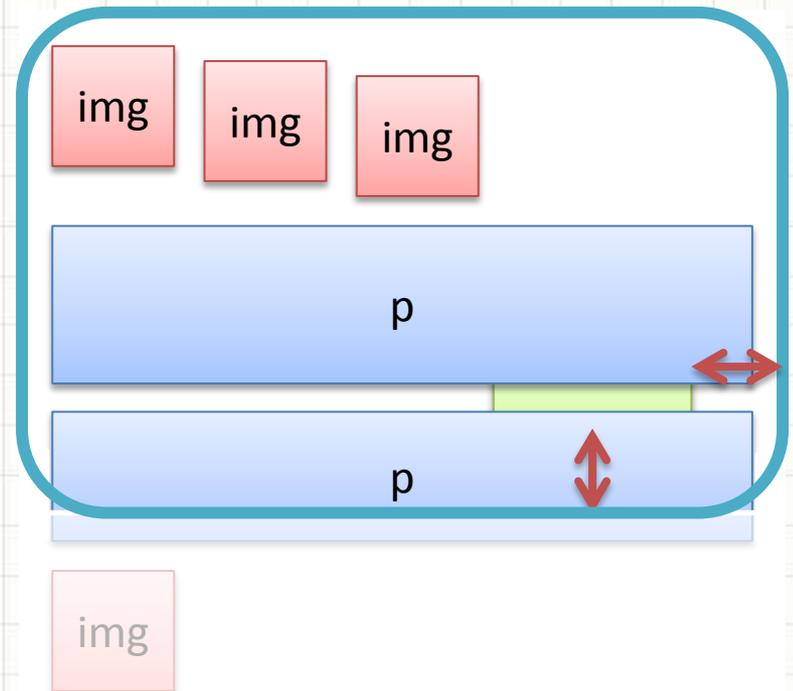
# Posicionamento + CSS: z-index

- Atributo **z-index**: [valor]
- Controla o nível do elemento
- Padrão: o que vem depois no XHTML, fica por cima



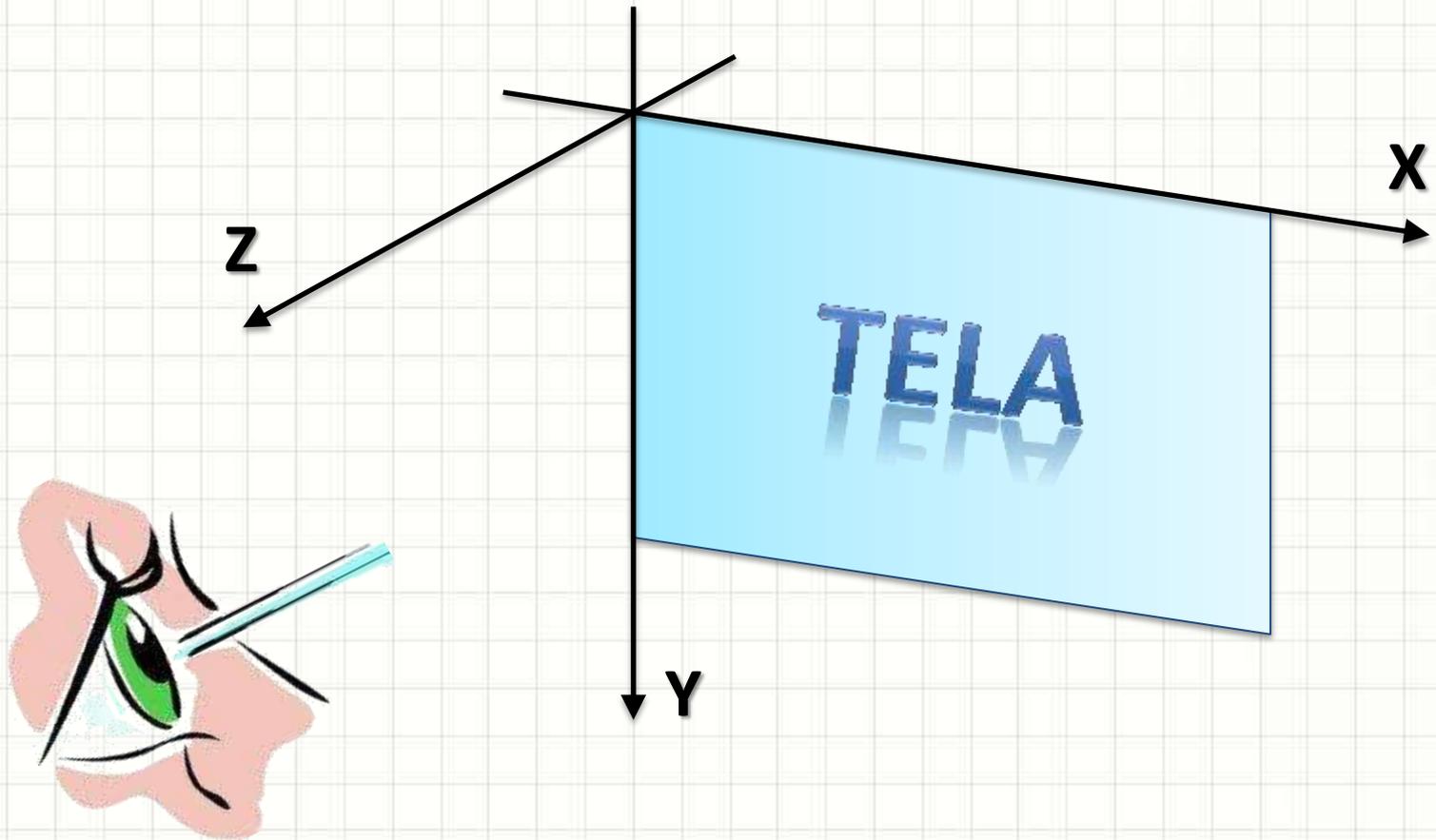
# Posicionamento + CSS: z-index

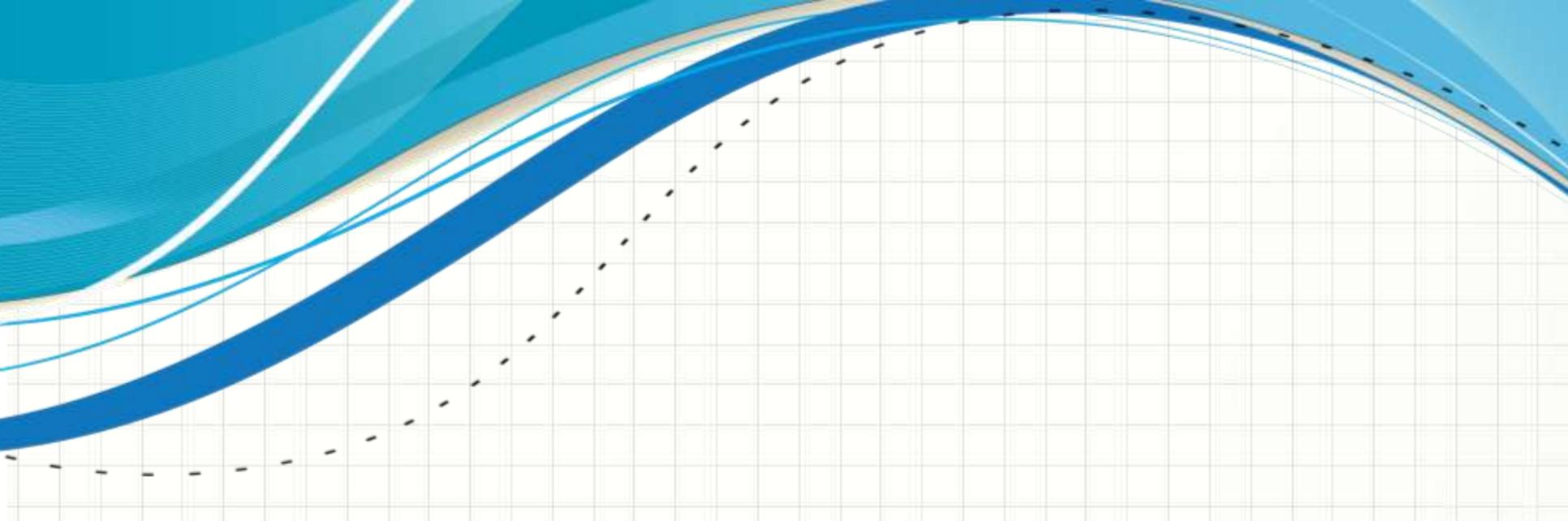
- Atributo **z-index**: [valor]
- Controla o nível do elemento
- Padrão: o que vem depois no XHTML, fica por cima
- Mudando o z-index...



# Posicionamento + CSS: z-index

- Atributo z-index: [valor]

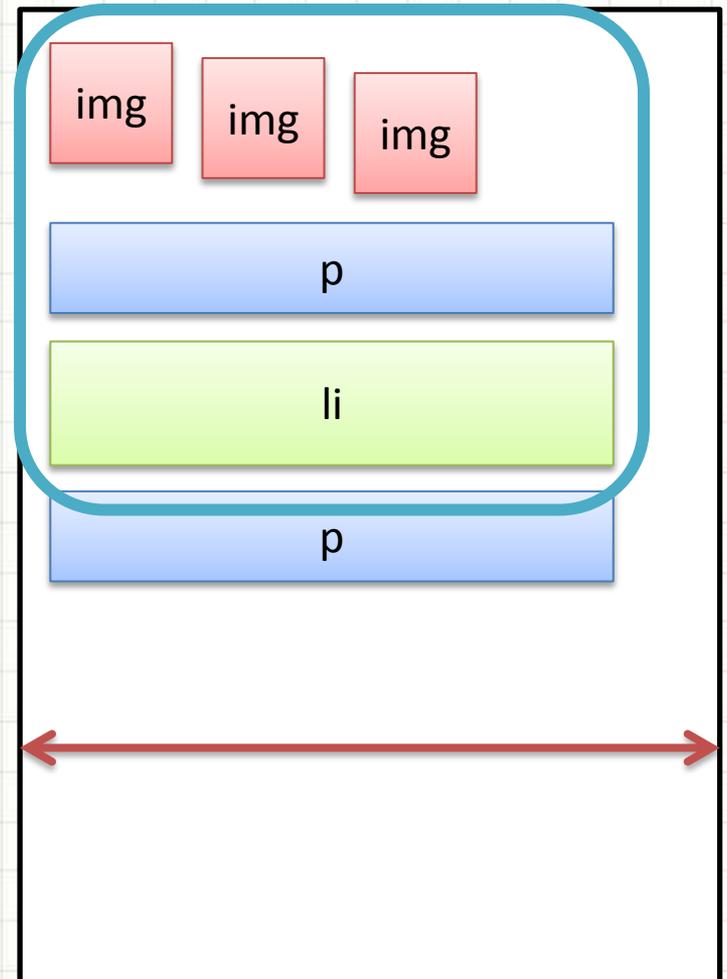




**POSICIONAMENTO COM O  
CSS: LIMITANDO O  
TAMANHO DO CONTEÚDO**

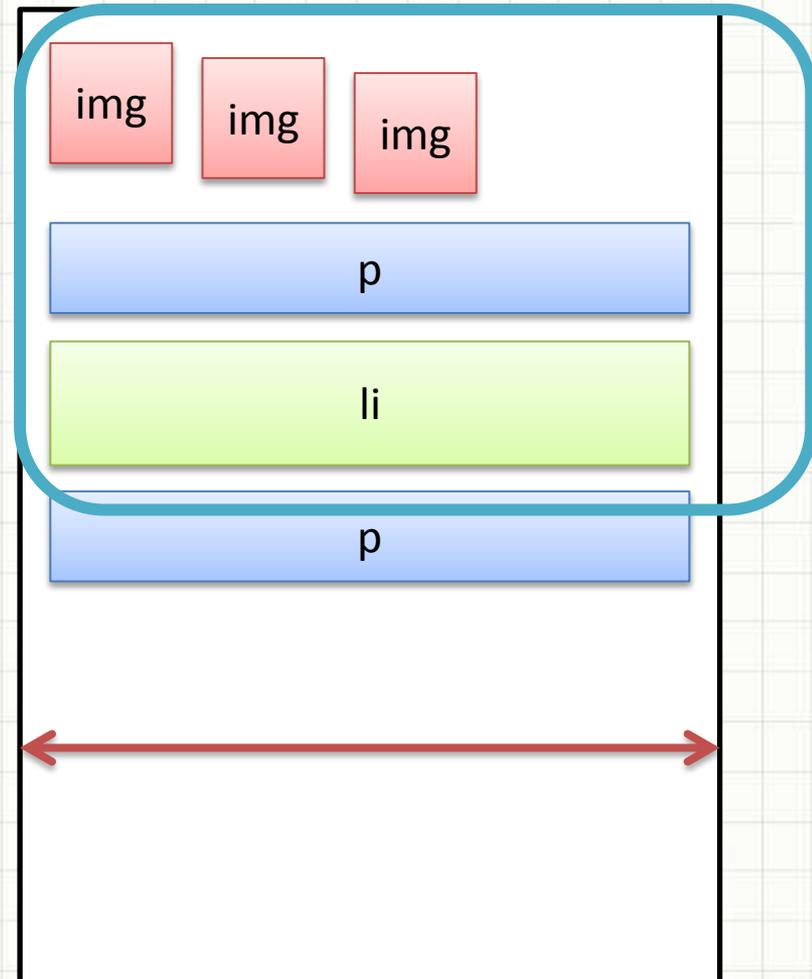
# Posicionamento + CSS: **max-width**

- Atributo **max-width: 700px**
- Se a tela for menor
  - Desenha menor



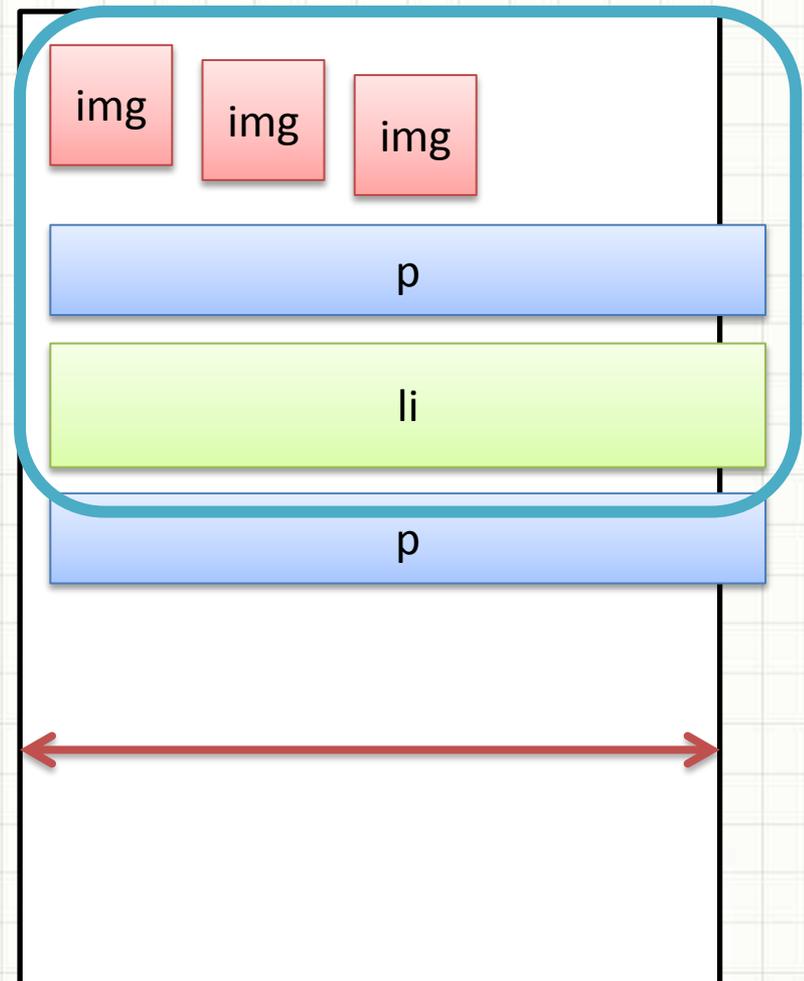
# Posicionamento + CSS: **max-width**

- Atributo **max-width: 700px**
- Se a tela for menor
  - Desenha menor
- Se a tela for maior
  - Limita o tamanho!



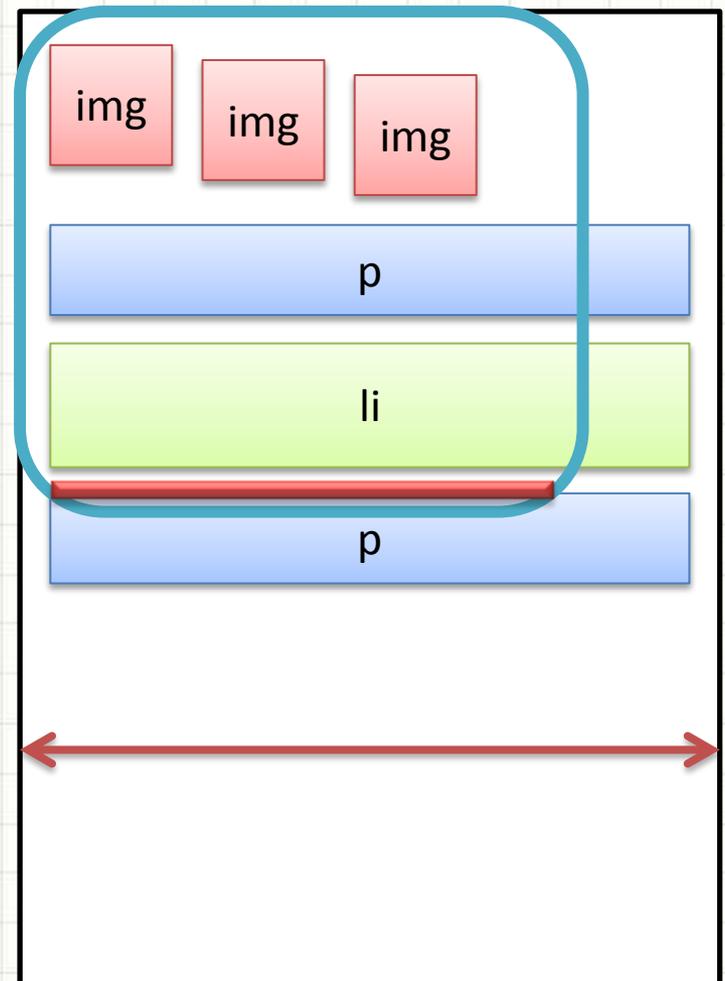
# Posicionamento + CSS: **min-width**

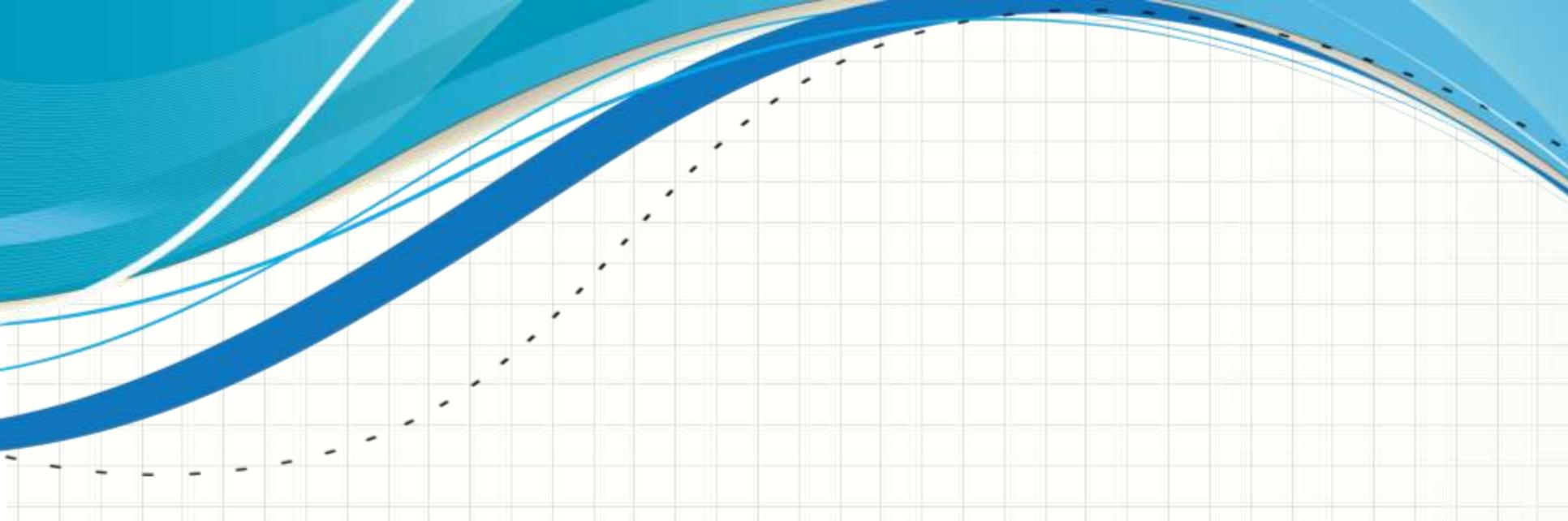
- Atributo **min-width: 700px**
- Se a tela for maior
  - Desenha maior



# Posicionamento + CSS: min-width

- Atributo **min-width: 700px**
- Se a tela for maior
  - Desenha maior
- Se a tela for menor
  - Põe barra de rolagem!

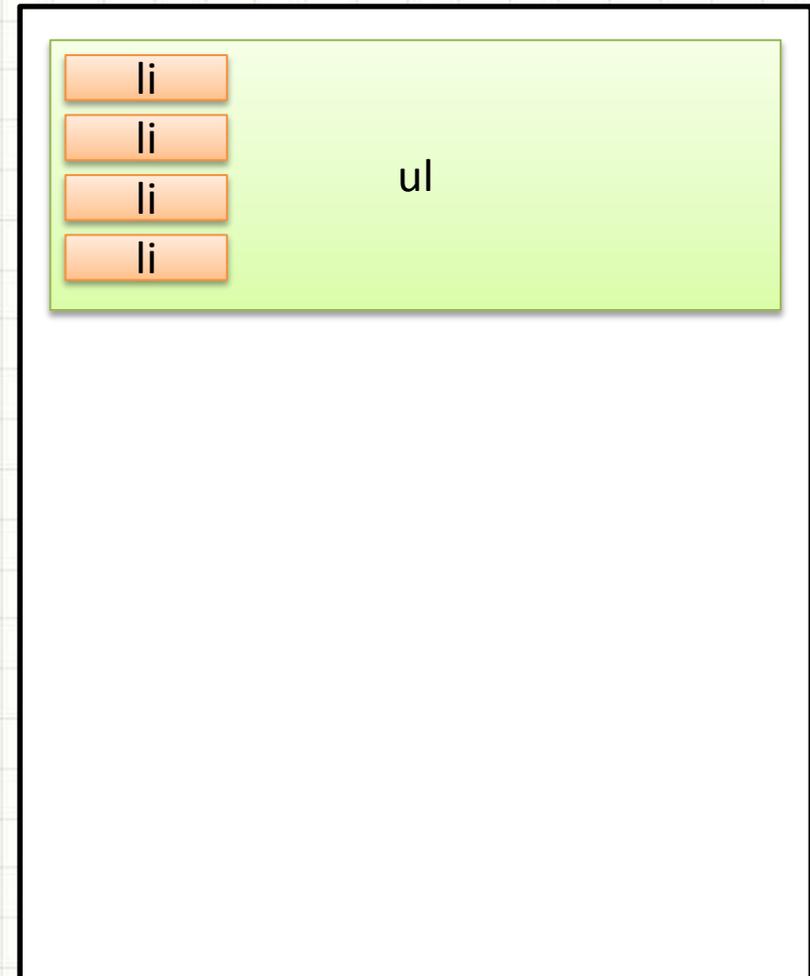




# **POSICIONAMENTO COM O CSS: MODIFICANDO A FORMA DE APRESENTAÇÃO**

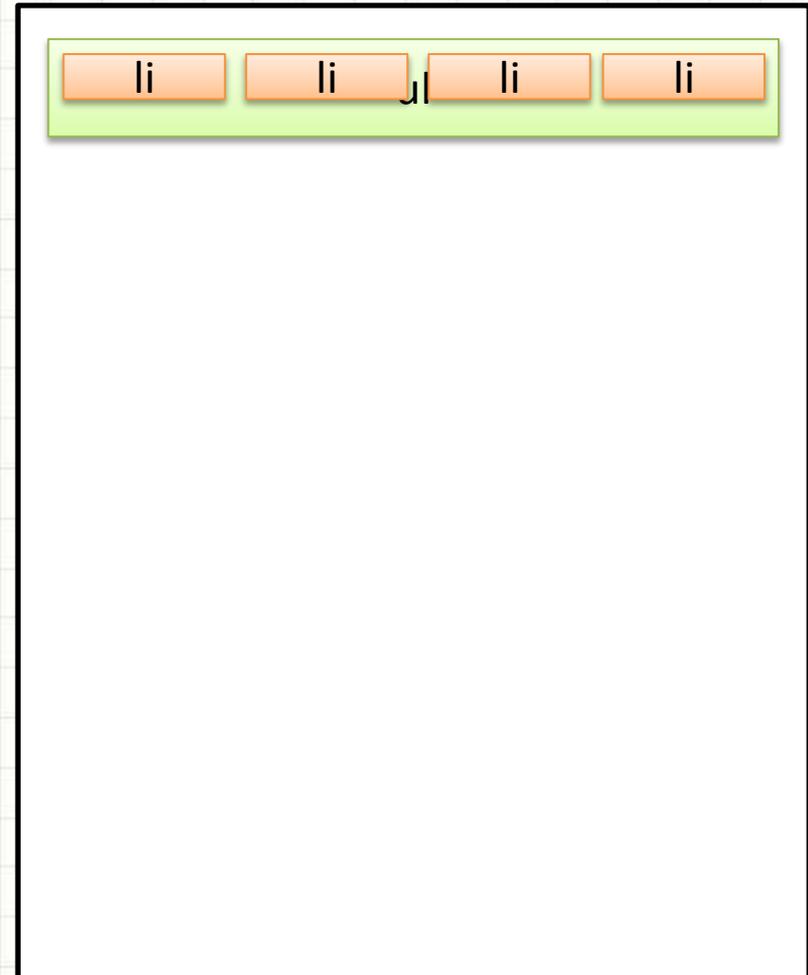
# Posicionamento + CSS: **display**

- Sem atributo **display: block**
- Apresenta elemento como previsto
- Listas, por exemplo



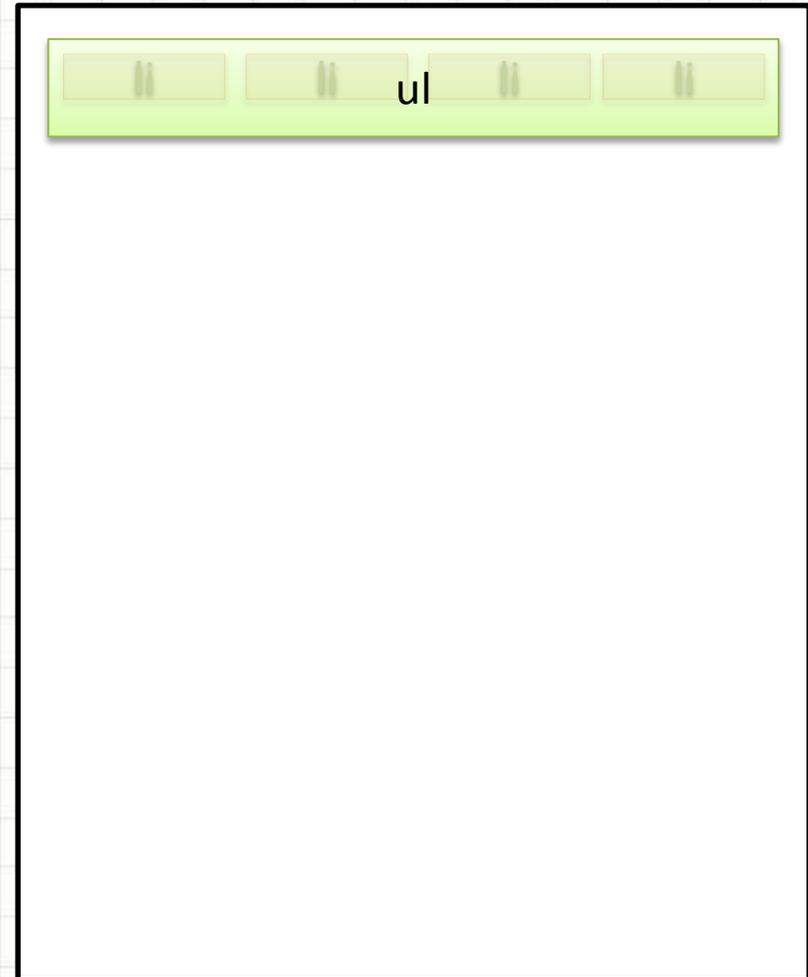
# Posicionamento + CSS: **display**

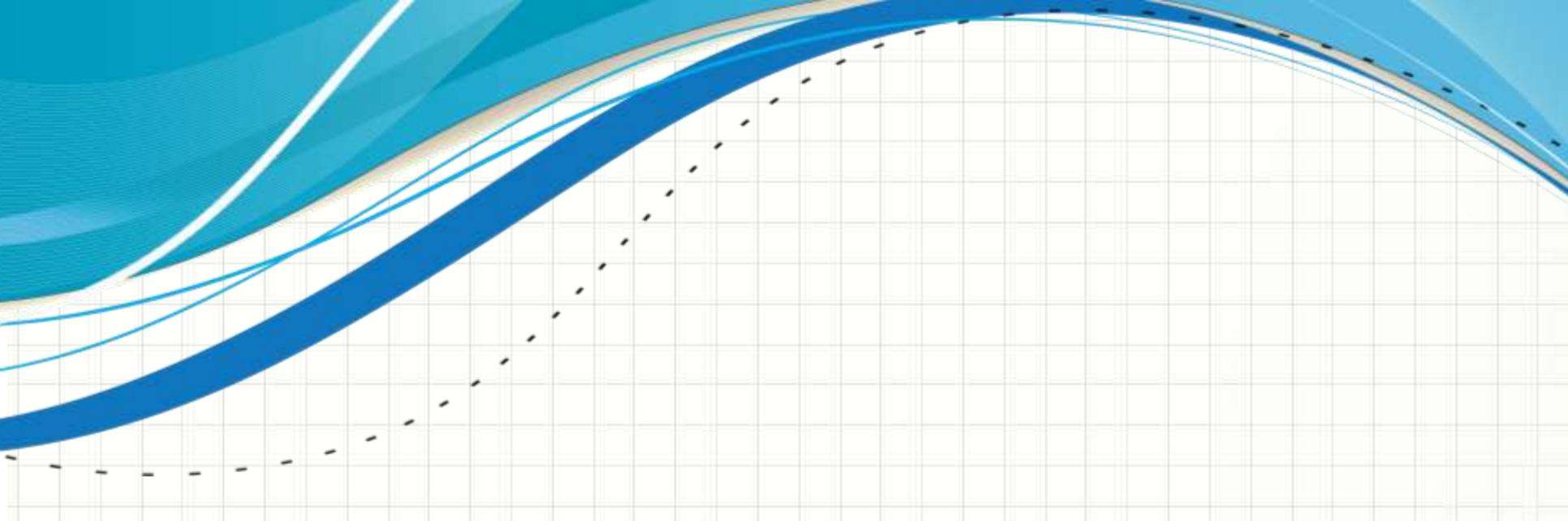
- Sem atributo **display: inline**
- Apresenta elementos em linha
- Por exemplo, listas



# Posicionamento + CSS: **display**

- Sem atributo **display: none**
- Some com elementos





# **POSICIONAMENTO NA PRÁTICA: TUTORIAL**

# Entendendo o posicionamento...

- Acompanhe o professor com o código “**pos**” para entender o mecanismo...

# Layout Base: Tutorial Básico

MENU

TÍTULO

SEÇÃO 1

# Layout Base: Tutorial Básico

MENU

TÍTULO

SEÇÃO 1

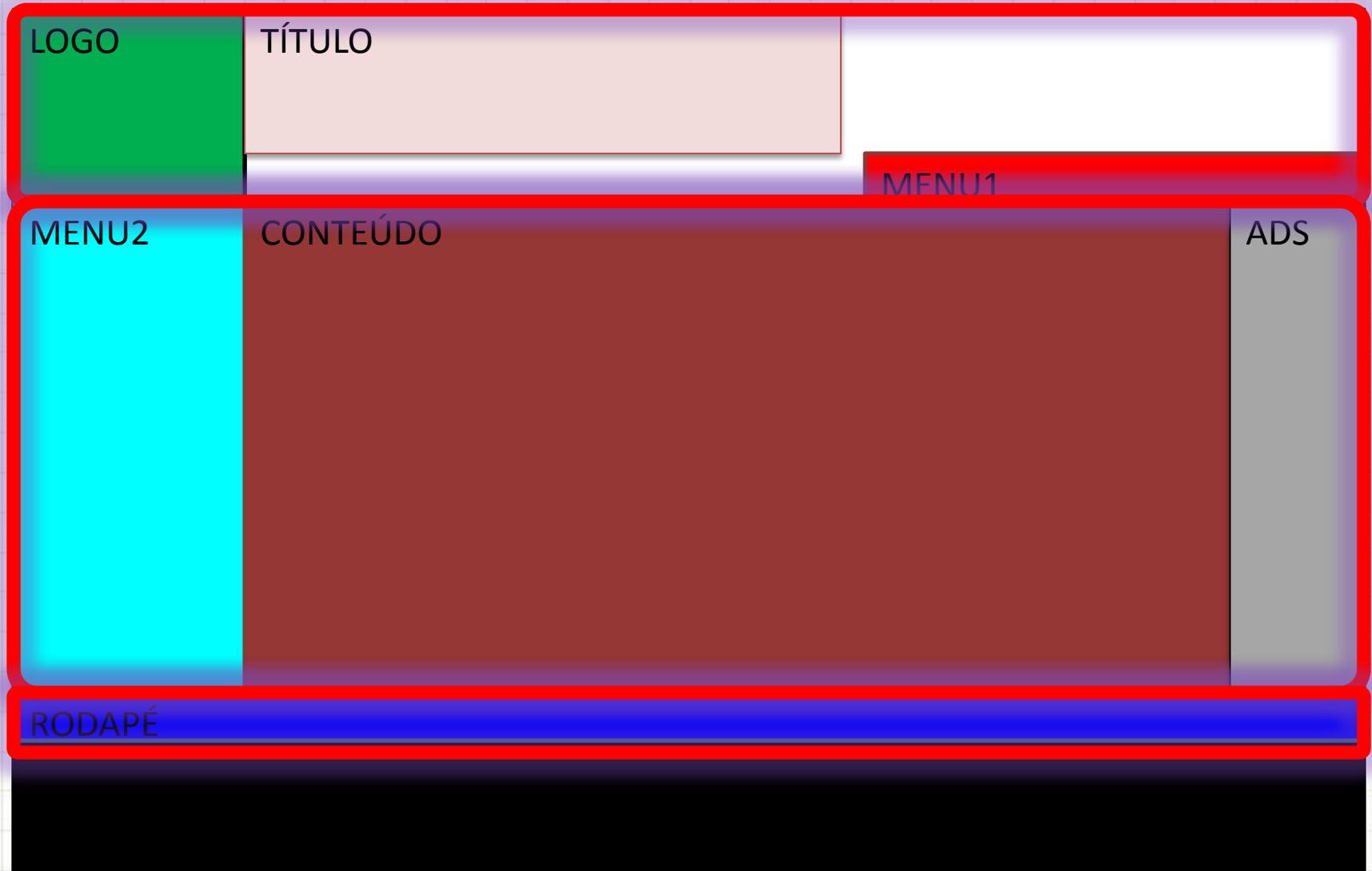
# Layout Base: Tutorial Básico

- Construa com o professor para compreender o processo...

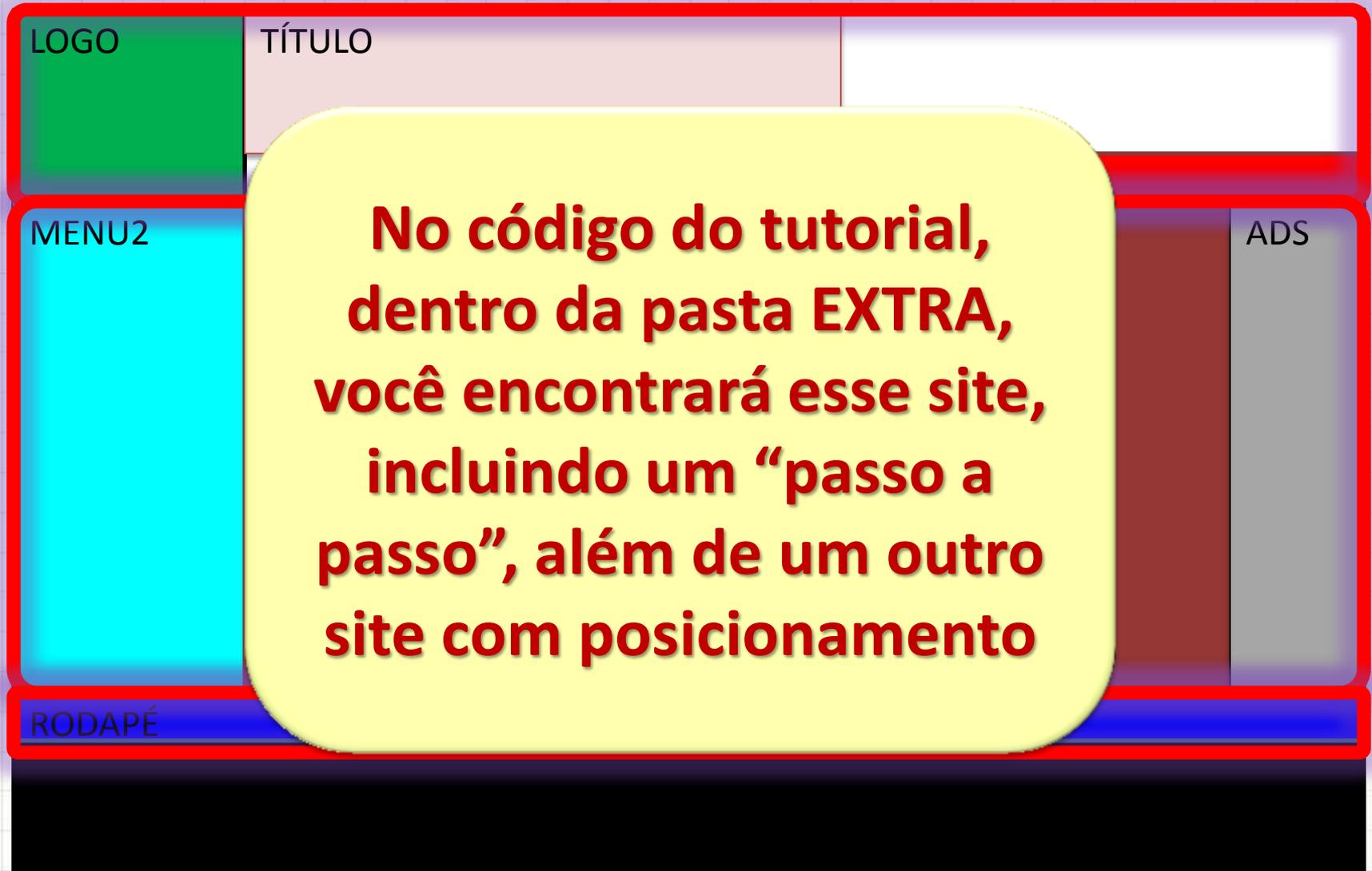
# Layout Base: Tutorial Avançado

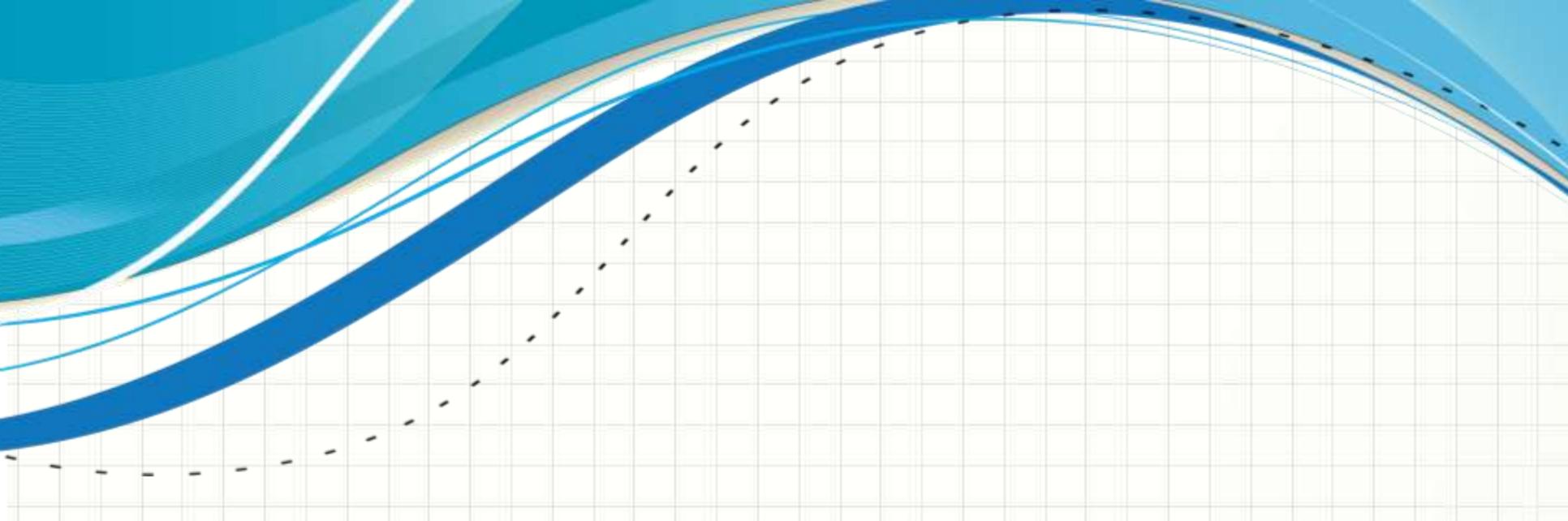


# Layout Base: Tutorial Avançado



# Layout Base: Tutorial Avançado

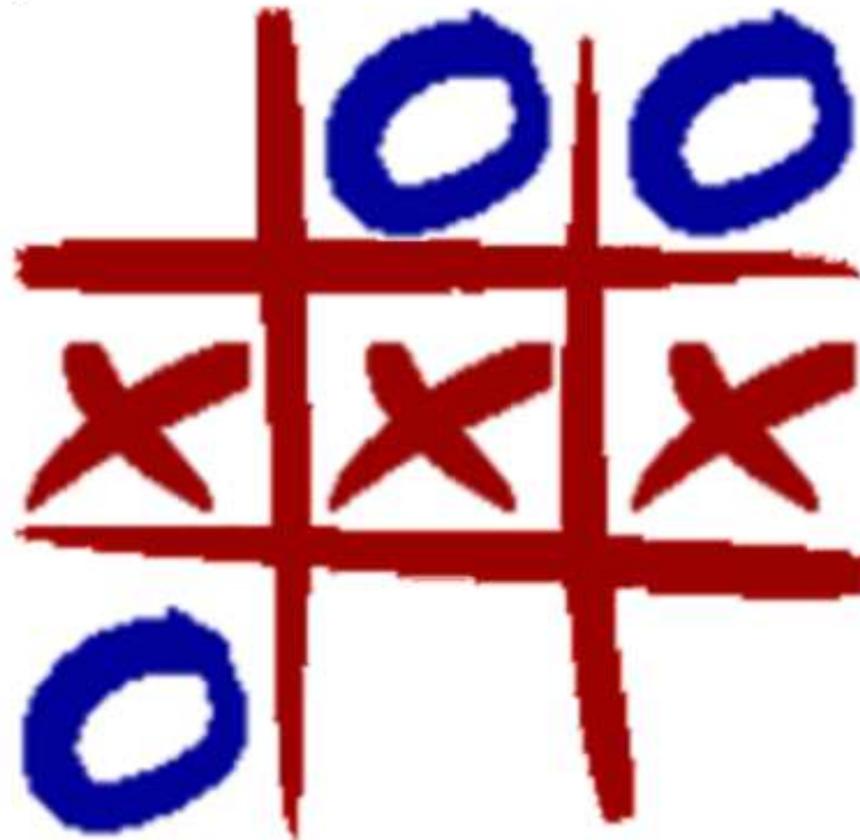




# **POSICIONAMENTO EM APLICAÇÕES: TUTORIAL**

# Jogo da Velha

X  O



**Agora V: 2 / P: 0 Total V: 2 / P: 0**

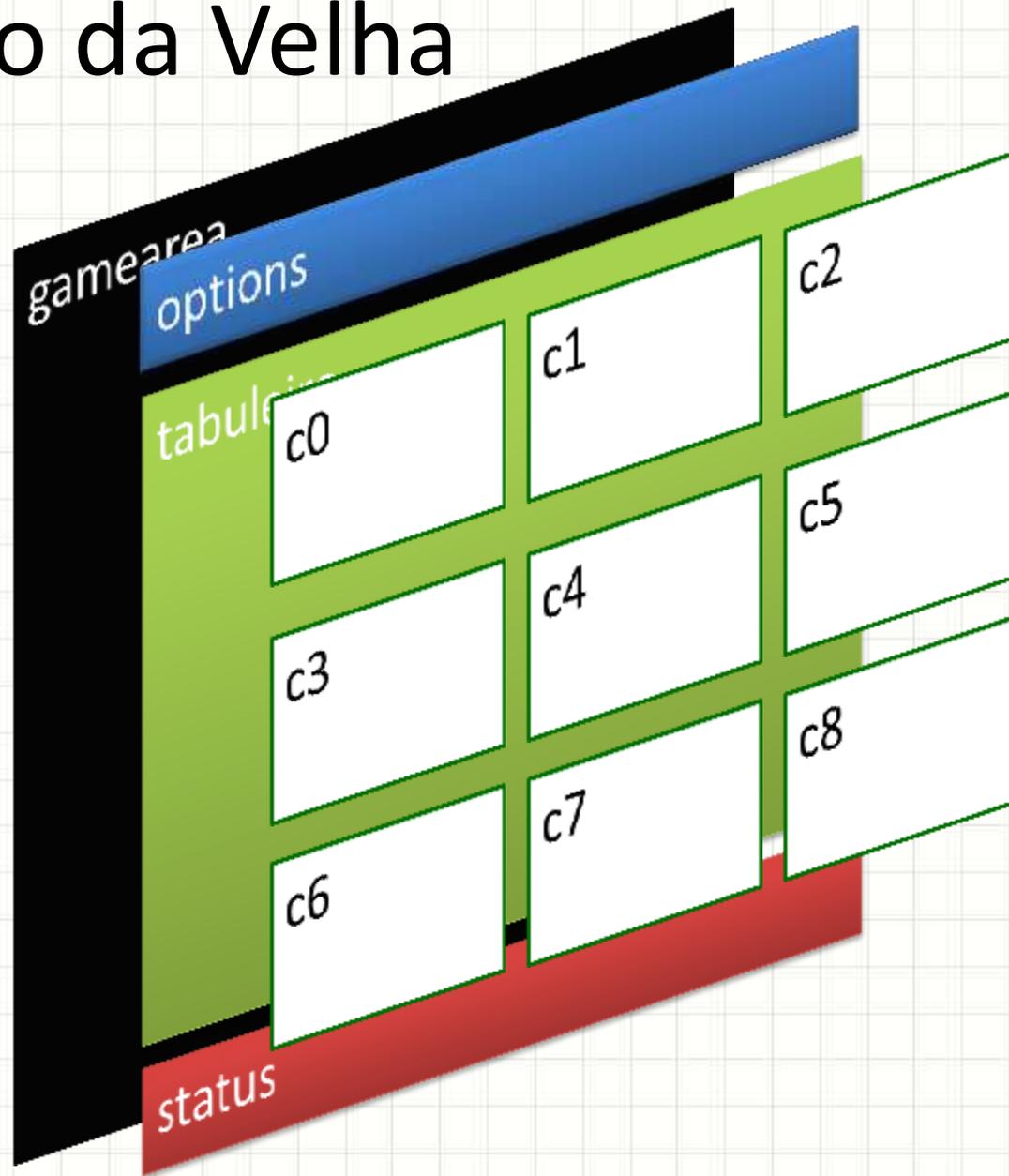
# Jogo da Velha

gamearea

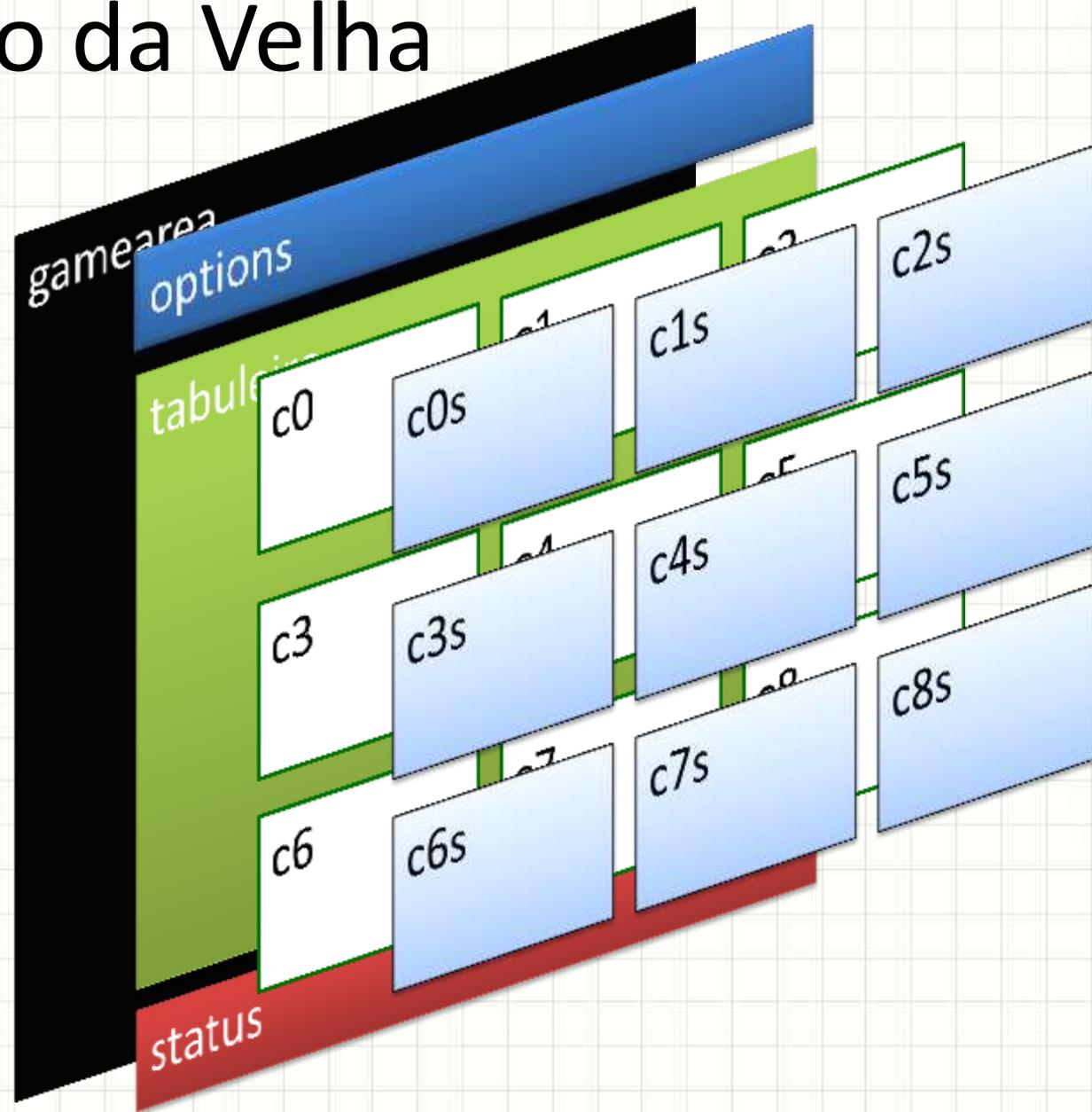
# Jogo da Velha

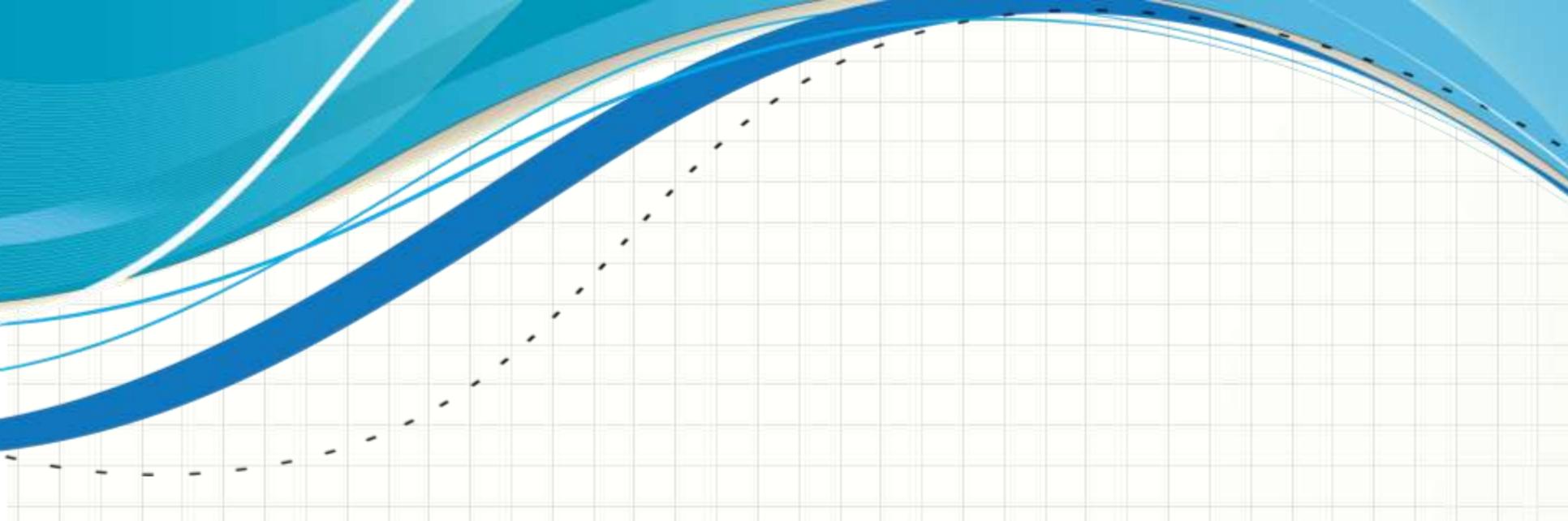


# Jogo da Velha



# Jogo da Velha





**CONCLUSÕES**

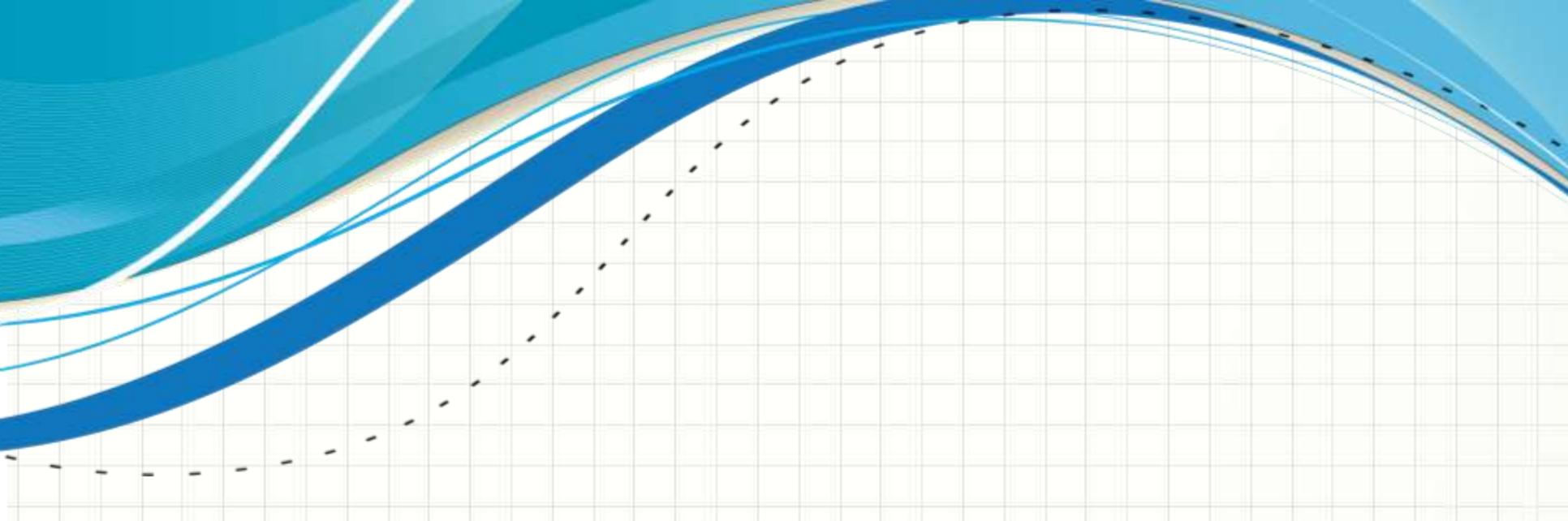
# Resumo

- CSS permite total liberdade de formatação de XHTML
- O uso depende da correta compreensão da lógica do posicionamento CSS...
- ...e de uma adequada análise prévia do layout desejado
- **TAREFA**
  - Trabalho A...! Já dá pra fazer!

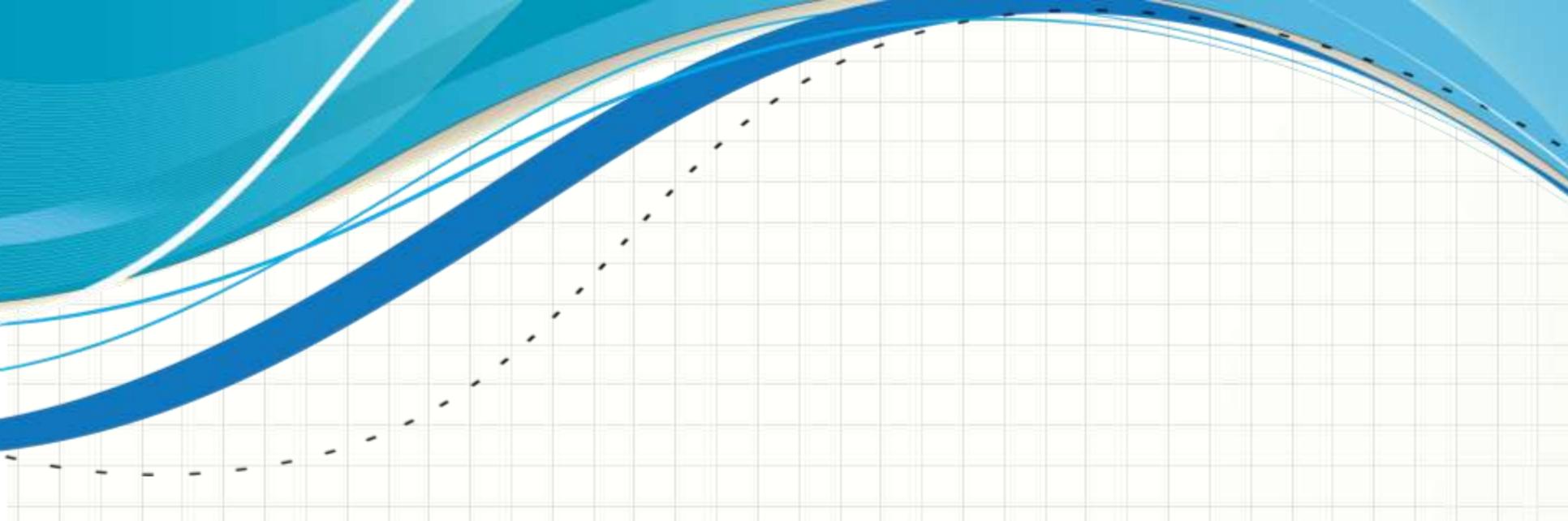
# Próxima Aula



- JavaScript...
  - Para fazer sites realmente dinâmicos e responsivos...
  - ...temos que alterar suas propriedades na máquina do cliente!



**PERGUNTAS?**



**BOM DESCANSO  
A TODOS!**