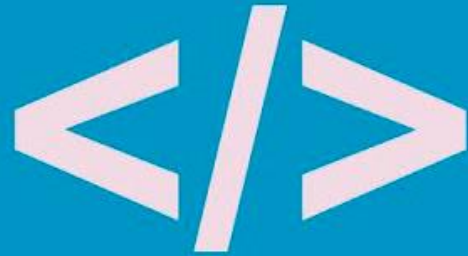


# HTML



```
<!DOCTYPE html>  
<html>  
  <head>  
    <title></title>  
  </head>  
  <body>  
  </body>  
</html>
```

## Estrutura de páginas

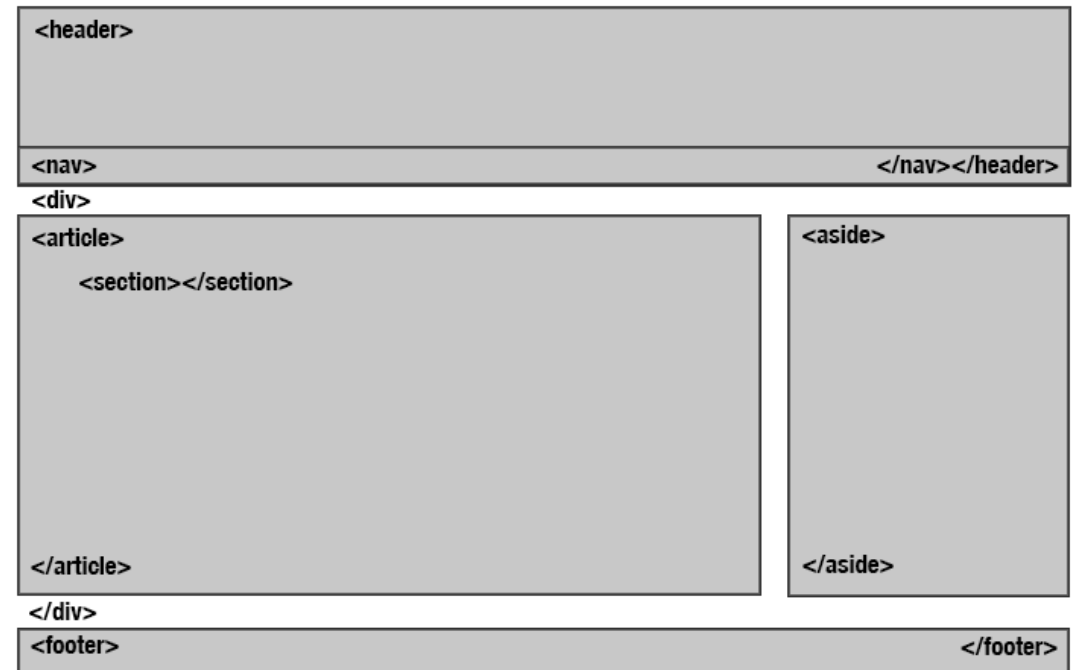
Prof. Djonatan Piehowiak

# Estrutura

---

Páginas apresentam um certo padrão. Em geral, há um cabeçalho (diferente da *tag* de **<head>**), onde se encontra o logo do site, um menu, uma área destinada a textos, uma barra lateral e um rodapé. A estrutura de uma página define onde cada elemento deve estar alocado.

Antigamente, Web designers mantinham o péssimo hábito de utilizar tabelas para estruturar páginas. Tabelas podem ser lidas de diferentes maneiras, dependendo do navegador usado. Utilizar divisórias **<div>** é a maneira correta para estruturar elementos em uma página. Recentemente, HTML5 adotou novas *tags* que permitem uma melhor organização do código. Veja abaixo o exemplo da estrutura de uma página e onde pode ser utilizado cada elemento. A seguir explicaremos cada um.



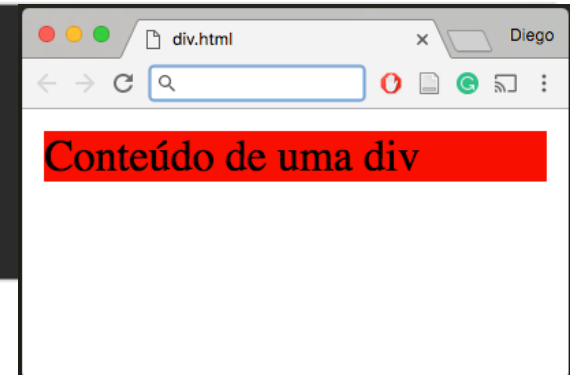
# Divisórias <div>

Divisórias (ou simplesmente divs) permitem, além da organização do código, a aplicação de estilos a trechos específicos da página.

```
1 <div>
2   <p>Conteúdo de uma div</p>
3 </div>
```

Divs não tem forma, cor ou tamanho pré-estabelecido. Isso dependerá do conteúdo alocado nela. Entretanto, para podermos visualizar modificações na estrutura da div, vamos aplicar a cor vermelho ao fundo div por meio da propriedade *background-color*.

```
1 <div style="background-color:red">
2   <p>Conteúdo de uma div</p>
3 </div>
```



# Definindo o tamanho de divs

---

Podemos definir o tamanho de uma div utilizando as propriedades *width* (largura) e *height* (altura). Tamanhos de divs são preferencialmente definidos em pixels.

```
1 <div style="background-color:red; width:100px; height:100px"> <p>Conteúdo de uma div</p> </div>
```

Observe que a div agora tem o tamanho fixo de 100 pixels de altura por 100 pixels de largura. Quando as propriedades de altura e largura estão definidas, o texto se ajusta à div.



# Posicionando divs

---

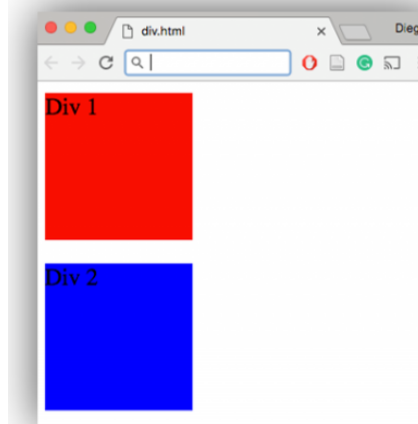
Divs por padrão são posicionadas abaixo de outras. Para compreender como alterar o posicionamento de divs precisamos inicialmente compreender a propriedade *position*.

Há três tipos de posicionamento de divs: *fixed*, *relative* e *absolute*. Divs do tipo *fixed* são posicionadas em um ponto fixo na tela. Divs do tipo *relative* são posicionadas com base no seu canto superior esquerdo. Já as divs do tipo *absolute* são posicionadas com base em sua div pai. Por padrão, todas as divs são declaradas como *relative*. Observe a diferença entre uma div do tipo *relative* para uma div do tipo *absolute*:

---

**Relative**

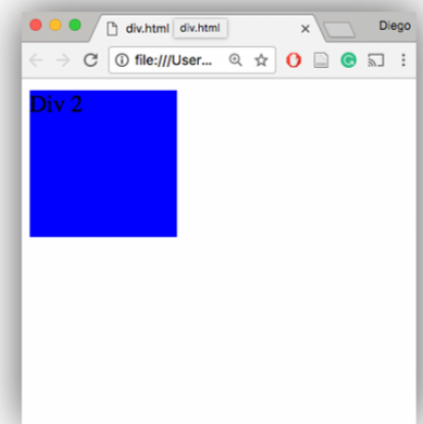
---



---

**Absolute**

---



# Posicionando divs

---

Uma div do tipo *relative* se posiciona por padrão abaixo de outra div. A div do tipo *absolute* se posiciona com base na div pai, ou seja, se há duas divs *absolute* filhas da mesma div pai e com posicionamento não declarado, as duas irão se sobrepor. No exemplo anteriormente descrito, a última div declarada (Div 2) sobrepõe a primeira (Div 1).

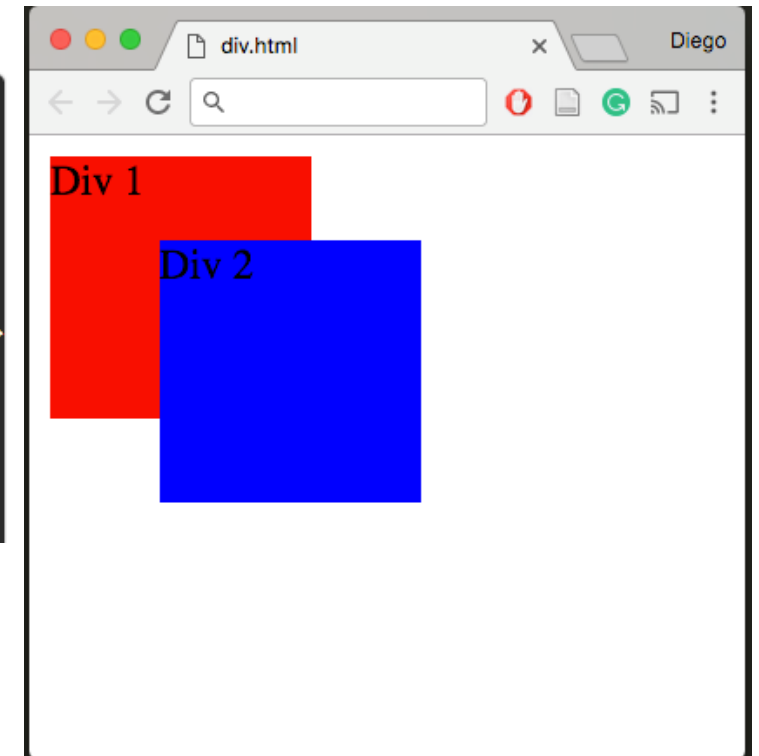
Bom, aprendemos os tipos de posicionamento das divs, mas como posicioná-las na prática? Usando as propriedades: *left*, *right*, *top* e *bottom*.

# Posicionamento divs

---

A propriedade *left* regula a distância para a margem esquerda, *right* para a direita, *top* para o topo da página e *bottom* a distância até o rodapé. Vamos utilizar o exemplo anterior e inserir as propriedades *left* e *top* para que a Div 2 seja posicionada a uma distância de 40 pixels do topo e de 50 pixels da margem esquerda.

```
<div style="background-color:red; width:100px; height:100px; position:absolute">
  <p>
Div 1
  </p>
</div>
<div style="background-color:blue; width:100px; height:100px; position:absolute; top: 40px; left:50px">
  <p>
Div 2
  </p>
</div>
```

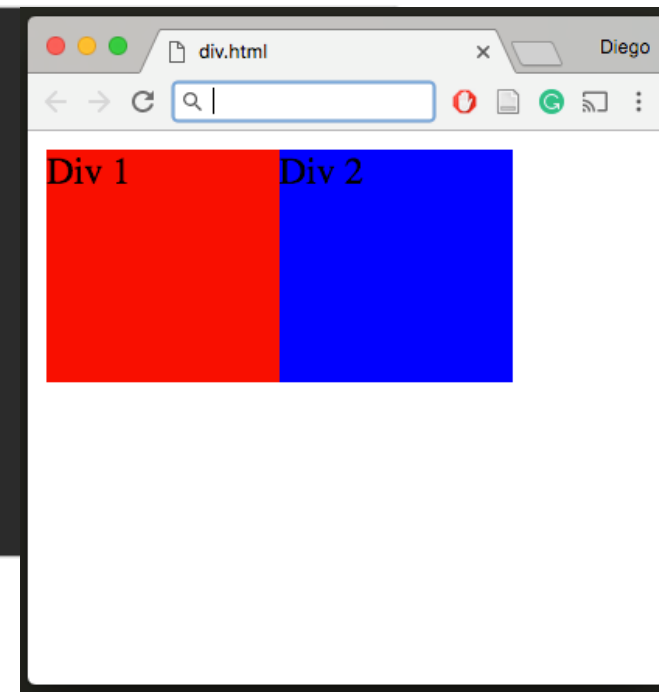


# Propriedade float

---

Como dito anteriormente, divs do tipo *relative* são posicionadas uma abaixo da outra. É possível alterar sua posição nativa usando a propriedade *float*.

```
1 <div style="background-color:red; width:100px; height:100px; float:left">
2 Div 1
3 </div>
4
5 <div style="background-color:blue; width:100px; height:100px; float:left;">
6 Div 2
7 </div>
8
9 <div style="clear:both"></div>
```







# Propriedade float

Observe no código que após aplicar a propriedade *float* a divs, é necessário criar uma div e aplicar a propriedade “clear:both”.

A propriedade *float* altera o fluxo natural de organização de uma página. Isso pode ser utilizado, por exemplo, para inserir uma imagem na página e forçar elementos textuais próximos a contornarem a imagem. Entretanto, isso pode causar efeitos indesejados nas divs declaradas posteriormente, desajustando seus posicionamentos na tela. A propriedade “clear:both” insere um ponto em que o fluxo retorna ao natural, impedindo que próximos elementos sejam alinhados baseado nas propriedades das div anteriores.

float

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean convallis laculis velit eu varius. In lacinia arcu et enim finibus, in vehicula eros scelerisque. Proin tincidunt risus ac lectus finibus, eget pretium nisi rutrum. Praesent aliquet tortor ac volutpat pellentesque. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam congue nisi ut nulla egestas consectetur. Suspendisse ac efficitur justo. Praesent nibh nulla, convallis in interdum porttitor, convallis non ante. Suspendisse vitae egestas leo, sagittis ullamcorper arcu. Curabitur mi urna, ultricies et gravida vitae, imperdiet sit amet augue. Sed elementum massa at mattis venenatis. Sed egestas euismod lectus, id luctus turpis mattis ut.

Nullam sagittis leo augue, at laoreet massa rhoncus vel. Aliquam orci orci, volutpat a diam nec, semper imperdiet nisi. Nulla ut scelerisque est. Quisque vehicula leo ut ante congue laoreet. Donec scelerisque nisi ac quam tristique, in ullamcorper urna elementum. Suspendisse fermentum nisi vitae arcu mattis, quis faucibus velit convallis. Donec laoreet dapibus massa vitae lobortis. Nullam vestibulum, sem ut efficitur venenatis, odio ante malesuada mi, ut efficitur quam velit a arcu. Vivamus ut feugiat tortor, vel accumsan mi. Etiam lobortis justo sit amet massa pharetra consequat nec a est. Duis tristique vestibulum viverra.

Praesent vel massa nisi. Sed id augue diam. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Pellentesque quis gravida nisi. Phasellus ornare ligula elit, sed aliquam ligula volutpat aliquam. Aenean finibus lorem eget elit rutrum porta. Nam euismod dolor quis lacus euismod hendrerit. Phasellus tempus, elit quis eleifend ornare, nibh urna laculis dolor, eget bibendum augue nulla et dolor. Mauris ac semper tortor, nec vestibulum sem. Suspendisse at urna nec justo consectetur eleifend ut quis nisi. Nulla porta ultricies interdum. Proin dignissim odio congue, varius elit nec, tristique diam. Nullam tempor risus non elementum volutpat. Fusce sit amet vestibulum odio, a rutrum nunc. Nullam eleifend nibh et nunc pharetra, at accumsan turpis sagittis.

Nam arcu libero, scelerisque ac dolor vitae, lobortis vestibulum leo. Sed eu massa eget metus egestas hendrerit. Proin pretium finibus ultricies. Donec at ligula arcu. Ut vel sollicitudin erat. Pellentesque vel elit nec diam interdum consectetur. Vivamus auctor dui a nunc placerat porttitor quis at metus. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Proin vel conenodo ex. Fusce tincidunt enim purus, vitae semper magna tristique eu. Sed mi quam, condimentum ac risus id, aliquam condimentum dui. Quisque ultricies conenodo augue, a aliquam ex elementum vel. Donec eros lorem, faucibus nec velit vel, tincidunt molestie sem. Mauris fringilla aliquam neque, vulputate tempus metus lobortis convallis. In ac hendrerit dolor, eu consectetur nunc. Suspendisse mollis vulputate turpis, in condimentum nisi scelerisque non. In eu velit in tortor sodales mollis. Mauris lectus mi, vestibulum vitae lacus in, laoreet lobortis metus. Aliquam erat volutpat. Class aptent taciti

clear:both

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean convallis laculis velit eu varius. In lacinia arcu et enim finibus, in vehicula eros scelerisque. Proin tincidunt risus ac lectus finibus, eget pretium nisi rutrum. Praesent aliquet tortor ac volutpat pellentesque. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam congue nisi ut nulla egestas consectetur. Suspendisse ac efficitur justo. Praesent nibh nulla, convallis in interdum porttitor, convallis non ante. Suspendisse vitae egestas leo, sagittis ullamcorper arcu. Curabitur mi urna, ultricies et gravida vitae, imperdiet sit amet augue. Sed elementum massa at mattis venenatis. Sed egestas euismod lectus, id luctus turpis mattis ut.

Nullam sagittis leo augue, at laoreet massa rhoncus vel. Aliquam orci orci, volutpat a diam nec, semper imperdiet nisi. Nulla ut scelerisque est. Quisque vehicula leo ut ante congue laoreet. Donec scelerisque nisi ac quam tristique, in ullamcorper urna elementum. Suspendisse fermentum nisi vitae arcu mattis, quis faucibus velit convallis. Donec laoreet dapibus massa vitae lobortis. Nullam vestibulum, sem ut efficitur venenatis, odio ante malesuada mi, ut efficitur quam velit a arcu. Vivamus ut feugiat tortor, vel accumsan mi. Etiam lobortis justo sit amet massa pharetra consequat nec a est. Duis tristique vestibulum viverra.

Praesent vel massa nisi. Sed id augue diam. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Pellentesque quis gravida nisi. Phasellus ornare ligula elit, sed aliquam ligula volutpat aliquam. Aenean finibus lorem eget elit rutrum porta. Nam euismod dolor quis lacus euismod hendrerit. Phasellus tempus, elit quis eleifend ornare, nibh urna laculis dolor, eget bibendum augue nulla et dolor. Mauris ac semper tortor, nec vestibulum sem. Suspendisse at urna nec justo consectetur eleifend ut quis nisi. Nulla porta ultricies interdum. Proin dignissim odio congue, varius elit nec, tristique diam. Nullam tempor risus non elementum volutpat. Fusce sit amet vestibulum odio, a rutrum nunc. Nullam eleifend nibh et nunc pharetra, at accumsan turpis sagittis.

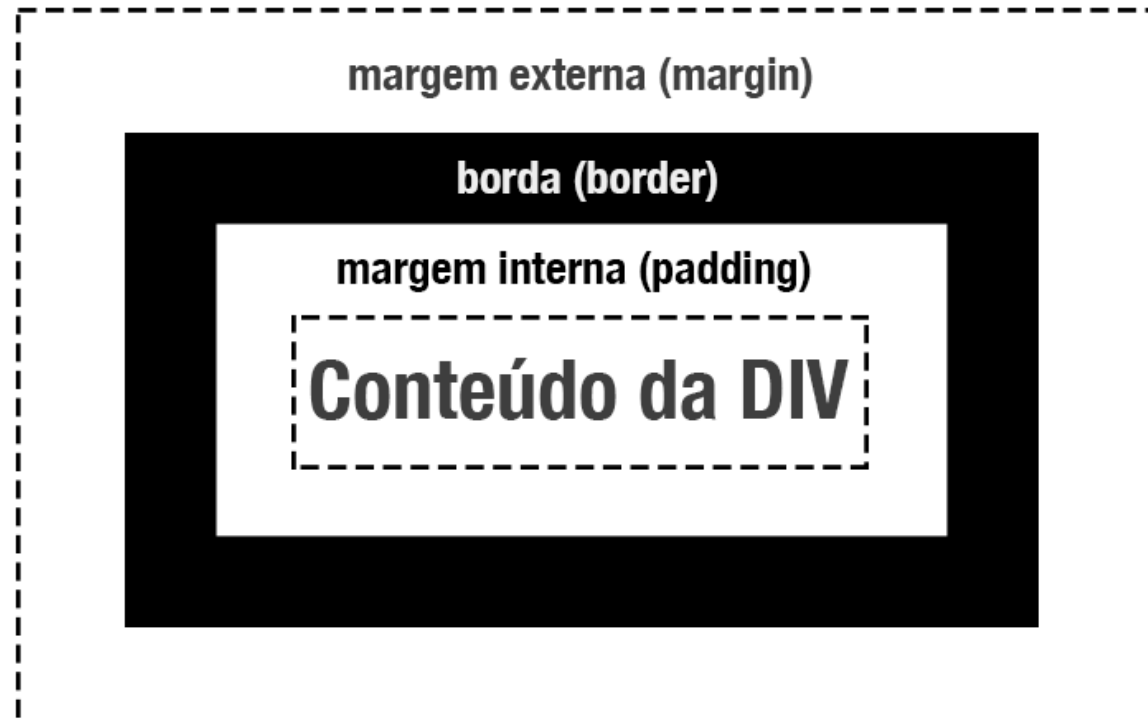
Nam arcu libero, scelerisque ac dolor vitae, lobortis vestibulum leo. Sed eu massa eget metus egestas hendrerit. Proin pretium finibus ultricies. Donec at ligula arcu. Ut vel sollicitudin erat. Pellentesque vel elit nec diam interdum consectetur. Vivamus auctor dui a nunc placerat porttitor quis at metus. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos.

# Margens

---

Há dois tipos de margem: externas (*margin*) e internas (*padding*). As margens externas definem a distância de um elemento para outros elementos. Enquanto isso, margens internas definem a margem entre o conteúdo do elemento e a borda (*border*).

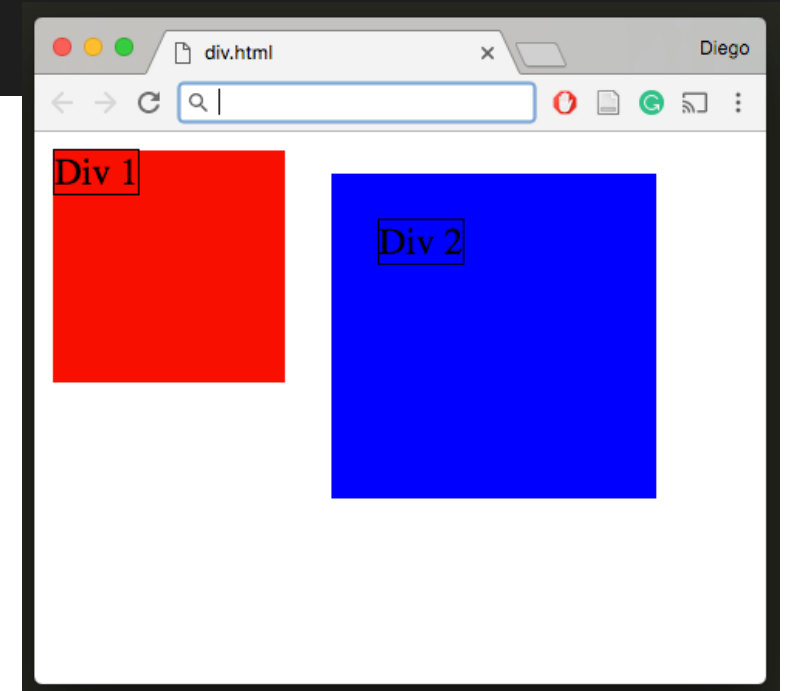


# Margens

A seguir alteraremos as propriedades da Div 2 (posicionada a esquerda da Div 1).

```
1 <div style="background-color: red; width:100px; height:100px; float:left">
2   <span style="border:1px solid black">Div 1</span>
3 </div>
4 <div style="background-color: blue; width:100px; height:100px; float:left; margin: 10px 20px 10px 20px; padding:20px">
5   <span style="border:1px solid black">Div 2</span>
6 </div> <div style="clear:both"></div>
```

Observe que a Div 2 recebe as margens “10px 20px 10px 20px”. Esses números indicam, respectivamente, as margens para: topo, direita, rodapé e esquerda (sentido horário). A Div 2 ainda recebe a propriedade “padding:20px”, que insere uma margem interna de 20 pixels.



# Div container

---

A div container permite que divs internas sejam contidas em uma área delimitada. No exemplo a seguir, vamos inserir três divs delimitadas por uma div pai. A div pai terá largura de 960 pixels com uma margem interna de 10 pixels em cada lado. A primeira div filha terá 100 pixels de altura e ocupará toda a largura da div pai. As outras duas divs filhas deverão estar lado a lado. Observe como a div pai centraliza o conteúdo das outras divs. Vamos colorir o fundo de cada div de uma cor distinta.



# Div container

---

Agora observe o código-fonte:

```
1  <div style="background-color:gray; width:960px; height:auto; margin:0 auto; padding:10px">
2      <div style="background-color:green; width:960px; float:left; margin-bottom:10px">
3          <h1 style="height:100px">Div 1</h1>
4      </div>
5      <div style="background-color:red; width:600px; float:left">
6          <h1 style="height:200px">Div 2</h1>
7      </div>
8      <div style="background-color:blue; width:350px; float:right;">
9          <h1 style="height:200px">Div 3</h1>
10     </div>
11     <div style="clear:both"></div>
12 </div>
```

# Div container

---

Perceba como a div pai recebe a cor cinza (*gray*), a largura de 960 pixels, a altura automática (*auto*), margem superior nula e lateral automática (*auto*) e margem interna de 10 pixels. Quando a altura é declarada como automática, o conteúdo interno se regulará à altura final da div (no exemplo, a altura total será de 300 pixels: 100 pixels da div 1 mais 200 pixels da div 2). A margem foi declarada com altura zero e largura automática. Isso fará com que a div, de largura 960 pixels, tenha margens idênticas tanto no lado direito quanto no lado esquerdo, o que centralizará a div. Essa div será denominada div **container**.

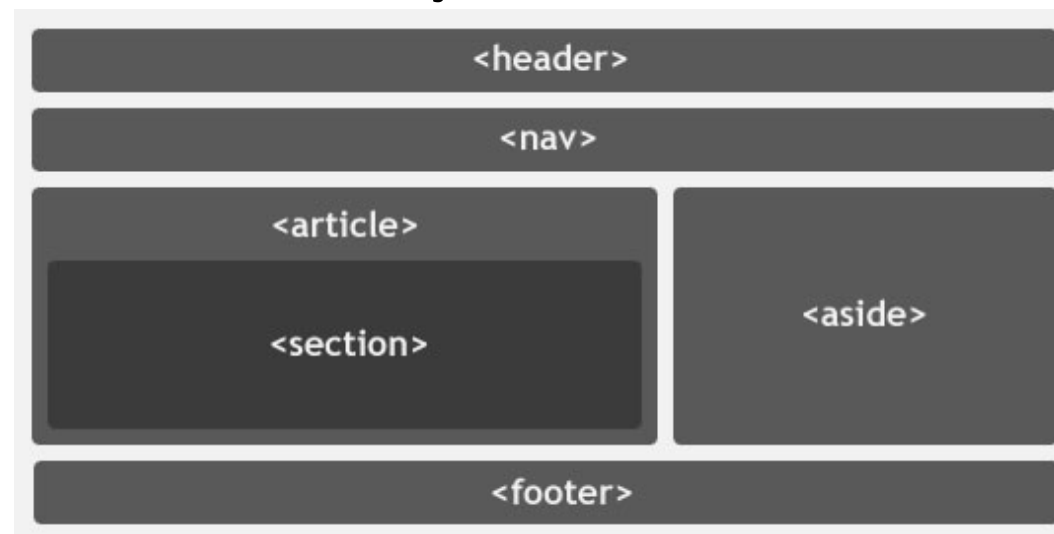
A div container (div pai) delimita os espaços das divs 1, 2 e 3. Observe que a div 2 está alinhada à esquerda, enquanto a div 3 está alinhada à direita. Poderíamos utilizar o alinhamento à esquerda para a div 3, bastaria adicionar uma margem à esquerda para garantir a margem de 10 pixels entre as duas divs.

# Cabeçalho <header> e rodapé <footer>

---

A *tag* <header> delimita a área de página destinada ao cabeçalho. Não confunda com a *tag* <head>, que armazena informações de uma página que não serão exibidas pelo navegador. A *tag* <header> delimita o espaço onde serão exibidas informações introdutórias, logo do site ou mesmo menus.

A *tag* <footer> delimita os elementos que irão compor o rodapé de uma página. Pode ser utilizado para armazenar menus secundários, informações sobre o autor, mensagens de direitos autorais, além de outras informações.





# Cabeçalho <header> e rodapé <footer>

---

Em geral, as *tags* <header> e <footer> são utilizadas para armazenar conteúdos constantes em páginas dinâmicas. Você já notou que ao navegar em um site o conteúdo do cabeçalho e do rodapé não se alteram? Sites dinâmicos utilizam trechos de código que serão repetidos em todas as páginas e os separam de trechos que serão alterados dependendo da página. Observe a seguir como declarar cabeçalhos e rodapés. Note que as duas *tags* devem ser declaradas dentro da *tag* <body>. Observe também que entre elas foi declarada uma *div* que armazenará um possível conteúdo dinâmico.

```
1 <body>
2   <header>Este é o cabeçalho</header>
3   <div>Conteúdo dinâmico</div>
4   <footer>Este é o rodapé</footer>
5 </body>
```

As *tags* <header> e <footer> são containers que não atribuem qualquer posicionamento na página aos seus elementos filhos, entretanto convém utilizá-la corretamente para organizar melhor a página.

# Barra de navegação <nav> e menu <menu>

---

A *tag* <nav> indica uma região da página que contém uma barra de navegação, ou seja, armazena uma lista de links. A *tag* <menu> também armazena uma lista de links. A *tag* <nav> indica a seção do site responsável por armazenar os links de navegação. Enquanto isso, a *tag* <menu> é mais genérica e pode ser utilizada em diversos trechos da página. A seguir vemos um exemplo de como declarar uma barra de navegação. Neste exemplo, os links são organizados a partir de itens de uma lista não ordenada. Também seria possível utilizar a tag <span> para organizá-los.

```
1  <nav>
2      <ul>
3          <li><a href="#">Link 1</a></li>
4          <li><a href="#">Link 2</a></li>
5          <li><a href="#">Link 3</a></li>
6      </ul>
7  </nav>
```

# Artigos <article> e seções <section>

---

A *tag* <article> indica uma parte do site destinada a receber conteúdos relevantes para a página. Sites de busca, como o Google, utilizam essa *tag* para buscar conteúdos exibidos. A *tag* <section> indica uma seção, ou seja, *essa tag divide o conteúdo aplicado a tag <article>*.

Observe o exemplo abaixo. Vamos criar uma página genérica denominada “Minha página”. Observe que inserimos o conteúdo principal da página dentro da *tag* <article>. Cada seção é delimitada pela *tag* <section>.

```
1  <article>
2    <h1>Minha página</h1>
3    <section>
4      <h2>1. Introdução</h2>
5      <p>Esta é minha página.</p>
6    </section>
7    <section>
8      <h2>2. Por que a criei?</h2>
9      <p>Eu a criei para falar sobre minha vida.</p>
10   </section>
11 </article>
```

# Barra lateral <aside>

---

A *tag* <aside> armazena conteúdo complementar para uma página, como por exemplo, menus secundários e publicidade. Essa *tag* é mais utilizada para produção de barras laterais, entretanto seu uso vai além disso, e pode ser utilizada para inserção de qualquer tipo de adendo ao conteúdo principal da página.

```
1  <article>[...]</article>
2  <aside>[...]</aside>
```

Vemos que a *tag* <aside> é utilizada em paralelo a tag <article>. Essa *tag* insere um conteúdo complementar ao artigo.