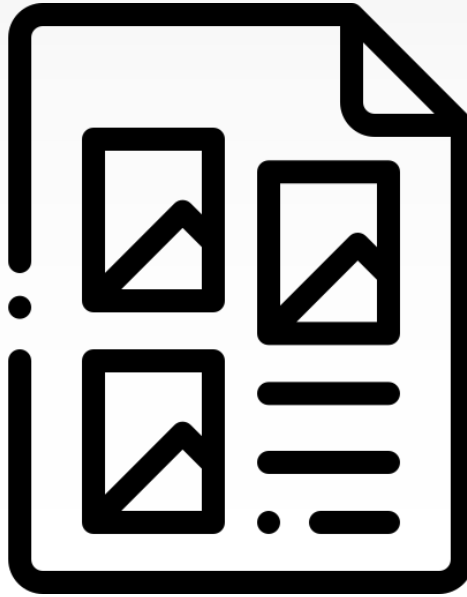


Desenvolvimento para Web

Módulo Básico





CONSTRUÇÃO DE LAYOUT

- *Tipos de Layout*

- Os layouts podem ser **classificados** segundo **dois** critérios:
 1. **Número** de **colunas**
 2. **Largura** da **página**
- Quanto ao **número** de **colunas**, os layouts podem ser de **uma**, **duas**, **três** ou, em casos muito especiais, **multicolunares**
- Quanto à **largura** das **páginas**, os layouts podem ser de **largura fixa**, **líquidos**, **elásticos**, **híbridos** ou **responsivos**

- ***Tipos de Layout***

- ***Largura Fixa***

- A **página** e suas **colunas componentes** têm uma **largura fixa**, **não escalável**, normalmente definida em **pixel**
 - Para **qualquer** que seja o **tamanho** da **janela** do **navegador** ou **resolução**, as **larguras não** variam
 - **Monitores** com **resolução menor** do que a **largura total** adotada apresentam **parcialmente** a **página** no **sentido horizontal**, e o **navegador** coloca uma **barra** de **rolagem** também **horizontal** para **acessar a lateral** fora da **visão**



- ***Tipos de Layout***

- ***Largura Fixa***

- ***Monitores com resolução maior do que a largura escolhida apresentam **espaços horizontais vazios** na tela, em volta da página, que, para resoluções maiores, resulta em mau aproveitamento do **espaço disponível*****
 - ***As larguras normalmente empregadas em sites que usam essa técnica são valores em torno de **760px** para contemplar resoluções de **800 x 600** ou por volta de **1000px** para resoluções de **1024 x 760*****

- ***Tipos de Layout***

- ***Largura Fixa***

- ***Vantagens:***

- ***Perfeito controle que o desenvolvedor tem sobre as dimensões dos componentes do layout, além da garantia de renderização igual em dimensões, independentemente do tamanho de janela adotado pelo usuário***



- ***Desvantagens:***

- ***Mau aproveitamento do espaço horizontal em altas resoluções e a necessidade de rolagem horizontal em resoluções abaixo da largura escolhida***



- ***Tipos de Layout***

- ***Líquido***

- As *larguras* são *definidas* em **porcentagem**
 - *Acomoda-se* em *largura* a **qualquer tamanho** de **janela** ou **resolução** do **monitor**, **otimizando** o aproveitamento do **espaço horizontal disponível**
 - **Layout líquido** é também **conhecido** como **layout fluido**



- ***Tipos de Layout***

- ***Líquido***

- ***Desvantagens:***

- ***Estão relacionadas diretamente ao comprimento das linhas de texto***

- ***Em tamanhos de janela menores, especialmente nos layouts multicolumnares, as larguras das colunas podem ser reduzidas a comprimentos mínimos dificultando a leitura***

- ***A solução para esse caso consiste em definir, para a página, uma largura mínima em pixel, cessando o efeito líquido e forçando o aparecimento de barra de rolagem horizontal, a partir daquela largura***

- ***Tipos de Layout***

- ***Líquido***

- *Em tamanhos de janela e **resoluções maiores**, os comprimentos das linhas de texto podem se tornar **tão extensos** que **dificultam** a **leitura***
 - *Algumas **soluções clássicas** são **adotadas** para **tais casos** e cabe ao **desenvolvedor** **escolher** uma delas ou, mesmo, uma **combinação** de **ambas**, baseando-se em **particularidades** do seu **layout***



- ***Tipos de Layout***

- ***Líquido***

- A primeira solução consiste em definir uma **largura total** para a página, **menor** que a **largura disponível**, por exemplo: ocupar 85% da largura da janela, centrar o layout e obter como resultado margens laterais à **esquerda** e à **direita** da página igual a 7,5% do espaço horizontal total
 - Isso diminui a taxa de crescimento da largura do layout e pode minimizar o efeito de linhas muito extensas

- ***Tipos de Layout***

- ***Líquido***

- ***Outra solução consiste em definir **paddings** e **margens** em **porcentagens**, o que também diminui a taxa de crescimento da largura do layout***
 - ***Definir para a página uma largura máxima em pixel (**cessando o efeito líquido**)***



- *Tipos de Layout*

- *Elástico*

- No **layout líquido**, o comprimento total das linhas de texto podem se tornar **tão reduzido** ou **tão extenso** que afete seriamente a legibilidade
 - Em substituição de definir larguras com base na largura da janela, no **tipo** de **layout elástico**, definimos com **base** no **tamanho** da **fonte**



- ***Tipos de Layout***

- ***Elástico***

- A unidade de medida CSS que se baseia em **tamanho** de **fonte** é o **em** (*usada para definir larguras em layout elástico*)
 - Por escalar com o tamanho de fonte, é particularmente útil em questões de acessibilidade para alguns grupos de pessoas com necessidades especiais de visão

LOREM
iPSuM DolOr siT
amet,
CoNsECtetUr
adiPIscing ELIt

- ***Tipos de Layout***

- ***Elástico***

- ***Desvantagens:***



- ***Nem sempre pode ser possível o aproveitamento máximo do espaço horizontal disponível, além do possível aparecimento de barra de rolagem horizontal a partir de determinado ponto de crescimento do tamanho da fonte, mesmo nas resoluções mais altas em uso***
 - ***Definir um comprimento máximo para a largura total da página pode ser uma solução***

- *Tipos de Layout*

- *Híbrido*

- A *largura* da *página*, de *colunas* e dos *demais elementos componentes* do *layout* são *definidas* em *unidades ems* *adicionalmente* definimos para *tais componentes* uma *largura máxima* em *porcentagem*
 - Há ainda, a *opção* de *inverter* o *conceito definido*, como *larguras* do *layout* em *porcentagem* e *larguras máxima* e *mínima* em *unidades ems*



- *Tipos de Layout*

- *Responsivo*

- A **largura** da **página**, de **colunas** e dos demais **elementos componentes** do **layout** são definidas em **unidades *ems*** ou ***porcentagem*** e **adicionalmente** **definimos** para tais **componentes** uma **largura máxima**
 - Em um **layout responsivo** usamos uma **funcionalidade** das **CSS3** denominada ***media query***, que **permite** ao **desenvolvedor** servir o **layout** de **acordo** com a **largura da viewport**.



- *Tipos de Layout*

- *Responsivo*

- Para **dispositivos móveis** ou com **pequenas larguras de viewport**, o **layout servido** será de **uma coluna** e a partir daí, quando a **largura da viewport do dispositivo** do usuário **atingir determinada largura**, o **layout assume** uma **configuração de duas colunas, três colunas** ou mesmo de **várias colunas**



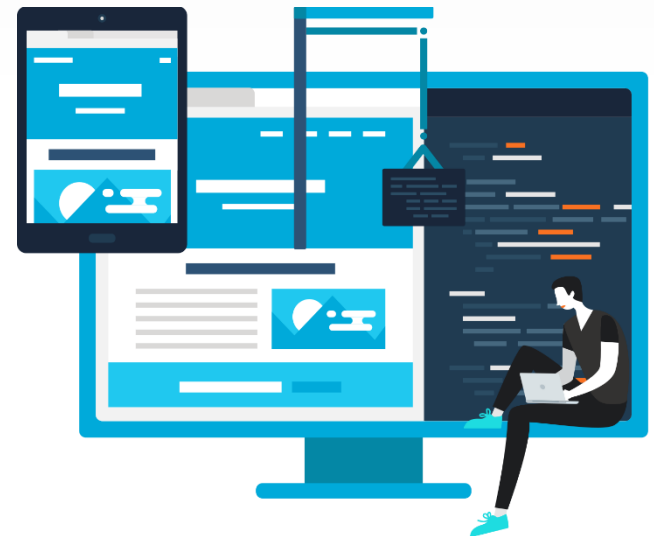
- **Centralizando o Layout**

- Com o **lançamento** de **monitores** com **resoluções** cada vez **maiores** e a **imposição** de se **definir** uma **largura máxima** ou **fixa** para as **páginas**, visando **resolver** o **problema** da **legibilidade** em **linhas** de **texto** muito **extensas**, **centralizar** as **páginas** do site é a forma de se **distribuir** pelos **dois** **lados** da **tela** os **espaços** **horizontais** **não** **aproveitados**, **minimizando** ou, pelo **menor**, **equalizando** o **efeito** de **áreas** **vazias** na **tela**
- A **prática** de **centralizar horizontalmente** a **página** tem sido **amplamente** **empregada** e, trata-se de um **efeito** **simples** de se **conseguir** com **uso** de **regras** **CSS**

- ***Centralizando o Layout***

- ***Existem dois métodos para centralizar:***

- ***Declarar margens automáticas para a página, com uso da propriedade `margin`***
 - ***Posicionar de `forma relativa` no centro da tela e usar declaração de `margem negativa` para obter o efeito final***



- **Centralizando o Layout**

- **Centralizar com uso de margens automáticas**

- Baseia-se no cálculo que o agente de usuário (browser) faz para encontrar o valor **auto** quando este for declarado para a propriedade **margin**
 - As recomendações do W3C para as CSS preveem que, quando o valor **auto** for declarado para as **margens esquerda** e **direita** de um elemento com largura explicitamente especificada, o agente de usuário deverá calcular as margens horizontais **subtraindo** a largura total disponível (**largura** do **elemento-pai**) da largura declarada para o elemento e **dividir** o resultado da subtração por dois, tornando o valor encontrado como **margens esquerda** e **direita** (**centraliza** o **elemento dentro** do seu **elemento-pai**)

- ***Centralizando o Layout***
 - ***Centralizar com uso de margens automáticas***
 - ***Considere a seguinte marcação HTML***

```
<body>
<div id="tudo">
  <h1> Cabeçalho nível 1 </h1>
    <p> Lorem ipsum dolor sit amet, ... </p>
    <p> Ut sollicitudin sodales purus. ... </p>
  <h2> Cabeçalho nível 2 </h2>
    <p> Duis posuere odio sed velit ... </p>
    <p> Curabitur hendrerit ... </p>
    <p> Quisque sodales lectus ac orci. ... </p>
</div>
</body>
```

- ***Centralizando o Layout***
 - ***Centralizar com uso de margens automáticas***
 - ***Aplique a seguinte folha de estilo***

```
#tudo {  
  width: 60%;  
  margin: 0 auto;  
  border: 1px solid #000;  
}
```

- **Centralizando o Layout**

- **Centralizar com uso de margens automáticas**

- A **marcação** simula os conteúdos de uma página contida dentro de um **div**, identificada com o par **atributo/valor** igual a **id="tudo"**
 - O **elemento-pai** de uma página é o **elemento body**, isto é, a **área de renderização na tela do usuário**
 - Por sua vez, **div#tudo** engloba tudo o que está contido em **body**
 - Dessa forma, **centralizando o div**, estaremos **centralizando a página**



- **Centralizando o Layout**
 - **Centralizar com uso de margens automáticas**

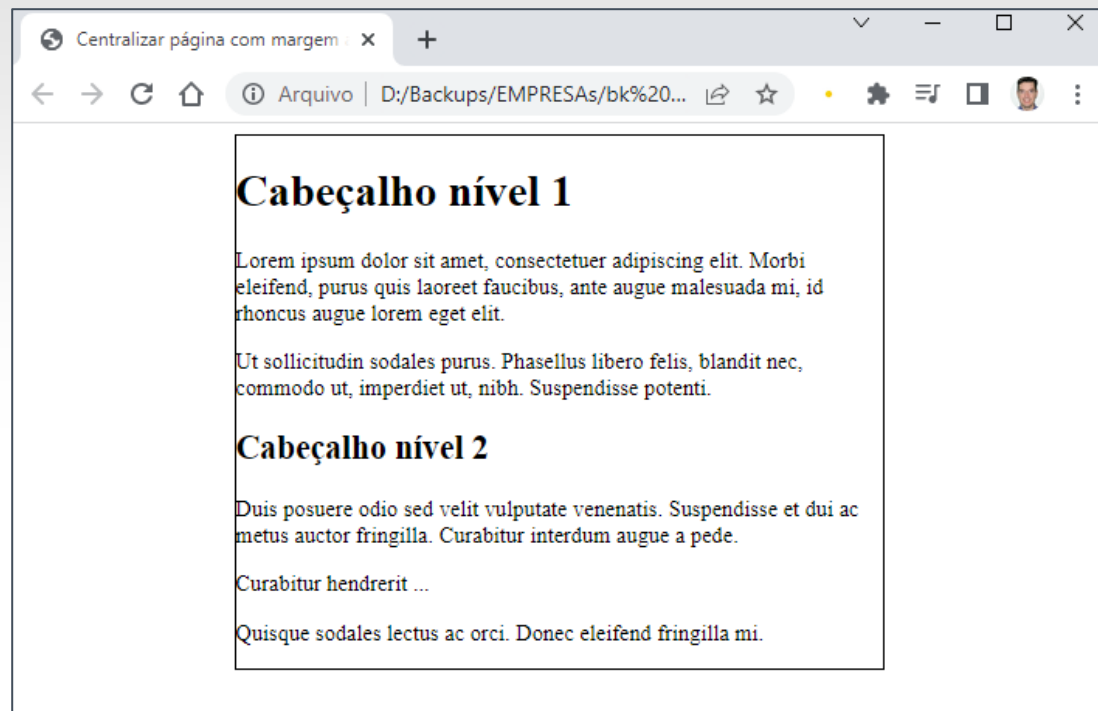


Figura 1 – Centralizar com margem

- **Layout de largura fixa**

- A estrutura típica de um layout de **duas colunas** com **largura fixa**, e que adotaremos como exemplo, consiste de uma área para o **topo** do site contendo logotipo e identificação do site, uma **coluna** de **navegação** e outra **coluna** de **conteúdo principal**, as quais chamaremos de **topo**, **auxiliar** e **principal**, respectivamente, e um fechamento que denominamos **rodapé**
- Toda a estrutura estará contida em um elemento-pai, **container geral** do layout (**div** com nome identificador **tudo**)

- ***Layout de largura fixa***
 - ***A marcação HTML é apresentada abaixo***

```
<body class="home">
<div id="tudo">
  <div class="topo">
    <header class="topo_header">
      <hgroup>
        <h1><a href="/"> Site do Geraldo </a></h1>
        <h2> Site destinado a divulgar o trabalho de Geraldo Neto </h2>
      </hgroup>
    </header>
  </div> <!-- /.topo -->
```

- *Layout de largura fixa*
 - *A marcação HTML é apresentada abaixo*

```
<div class="principal">
  <main>
    <h1> Título UM </h1>
    <section class="principal_secaoum">
      <h2> Sub-título um </h2>
      <p> Lorem ipsum dolor sit amet, ... </p>
      <h2> Sub-título dois </h2>
      <p> Curabitur hendrerit erat ut augue. ... </p>
    </section>
  </main>
</div> <!-- /.principal -->
```

- *Layout de largura fixa*
 - *A marcação HTML é apresentada abaixo*

```
<div class="auxiliar">
  <nav class="auxiliar_navegacao">
    <h2> Menu </h2>
    <ul>
      <li><a class="corrente" href="../html/home.html">home</a></li>
      <li><a href="../html/portfolio.html"> portfólio </a></li>
      <li><a href="../html/artigos-html.html"> artigos HTML </a></li>
      <li><a href="../html/artigos-css.html"> artigos CSS </a></li>
      <li><a href="../html/contato.html"> contato </a></li>
    </ul>
  </nav>
```

- ***Layout de largura fixa***
 - ***A marcação HTML é apresentada abaixo***

```
<aside class="auxiliar_parceiros">  
  <ul class="auxiliar_parceiros_textos">  
    <li><a href="#"> Parceiro Um </a></li>  
    <li><a href="#"> Parceiro Dois </a></li>  
    <li><a href="#"> Parceiro Três </a></li>  
    <li><a href="#"> Parceiro Quatro </a></li>  
  </ul>
```

- ***Layout de largura fixa***
 - ***A marcação HTML é apresentada abaixo***

```
<div class="auxiliar_parceiros_imagens">  
  <figure>  
    <a href="#">  
        
    </a>  
  </figure>  
  <figure>  
    <a href="#">  
        
    </a>  
  </figure>
```

- ***Layout de largura fixa***
 - ***A marcação HTML é apresentada abaixo***

```
<figure>
  <a href="#">
    
  </a>
</figure>
<figure>
  <a href="#">
    
  </a>
</figure>
```

- ***Layout de largura fixa***
 - ***A marcação HTML é apresentada abaixo***

```
<figure>
  <a href="#">
    
  </a>
</figure>
</div>
</aside>
</div> <!-- /.auxiliar -->
```


- ***Layout de largura fixa***
 - ***A marcação HTML é apresentada abaixo***

```
<div class="rodape">  
  <footer>  
    <small>Site do Geraldo - 2022 &copy; Todos os direitos reservados</small>  
  </footer>  
</div> <!-- /.rodape -->  
</div> <!-- /.tudo -->  
</body>  
</html>
```

- ***Layout de largura fixa***
 - ***Renderização da marcação sem estilização***

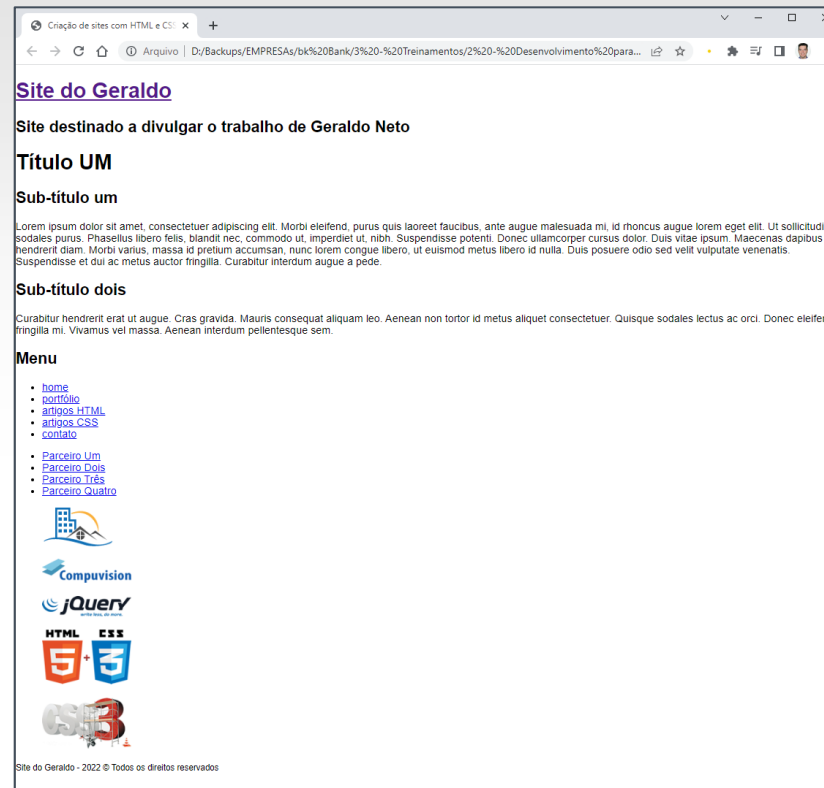


Figura 2 – Layout sem estilização

- **Layout de largura fixa**
 - **Renderização da marcação sem estilização**

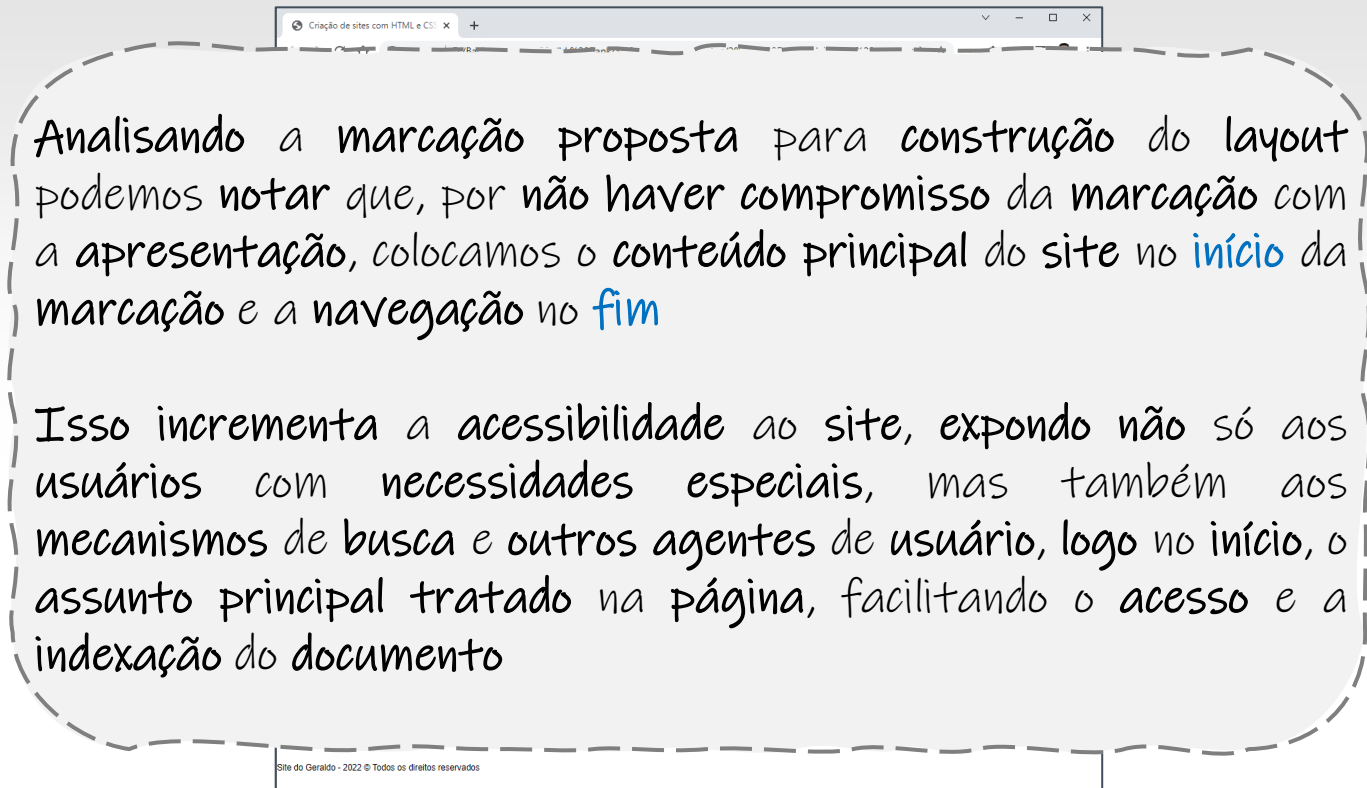


Figura 2 – Layout sem estilização

- ***Layout de largura fixa***
 - ***Premissas para a construção do layout***
 - ***Página centralizada na janela do navegador***
 - ***Largura total da página igual a 960px***
 - ***Coluna principal à direita com 720px e navegação à esquerda com 220px, resultando em um espaço de 20px entre elas***
 - ***Altura do rodapé igual a 25px***



- *Layout de largura fixa*
 - *Existem dois métodos para posicionar as colunas com float*
 1. *Flutuar ambas as colunas à direita e definir uma margem entre elas para separá-las em 20px*
 2. *Flutuar uma para cada lado, deixando que a diferença da soma de suas larguras para a largura total resulte nos 20px de separação*
 - *Analise o seguinte cálculo: $960\text{px} - (720\text{px} + 220\text{px}) = 20\text{px}$*

- ***Layout de largura fixa***

- ***A folha de estilo para cumprir as premissas do layout:***

```
<link rel="stylesheet" href='normalize.css'></link>
<style rel="stylesheet">
*, *:before, *:after { /* regras para alterar o box model padrão */
  -webkit-box-sizing: border-box;
  -moz-box-sizing: border-box;
  box-sizing: border-box;
}
.clear { clear: both; } /* classe para "limpar" elementos flutuados */
```

- ***Layout de largura fixa***
 - ***A folha de estilo para cumprir as premissas do layout:***

```
body {  
    font: 18px/1.4 arial, sans-serif;  
    color: #333;  
}  
#tudo { /* centraliza a página de 960px de largura */  
    width: 960px;  
    margin: 0 auto;  
}
```

- ***Layout de largura fixa***
 - ***A folha de estilo para cumprir as premissas do layout:***

```
.topo_header h1 {  
    font-size: 36px;  
    margin: 4px 0 2px 0;  
    text-align: right;  
}  
.topo_header a {  
    color: #900;  
    text-decoration: none;  
}
```


- ***Layout de largura fixa***

- ***A folha de estilo para cumprir as premissas do layout:***

```
.topo_header h2 {  
    font-size: 20px;  
    margin: 2px 0 4px 0;  
    text-align: right;  
}  
.principal { /* flutua à direita */  
    width: 720px;  
    float: right;  
}
```

- ***Layout de largura fixa***

- ***A folha de estilo para cumprir as premissas do layout:***

```
.auxiliar { /* flutuar à esquerda */  
    width: 220px;  
    float: left;  
}  
.rodape { /* "limpa" as colunas flutuadas e define altura do rodapé */  
    clear: both;  
    height: 25px;  
}
```

- ***Layout de largura fixa***
 - ***Observe na folha de estilo utilizada:***
 - O link para a ***folha de estilo*** normalize.css
 - A ***página*** foi ***centralizada*** adotando-se o ***método*** das ***margens automáticas***
 - As ***regras de estilos*** para o ***topo*** do ***site*** são ***conforme*** ***apresentada em exemplos anteriores***



- ***Layout de largura fixa***
 - ***A renderização do layout com a aplicação da CSS***



Figura 3 – Estilização inicial do layout de duas colunas

- *Layout de largura fixa*

- *Na sequência, criaremos regras de estilos para **ajustar** e **alterar** a apresentação do **topo** e do **rodapé**, bem como definir uma **cor** de **fundo** cinza para o elemento **body** e **branca** para a **área** das **colunas** do **site***
- *Considere as seguintes **alterações** na **folha** de **estilo** do **exemplo** anterior*

```
body {  
    font: 18px/1.4 arial, sans-serif;  
    color: #333;  
    background: #ccc; /* cor de fundo cinza */  
}
```

- **Layout de largura fixa**
 - Considere as seguintes **alterações** na **folha de estilo** do **exemplo anterior**

```
#tudo {  
    width: 960px;  
    margin: 0 auto;  
    background: #fff; /* cor de fundo branca */  
}  
h1, h2, h3, h4, h5, h6 { color: #036; }  
.topo_header {  
    background: #036; /* cor de fundo azul para o topo */  
    padding: 5px 20px;  
}
```

- *Layout de largura fixa*
 - *Considere as seguintes alterações na folha de estilo do exemplo anterior*

```
.topo_header a {  
    color: #900;  
    color: #fff;  
    text-decoration: none;  
}
```

- *Layout de largura fixa*
 - *Considere as seguintes alterações na folha de estilo do exemplo anterior*

```
.topo_header h2 {  
    font-size: 20px;  
    margin: 2px 0 4px 0;  
    text-align: right;  
    color: #fff;  
}
```


- **Layout de largura fixa**
 - **Considere as seguintes alterações** na **folha de estilo** do **exemplo anterior**

```
.rodape {  
  clear: both;  
  height: 25px;  
  background: #036; /* cor de fundo azul para o rodapé */  
  color: #fff;  
  padding: 5px 0;  
  text-align: center;  
}
```

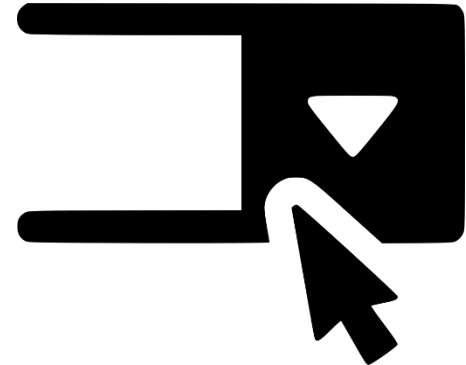
- **Layout de largura fixa**
 - **Estilização da cor de fundo de *body* e da área de conteúdos e a estilização do topo e do rodapé após alteração da CSS**



Figura 4 – Estilização do topo e rodapé

- *Layout de largura fixa*

- *Na sequência, criaremos um **menu** do **tipo botões** com **efeito rollover***
- *Esse efeito consiste em fazer com que a **estilização** do botão **mude** quando o **usuário colocar o ponteiro do mouse sobre ele** e **volte** ao **estado inicial** quando o **ponteiro do mouse for retirado de sobre ele***
- *No exemplo, procedemos com a **alteração da cor de fundo** do botão para obter o **efeito rollover***



- ***Layout de largura fixa***
 - ***Observe as regras CSS para estilizar o menu***

```
.principal {
```

```
...
```

```
padding-right: 15px;
```

```
}
```

```
.auxiliar {
```

```
...
```

```
padding-left: 15px;
```

```
}
```

/ as declarações anteriores estabelecem margens para os conteúdos das colunas */*

- ***Layout de largura fixa***
 - ***Observe as regras CSS para estilizar o menu***

```
/* Novas regras CSS */  
a { text-decoration: none; } /* retira o sublinhado do link */  
.auxiliar_navegacao h2 { /* centraliza o título do menu */  
    text-align: center;  
}  
ul {  
    list-style: none;  
    padding: 0;  
    margin: 0;  
}
```

- **Layout de largura fixa**
 - **Observe as regras CSS para estilizar o menu**

```
/* Novas regras CSS */  
a { text-decoration: none; } /* retira o sublinhado do link */  
.auxiliar_navegacao h2 { /* centraliza o título do menu */
```

As declarações de estilo para o elemento `ul` visam retirar o marcador (bolinhas à esquerda) dos itens da lista e zerar `padding` e `margin` padrão das linhas

```
padding: 0;  
margin: 0;  
}
```

- ***Layout de largura fixa***
 - ***Observe as regras CSS para estilizar o menu***

```
.auxiliar_navegacao a {  
    background: #ff6e19;  
    color: #fff;  
    display: block;  
    margin-bottom: 1px;  
    line-height: 2;  
    padding-left: 10px;  
    text-transform: capitalize;  
}  
.auxiliar_navegacao a:hover { /* efeito rollover */  
    background: #036;  
}
```

- **Layout de largura fixa**
 - **Estilização do menu do site**



Figura 5 – Estilização do menu

- **Layout de largura fixa**
 - *Na sequência, estilizaremos a **área** dos **parceiros***
 - *Repare que essa **área** está **dividida** em **duas outras áreas**, uma contendo **links** de **textos** e outra contendo **links** de **imagens** dos **parceiros***
 - *Observe as regras **CSS** para **estilizar** a **área** de **links** de **textos***

```
.auxiliar_parceiros_textos {  
    margin: 30px 0;      /* cria espaçamento acima e abaixo */  
    text-align: center;  /* centraliza os textos dos links */  
}
```

- ***Layout de largura fixa***
 - ***Observe as regras CSS para estilizar a área de links de textos***

```
.auxiliar_parceiros_textos li {  
    margin-bottom: 3px; /* cria espaçamento entre links */  
    background: #eee; /* cor de fundo dos links */  
}  
.auxiliar_parceiros_textos a {  
    padding: 8px 0;  
    color: #333; /* cor dos textos dos links */  
    display: block; /* toda a área do link será clicável */  
}
```

- **Layout de largura fixa**
 - **Estilização da área parceiros**



Figura 6 – Estilização da área de parceiros

- **Layout Elástico**

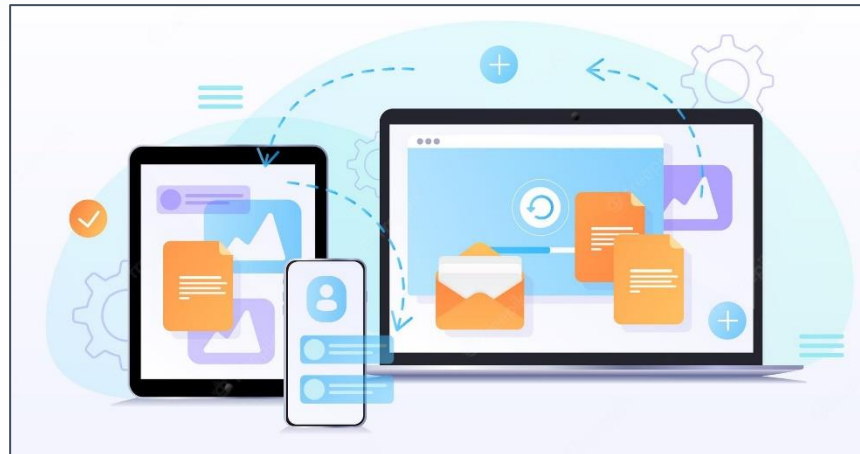
- Nesse tipo de **layout** as **larguras** são definidas em **porcentagem**
- O layout acomoda-se em largura a **qualquer tamanho** de **janela** do **navegador** ou **resolução** do **monitor**, otimizando o aproveitamento do **espaço horizontal** disponível
- Para **larguras** de **janela maiores** ou **menores** que um determinado tamanho, definimos uma **largura máxima** e uma **largura mínima** de modo que o **layout** passe a ser de **largura fixa** quando a largura da janela estiver fora da faixa entre **mínima** e **máxima** (**estabelecida** pelo **desenvolvedor**)

- **Layout Elástico**

- *Para exemplificar, estabeleceremos como ponto de partida os valores 1200px para largura máxima e 760px para largura mínima (os valores depende de vários fatores inerentes a cada layout)*
- *O desenvolvedor deverá realizar experiências em diferentes tamanhos de janela e considerar diferentes resoluções de monitor para estabelecer essas larguras*
- *Podemos adotar os valores que bem entender em nossos projetos*

- **Layout Elástico**

- Para exemplificar esse tipo de layout, usaremos a **marcação HTML e estilização** do **exemplo anterior**, considerando uma **largura máxima** de **1200px**, sendo **960px** para a **coluna principal**, **220px** para a **coluna auxiliar** e **20px** de **distância entre elas**



- ***Layout Elástico***

- ***Observe a Tabela 1 de correspondência das dimensões fixas com porcentagens***

Referência	Pixel	% calculada
Largura total	1200px	100%
Coluna principal	960px	80%
Coluna auxiliar	220px	18.3333%
Espaçamento colunas	20px	1.6666%

Tabela 1 – Larguras fixas e porcentagens

- ***Layout Elástico***

- *As alterações a serem marcadas na folha de estilo anterior (**largura fixa**) são apresentadas a seguir:*

```
#tudo {  
    width: 100%;  
    max-width: 1200px;  
    min-width: 760px;  
    margin: 0 auto;  
    background: #fff;  
}
```


- ***Layout Elástico***

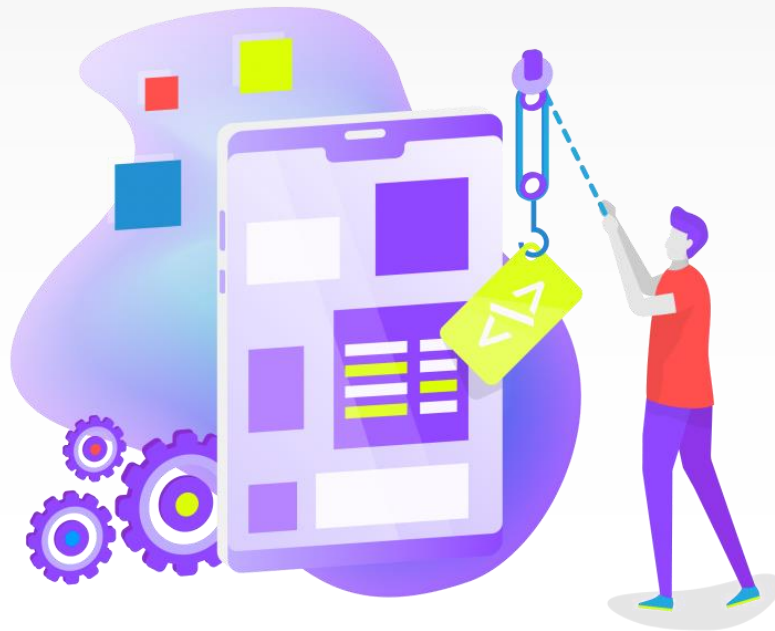
- *As alterações a serem marcadas na folha de estilo anterior (**largura fixa**) são apresentadas a seguir:*

```
.principal {  
  width: 80%;  
  float: right;  
  padding-right: 15px;  
}
```

- ***Layout Elástico***

- As ***alterações a serem marcadas na folha de estilo anterior*** (***largura fixa***) são ***apresentadas a seguir:***

```
.auxiliar {  
  width: 18.3333%;  
  float: left;  
  padding-left: 15px;  
}
```



EXERCÍCIOS

- ***(1) Exercício***

- ***Abra o arquivo no navegador e movimente a janela para diferentes resoluções observando o comportamento do layout***
- ***Retire as declarações de max-width e min-width do div#tudo e observe o comportamento do layout***
- ***Faça experiências com outros valores de porcentagem***

- **(2) Exercício**
 - *Baseando no layout escolhido (última aula) elabore um layout concorrente, todavia, agora considerado a temática “Copa de Mundo 2022”*
 - *Se preferir, desempenhe o exercício em dupla*
 - *Como dica, utilize qualquer ferramenta como auxílio, exceto framework*
 - *Seria oportuno utilizar uma ferramenta, a sua escolha, para gerenciar as tarefas que deverão ser realizadas para o êxito do exercício*

Referências

Silva, M. S. Fundamentos de HTML5 e CSS3. 1. ed. São Paulo: Novatec, 2015.

Duckett, J. HTML e CSS Projete e Construa Websites. 1. ed. Rio de Janeiro: Alta Books, 2011.

Hyslop, B., Castro, E. HTML and CSS: Visual Quickstart Guide. 8. ed. Barcelona: Peachpit Press, 2013.

