

Containers

`<div>`, `<article>`, `<section>`,
`<header>`, `<footer>`, ...



Professores:

Carla Plantier Message – cpmessage@unoeste.edu.br

Tamiris Malacrida – tamiris@unoeste.edu.br

Cristiane Maciel Rizo - cris@unoeste.br

Cássia Alves Perego - cassia@unoeste.br

Silvio Carro - silvio@unoeste.br



Home | Company | Contact Us | Search:

Log In | Why Become an Argent Approved broker? See the benefits.

Company Branded Messaging

Main Navigation

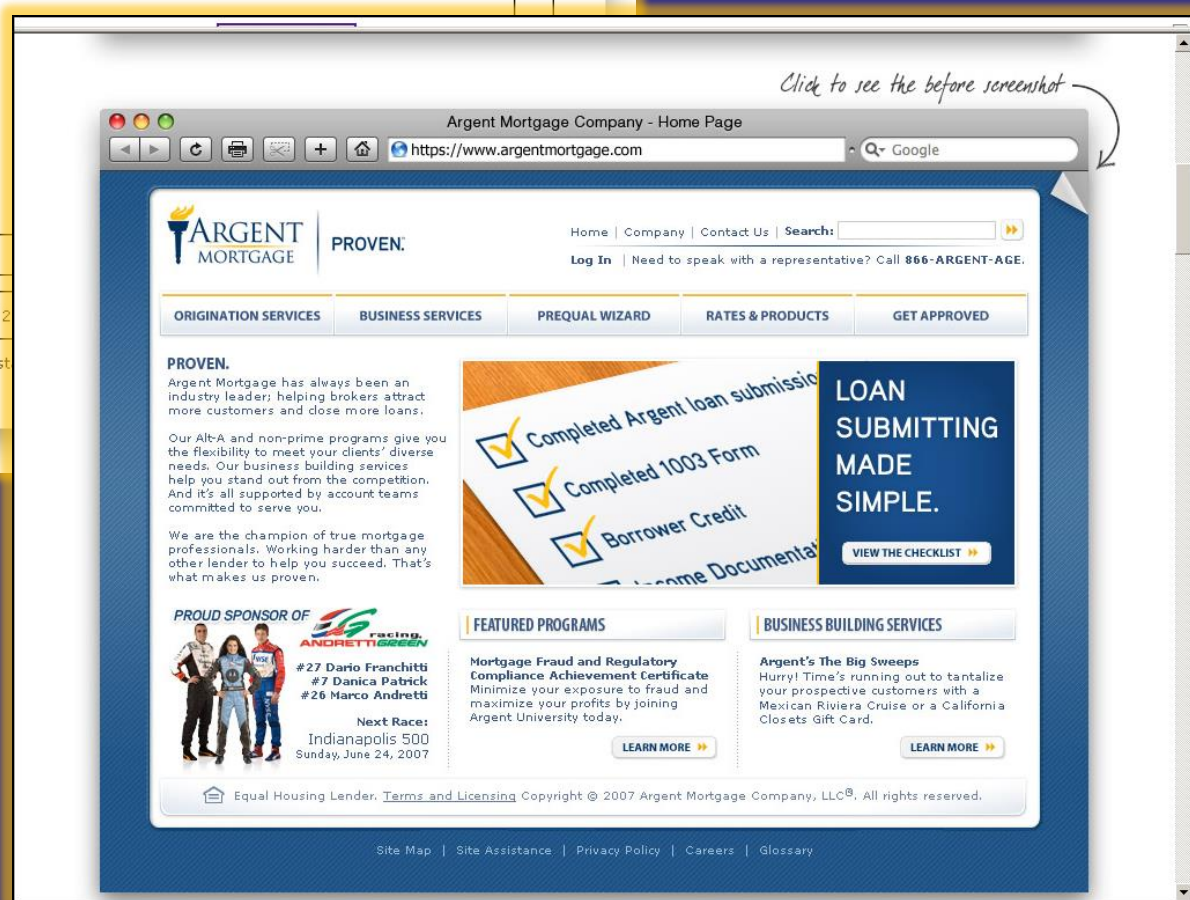
Get Approved

SEO Copy

Equal Housing Lender. Terms and Licensing Copyright © 2007

Site Map | Site Assistance

Divs



Definindo tamanho e posicionamento dos blocos

- Em CSS podemos posicionar elementos de bloco por meio do atributo position.
- Possíveis valores:
 - absolute
 - relative
 - fixed
 - static

Definindo tamanho e posicionamento dos blocos

- **static:** é o valor padrão, e indica que o posicionamento do elemento não foi modificado e seguirá o fluxo normal. **Ignora o posicionamento.**
- **absolute:** determina que o elemento deve ficar na posição que você mandar. O local especificado para o elemento é em relação ao seu elemento pai que esteja em posição relativa. Se não houver um elemento pai com posição relativa, o elemento ficará posicionado em relação à janela.

Definindo tamanho e posicionamento dos blocos

- **relativo:** determina que o elemento deve ficar posicionado de acordo com sua posição original no fluxo normal.
 - Ao usar as propriedades **top**, **left**, **right** e **bottom**, o deslocamento acontece a partir de sua posição no fluxo.
- **fixed:** determina que o elemento deve ficar fixo (em relação à janela) na posição que você mandar, e quando a barra de rolagem é movida, este elemento fica fixado na posição que foi definida.

Definindo tamanho e posicionamento dos blocos

- Para o posicionamento:
 - Os atributos **top**, **left**, **bottom** e **right** permitem indicar o posicionamento do elemento.
 - Os atributos **width** e **height** definem a dimensão do elemento.
 - O atributo **z-index** define a ordem de empilhamento (visualização) dos elementos da página, sendo sempre do menor para o maior.

Flutuando blocos

- A CSS disponibiliza uma outra propriedade capaz de mudar o posicionamento padrão de elementos a nível de bloco: **float**.
- Por exemplo, pode ser usada para posicionar uma imagem ao lado de um bloco de texto, como também construir layouts.
- Possíveis valores: **left** e **right**.
- Os valores **left** e **right** posicionam os elementos à esquerda ou à direita de seu container sem quebrar a linha, e os outros elementos vão contorná-lo.

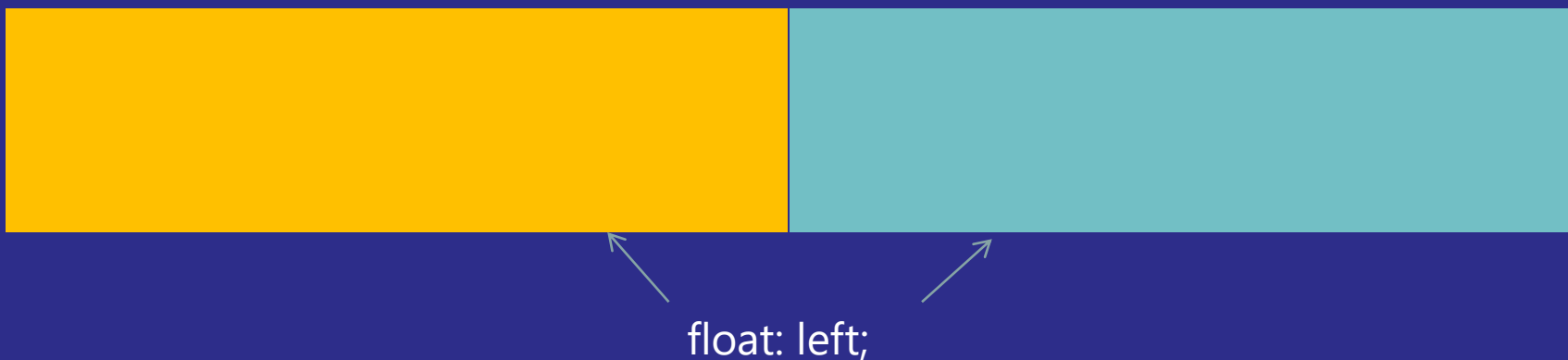
Flutuando blocos

- Exemplo: Dois elementos de bloco com comportamento padrão (um abaixo do outro conforme o padrão de nível de blocos).



Flutuando blocos

- A propriedade **float** permite que um elemento de nível de bloco não quebre a linha no final. O comportamento é de flutuação. O bloco pode ser colocado a esquerda e o restante vai lhe contornar.

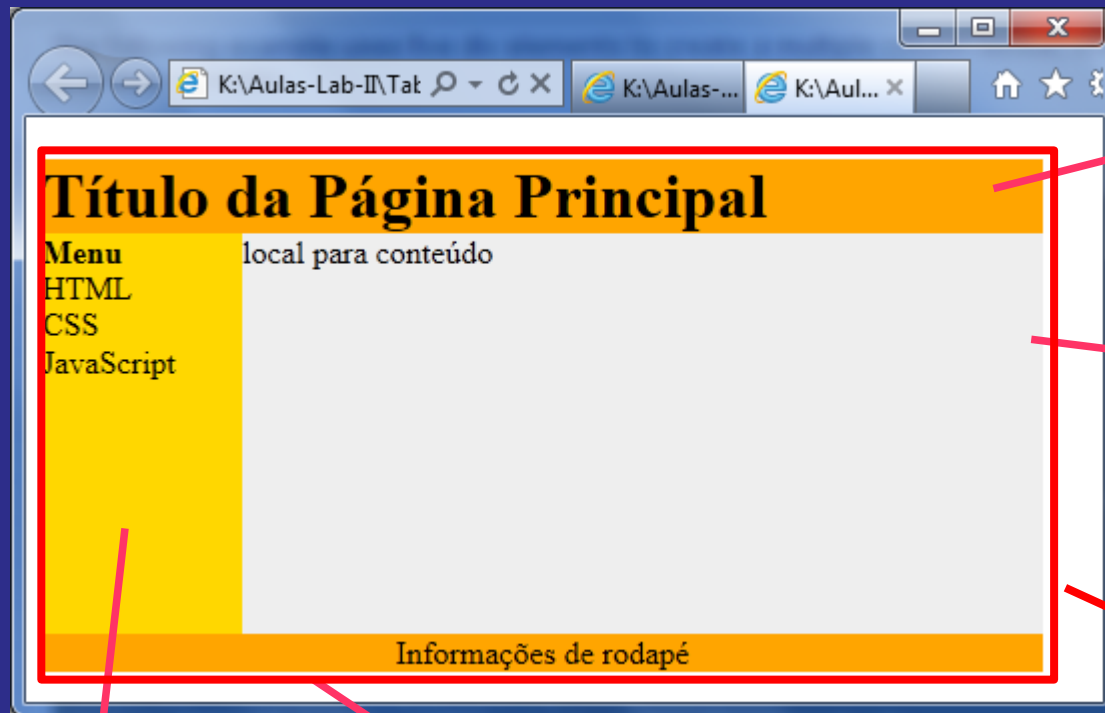


- Podemos “quebrar o fluxo” com a propriedade **clear:both**

Propriedades úteis em DIVs

Propriedade	Significado	Exemplo
width	Largura da div	width:1024px
height	Altura da div	height:500px
float	Posicionar a div na esquerda ou direita	Valores: left, right,
clear	Utilizado em conjunto com float, força que as próximas divs sejam empurradas para baixo	clear:both;
background-color	Cor de fundo da div	background-color:#FFCCAA
text-align	Alinha o texto dentro da div	text-align:center;
overflow	Configura o comportamento da div no caso de ocorrer estouro de conteúdo http://www.w3schools.com/cssref/playit.asp?filename=playcss_overflow	Valores: visible, hidden, auto e scroll;
z-index	Define a ordem de sobreposição das divs	z-index:-1 (vai para fundo de tela)
position	Define regras para serem aplicadas no sistema de coordenadas	Valores: absolute, relative e fixed
left, top, bottom e right	Define as coordenadas para posicionamento de um div	left:0px; top:0px; bottom:100px; right:100px;
border-style border-width border-color	Propriedades relacionadas às bordas	border-style: solid; border-width:5px; border-color:red;

Exemplo 1: layout tradicional



div: topo

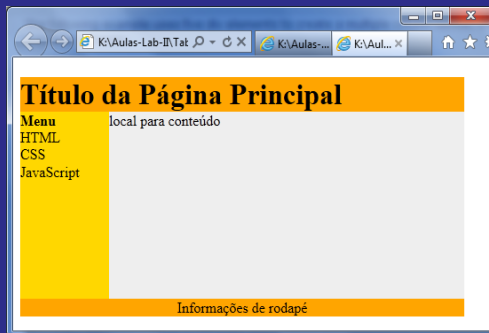
div: conteudo

div: principal

div: menu

div: base

Exemplo 1: layout tradicional



```
<body>
<div id="principal">
  <div id="topo">
    <h1 style="margin-bottom:0;">
      Título da Página Principal</h1>
    </div>
    <div id="menu">
      <b>Menu</b><br>HTML<br>CSS<br>JavaScript
    </div>
    <div id="conteudo">
      local para conteúdo
    </div>
    <div id="base">
      Informações de rodapé
    </div>
  </div>
</body>
```

#principal

```
{ margin:0 auto;
  width:500px;
}
```

#topo

```
{ background:#FFA500;
  height:50px;
}
```

#menu

```
{ background-color:#FFD700;
  width:100px;
  height:200px;
  float:left;
}
```

#conteudo

```
{ background-color:#EEEEEE;
  width:400px;
  height:200px;
  float:left;
}
```

#base

```
{ background-color:#FFA500;
  width:500px;
  height:40px;
  text-align:center;
  float:left;
}
```

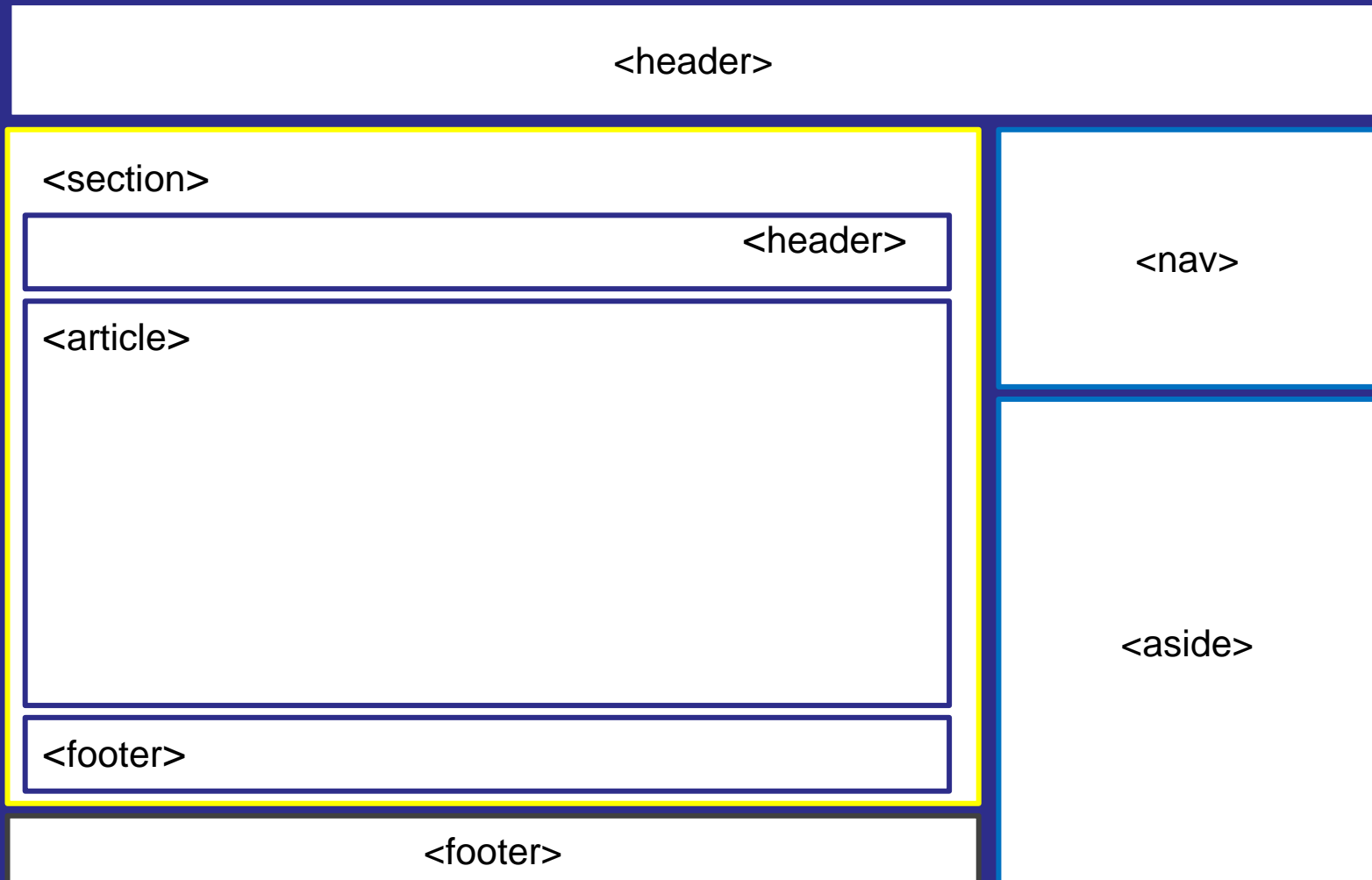
Principais novas Tags – HTML 5

- **<section>**: Demarca uma ou várias áreas genéricas que agrupa elementos de conteúdo dentro da página.
- **<header>**: Demarca o conteúdo relacionado ao cabeçalho da página ou uma sessão (não confundir com a tag <head>). Pode ser utilizado para agrupar índices de conteúdos, campos de busca ou até mesmo logos.
- **<footer>**: Demarca o conteúdo relacionado ao rodapé da página ou a uma sessão.

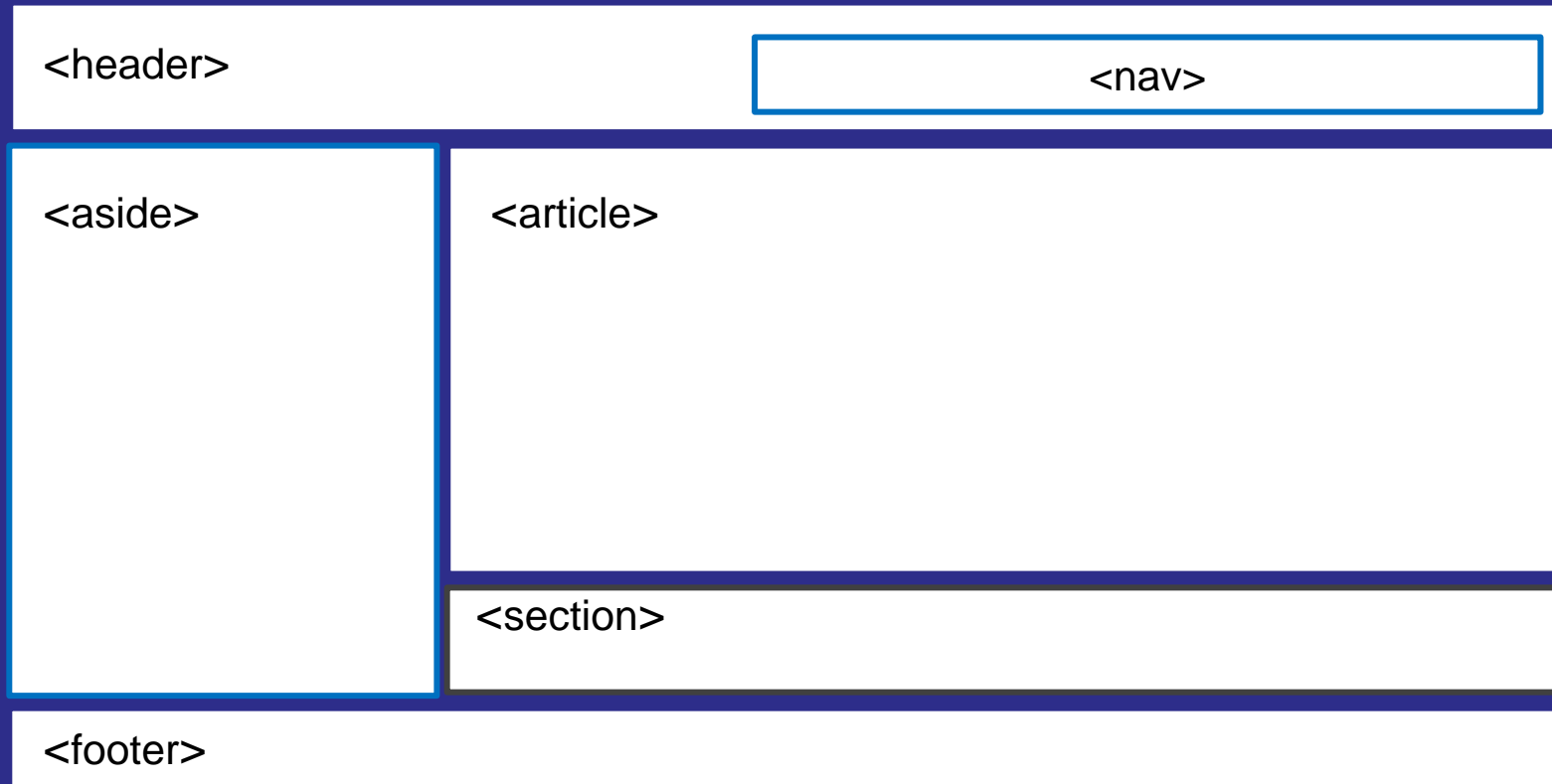
Principais novas Tags – HTML 5

- **<article>**: Demarca uma área de conteúdo autossuficiente: um post, notícia, artigo, um bloco de comentários de usuários ou apenas um bloco de texto comum dentro da página ou em uma sessão.
- **<nav>**: Define uma área de navegação, normalmente o menu da página que leva a outras páginas do site. Nem todos os grupos de links devem ser elementos nav.
- **<aside>**: Define uma área com conteúdo adicional/complementar (arquivos, posts, links, etc...), que está ligeiramente associado ao conteúdo principal.

Layout em HTML 5 – Exemplo 1



Layout em HTML 5 – Exemplo 2



Layout em HTML 5 – Exemplo 3

<header>

<aside>

<nav>

<article>

<article>

<article>

<aside>

<footer>

Exemplo de layout

