

Aula 06 – Criação de Menus

div, nav, section, article, footer são tags para organização, eles são tags que pegam a largura < > da tela inteira

span é uma tag que se adequa ao tamanho do seu interior. Criamos, primeiramente, uma lista não ordenada.

- Englobamos essa lista com tag **nav** – navegação
- Todo o menu deve estar entre <nav>... </nav>
- Aplicamos as formatações para que o menu fique no topo da página.
- Definimos o display do menu como bloco,

display: block;

- Retiramos os marcadores da lista não ordenada e definimos a posição como absoluta – para que possa passar sobre os outros elementos da tela, e definimos a sua posição na tela:

List-style: none; /* estilo do marcador */

Position: absolute; /* posição da lista não ordenada, em relação ao resto da página */

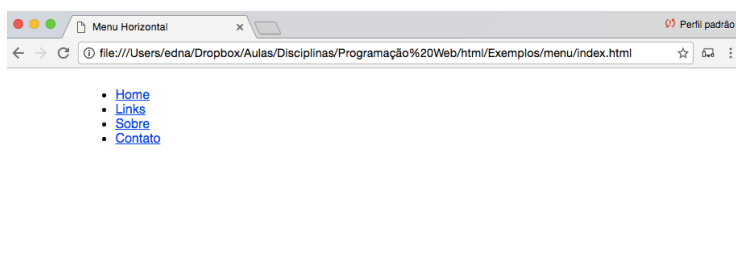
Top: 0px; /* posição topo */

Left: 100px; /* posição esquerda */

- Definimos os itens da lista para exibição in-line (transformamos em uma única linha)

Display: inline-block;

```
1  <!DOCTYPE HTML>
2  <html lang="pt-br">
3  <head>
4      <meta charset="UTF-8">
5      <title>Menu Horizontal </title>
6      <link rel="stylesheet" type="text/css" href="/css/estilo.css" />
7  </head>
8  <body>
9  <nav>
10     <ul class="menu">
11         <li><a href="index.html">Home</a></li>
12         <li><a href="#links">Links</a></li>
13         <li><a href="sobre.html">Sobre</a></li>
14         <li><a href="mailto:seuemail@site.com.br">Contato</a></li>
15     </ul>
16 </nav>
17 </body>
18 </html>
```

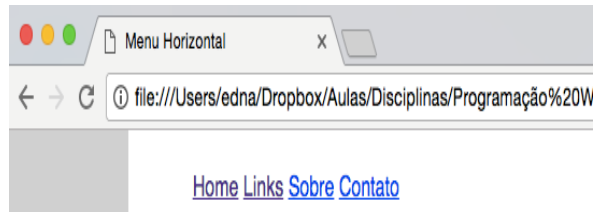


A formatação da tag nav, ul e li

```
nav#menu{
  display: block;
}

nav#menu ul{
  list-style: none;
}

nav#menu li{
  display: inline-block;
}
```



Acrescentando cor de fundo e padding // Retirando o sublinhado do link

```
nav#menu li{
  display: inline-block;
  background-color: #d0d0d0;
  padding: 10px;
}
```

```
nav#menu a{
  text-decoration: none;
  color: #000;
}
```

```
nav#menu ul{
  list-style: none;
  position: absolute; /*posicao da UL em relação ao resto da página*/
  top: 0px; /*posição topo*/
  left: 800px; /* posição esquerda*/
}
```

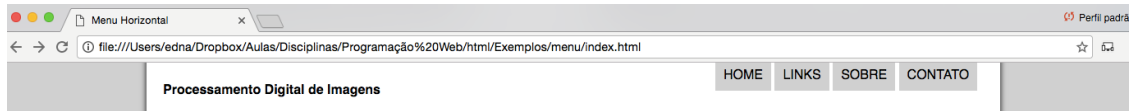


Transformando em maiúsculo –
text-transform:uppercase

```
nav#menu ul{
  list-style: none;
  position: absolute; /*posicao da UL em relação ao resto da página*/
  top: -20px; /*posição topo*/
  left: 800px; /* posição esquerda*/
  text-transform: uppercase;
}
```

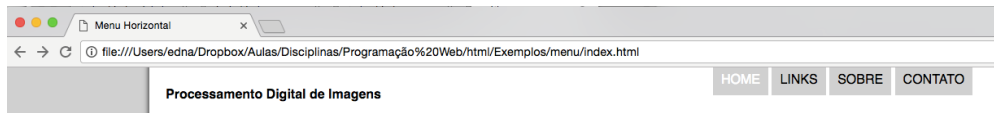
Acrescentei uma sombra e fixei o tamanho da id interface

```
div#interface{
  width: 1000px;
  background-color: #ffffff;
  margin: -20px auto 0px auto;
  padding: 20px;
  box-shadow: 0px 0px 10px black;
}
```



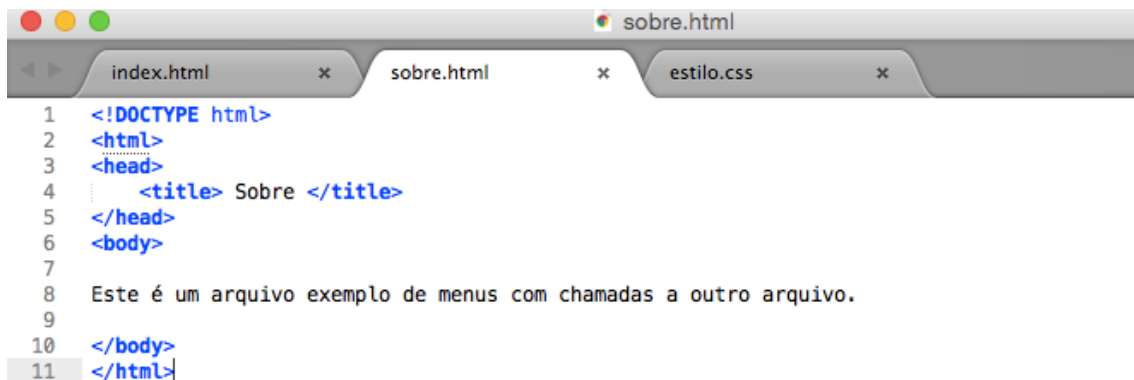
O texto muda de cor quando o mouse está sobre o item de menu

```
nav#menu a:hover{
  color: white;
}
```



Criando os destinos dos hiperlinks]

```
<div id="links">
  Links Relacionados: <br>
  <a href="http://wiki.icmc.usp.br/images/5/5e/DIP_02_Realce_p1.pdf"> Material
  Realce Imagens - ICMC - USP </a><br>
  <a href="http://wiki.icmc.usp.br/images/1/13/DIP_01_Fundamentos.pdf">
  Material de Fundamentos de Imagens digitais - ICMC - USP</a>
</div>
```



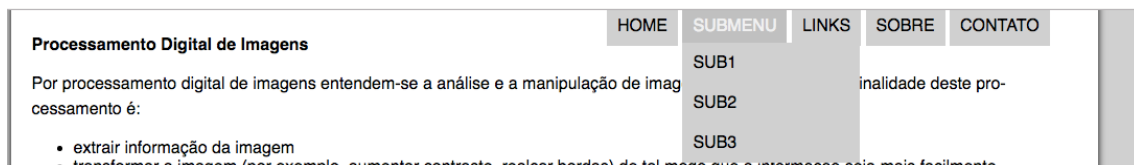
Aula 07 – submenu

Estrutura submenu

```
<nav id="menu">
  <ul class="submenu">
    <li><a href="index.html">Home</a></li>
    <li>Submneu
      <ul>
        <li><a href="sub1.html">sub1</a></li>
        <li><a href="sub2.html">sub2</a></li>
        <li><a href="sub3.html">sub3</a></li>
      </ul>
    </li>
    <li><a href="#links">Links</a></li>
    <li><a href="sobre.html">Sobre</a></li>
    <li><a href="mailto:seuemail@site.com.br">Contato</a></li>
  </ul>
</nav>
```

- [Home](#)
- Submneu
 - [sub1](#)
 - [sub2](#)
 - [sub3](#)
- [Links](#)
- [Sobre](#)
- [Contato](#)

Objetivo

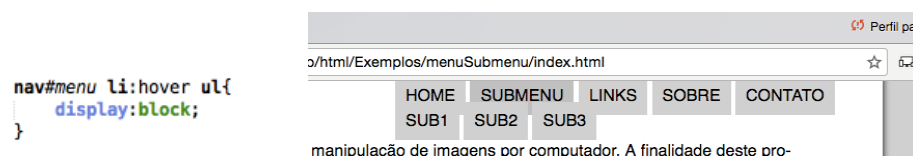


Adicionando CSS

```
nav#menu li ul{
  position: absolute;
  top: 25px;
  left: 0;
  display: none;
}
```

Ainda não acontece nada, porque definimos o display como none. Agora precisamos fazer com que apareça o menu quando o mouse estiver sobre esse item de menu

Submenu



```
nav#menu li:hover ul{
  display: block;
}
```

Por que isso acontece?

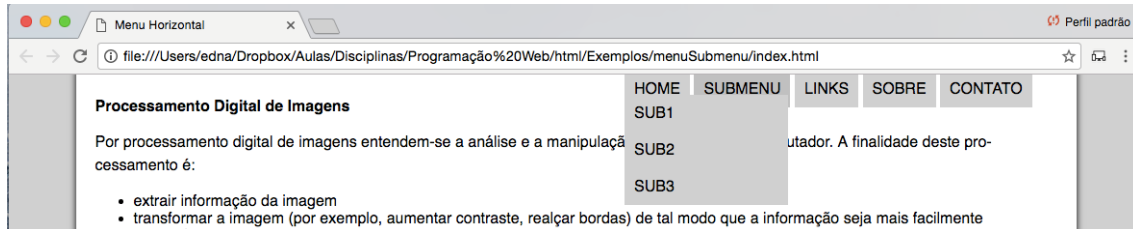
Porque quem li está programada para aparecer in-line

Definimos anteriormente:

```
nav#menu li{
  display: inline-block;
  background-color: #d0d0d0;
  padding: 10px;
}
```

Agora temos que redefinir a que está dentro de , que está dentro da submenu

```
nav#menu li ul li{
    display: block;
    width: 150px;
}
```



Mas ainda está aparecendo errado

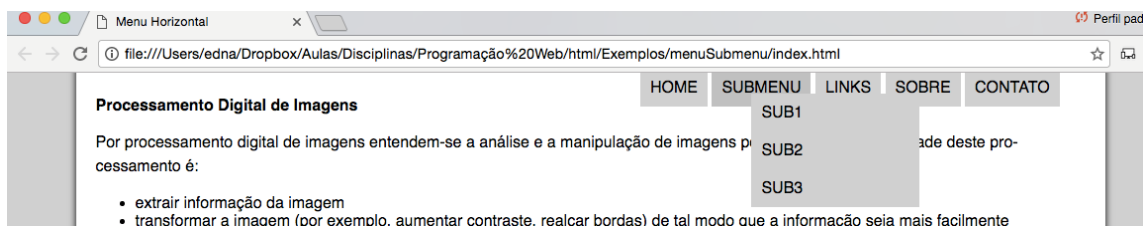
porque nav#menu li ul, está definida como absolut, temos, que acertar a

Nav#menu li,

Acrescentar:

Position: relative;

```
nav#menu li{
    display: inline-block;
    background-color: #d0d0d0;
    padding: 10px;
    position: relative;
}
```



Aula 08 – Layout

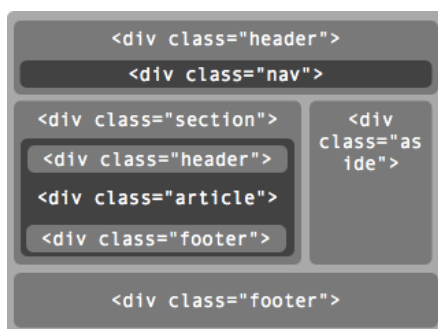
- Dentro de uma página temos, geralmente:

Cabeçalho <header> ... </header>

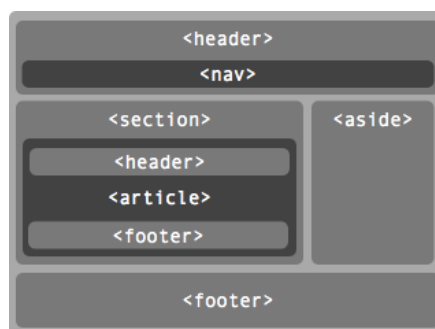
Corpo principal <section> ... </section> ;; <article> ... </article>

Rodapé <footer>...</footer>

HTML 4



HTML 5



- Podemos ter várias <section>... </section> em uma página HTML.
- Por isso, é importante, estruturar a página primeiro e depois trabalhar o layout.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Exemplo de estrutura de site em HTML5</title>
```

```
<meta charset="utf-8">
```

```
<link rel="stylesheet" href="_css/estilo.css">
```

```
</head>
```

```
<body>
```

```
<div id="site">
```

```
<header id="topo">
```

```
<h1>Titulo do site</h1>
```

```
<nav id="menu">
```

```
<ul>
```

```
<li><a href="#">Link 1</a></li>
```

```
</ul>
```

```
</nav>
```

```
</header>
```

```
<section id="conteudo">
```

```
<header>
```

```
<h2>Titulo do artigo</h2>
```

```
<small>12/12/2012</small>
```

```
</header>
```

```
<article>
```

```
</article>
```

```
<footer>
```

```
Por Fulano
```

```
</footer>
```

```
</section>
```

```
<aside>
```

```
</aside>
```

```
<footer>
```

```
</footer>
```

```
</div>
```

```
</body> </html>
```

Titulo do site

- [Link 1](#)

Titulo do artigo

12/12/2012

Por Fulano

Definições

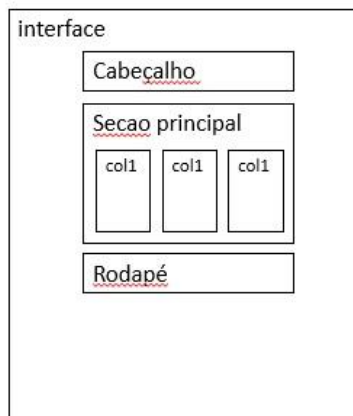
- A tag *section* define uma nova seção do site, uma área genérica do site. Por exemplo a home pode ser dividida em destaque, novidades, chamada para conteúdo completo.
- O elemento *header* representa uma área de introdução. Pode ser usado em qualquer tag, não somente na tag body
- A tag *footer* representa o rodapé do site, ou de uma seção. Pode ser usado em qualquer tag, não somente na tag body
- O elemento *nav* é utilizado para representar uma seção da página que contém links para outras partes do site. Ex. menus de navegação.
- O elemento *article* é utilizado para representar o conteúdo do site propriamente dito, como posts, artigos e outros textos em geral.

A tag *aside* representa um bloco de conteúdo que referencia o conteúdo que envolve o elemento aside.

Criando Layout:

```
<body>
|
| <div id="tela">
|
| | <header id="cabecalho">
| | | ...
| | | </header>
|
| | <section id="corpo">
| | | ...
| | | </section>
|
| | <footer id="rodape">
| | | ..
| | | </footer>
|
| </div>
</body>
```

Criando Layout

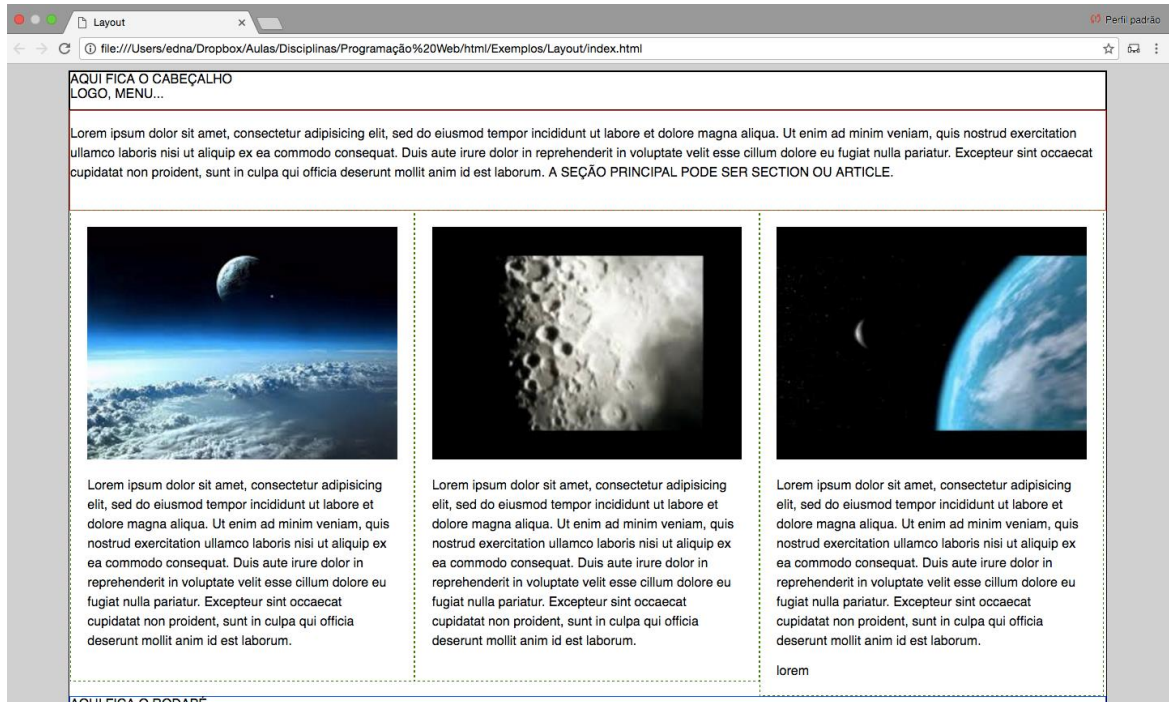


```
<body>
|
| <div id="site">
| | <header id="topo">
| | | ...
| | | </header>
|
| | <section>
| | | ...
| | | <div class="coluna">
| | | | ...
| | | | </div>
| | |
| | | <div class="coluna">
| | | | ...
| | | | </div>
| | |
| | | <div class="coluna">
| | | | ...
| | | | </div>
| | |
| | | </section>
|
| | <footer>
| | | ...
| | | </footer>
|
| </div>
</body>
```

- Para que o texto fique em colunas, temos que definir o tamanho de cada coluna e usar o atributo float.

```
.coluna{
  padding: 1.59%;
  width: 30%;
  float:left;
}
```

Resultado:



Aula 09 – JS

Javascript é uma **linguagem interpretada**, isto é, os scripts são lidos **linha a linha e cada instrução** é traduzida por um interpretador.

Como o browser interpreta o código javascript

- Se quisermos exibir um alerta, simplesmente usamos

Alert("olá Mundo");

- Onde adicionarmos esse código?? Podemos adicioná-lo no corpo do texto html.

```
1  <!-- introducao.html -->
2  <!DOCTYPE html>
3  <html lang="pt-br">
4    <head>
5      <meta charset="utf-8">
6      <title>Introdução</title>
7    </head>
8    <body>
9      <h1 id="titulo">Bem-vindo</h1>
10     alert("Bem-vindo");
11   </body>
12 </html>
13
```


- Quando visualizamos, vemos que é simplesmente exibido como um texto.
- O que significa que precisamos mostrar para o html que estamos trabalhando com javascript e isso é feito usando a tag <script> ... </script>

```

1 <!-- introducao.html -->
2 <!DOCTYPE html>
3 <html lang="pt-br">
4   <head>
5     <meta charset="utf-8">
6     <title>Introdução</title>
7   </head>
8   <body>
9     <h1 id="titulo">Bem-vindo</h1>
10    <script>
11      alert("Bem-vindo");
12    </script>
13  </body>
14 </html>

```

Javascript

- Agora será exibido



Usando variáveis

```

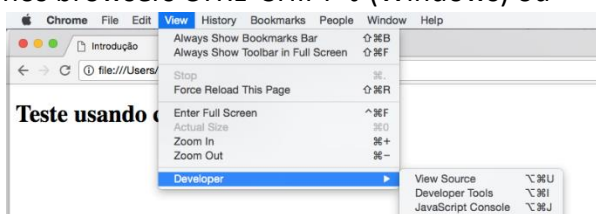
1 <!-- introducao.html -->
2 <!DOCTYPE html>
3 <html lang="pt-br">
4   <head>
5     <meta charset="utf-8">
6     <title>Introdução</title>
7   </head>
8   <body>
9     <h1 id="titulo">Bem-vindo</h1>
10    <script>
11      var nome = prompt ("Digite o nome:");
12      alert(nome);
13    </script>
14  </body>
15 </html>
16

```

Usando variáveis



- Console
 - é um comando que não aparece na tela html, mas é muito útil para os programadores, principalmente para debugar.
 - Para exibir, usamos nos browsers CTRL+SHIFT+J (Windows) ou



CMD+OPT+J (MAC).

A vantagem de usar o Console é de que o browser no modo de visualização desenvolvedor javascript ele te aponta o erro

Declarando variáveis e usando if else

- Não precisamos declarar o tipo da variável, basta declarmos, exemplo

var peso = 50;

var altura = 1.60;
- Os cálculos seguem a mesma sintaxe dos cálculos de linguagem C ou linguagem java.
- Exemplo para calcular o imc

imc = peso/(altura*altura);

Inserindo no código html

```
<!-- introducao.html -->
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="utf-8">
    <title>Introdução</title>
  </head>
  <body>
    <h1 id="titulo">Cálculo do IMC</h1>
    <script>
      var peso;
      var altura;
      peso = prompt("Digite o peso:");
      altura = prompt("Digite a altura");
      imc = peso/(altura*altura);
      console.log(imc);
    </script>
  </body>
</html>
```

Inserindo o código em arquivo separado

- A sintaxe

<script type="text/javascript" src="calcula-imc.js"></script>

- type="text/javascript" não é mais obrigatório em html5
- Veja como fica no código:

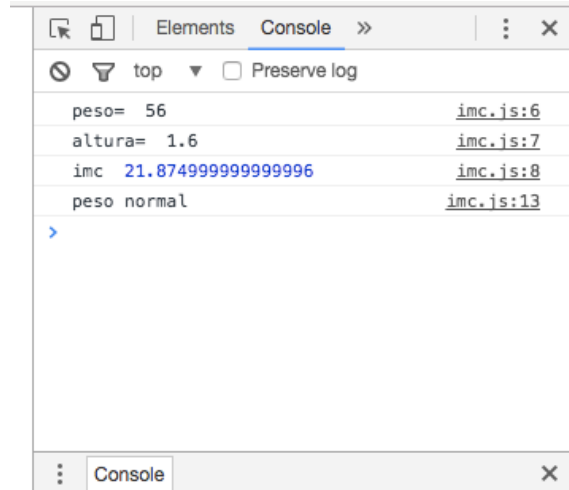
```
1 <!-- introducao.html -->
2 <!DOCTYPE html>
3 <html lang="pt-br">
4   <head>
5     <meta charset="utf-8">
6     <title>Cálculo do imc</title>
7   </head>
8   <body>
9     <h1 id="titulo">Cálculo do IMC</h1>
10    <script type="text/javascript" src="_js/imc.js"></script>
11  </body>
12 </html>
```

E o código Js

```
1 var peso;  
2 var altura;  
3 peso = prompt("Digite o peso:");  
4 altura = prompt("Digite a altura");  
5 imc = peso/(altura*altura);  
6 console.log(imc);  
-
```

Inserindo if else

```
1 var peso;  
2 var altura;  
3 peso = prompt("Digite o peso:");  
4 altura = prompt("Digite a altura");  
5 imc = peso/(altura*altura);  
6 console.log("peso= ", peso);  
7 console.log("altura= ", altura);  
8 console.log("imc ", imc);  
9 if (imc<18.5)  
10 console.log("peso abaixo do normal");  
11 else  
12 if(imc<25)  
13 console.log("peso normal");  
14 else  
15 if(imc<30)  
16 console.log("Sobrepeso");  
17 else  
18 console.log("Obeso");  
19  
20
```



Aula 10 - Vetores

Declarando vetores em Js

- Para declaramos vetores usamos

var vetor = new Array();

- Lembrando que Js é case sensitive
- Para acessar os elementos do vetor usamos

vetor[i]

- Para acrescentarmos elementos a um vetor, usamos

vetor.push(elementoAserAcrescentado);

```
1 var tam = prompt ("Quantas pessoas deseja calcular? ");  
2 var aux;  
3 var peso = new Array();  
4 var i;  
5 for (i=0;i<tam;i++){  
6     aux = prompt("Digite o peso:");  
7     peso.push(aux);  
8 }  
9  
10  
11 for(i=0;i<tam;i++){  
12     document.write(peso[i]+" ");  
13 }  
14  
15
```

O tamanho do vetor também pode ser obtido usando vetor.length.

Observe que acrescentamos o comando `document.write` para que possamos exibir texto

na página web, para facilitar a exibição de dados.

Dessa forma podemos usar o `
` para mudarmos de linha, basta usar o `+` e acrescentar

mais itens ao que será exibido.

```
1  <!-- vetor.html -->
2  <!DOCTYPE html>
3  <html lang="pt-br">
4      <head>
5          <meta charset="utf-8">
6          <title>Vetor</title>
7      </head>
8      <body>
9          <h1 id="titulo">Vetor </h1>
10         <script type="text/javascript" src="_js/vetor.js"></script>
11     </body>
12 </html>
```

- Escreva o código Javascript para calcular o IMC para n pessoas e exibir na tela do Browser o nome da pessoa, seu peso, sua altura, seu IMC e se a pessoa está abaixo do peso, peso normal, sobrepeso ou obesa.

```
<!DOCTYPE html>
<html lang="pt-br">
    <head>
        <meta charset="utf-8">
        <title>Introdução</title>
    </head>
    <body>
        <h1 id="titulo">Vetor</h1>
        <script>
            var tam = prompt ("Quantas pessoas deseja calcular?");
            var aux;
            var peso = new Array();
            var altura = new Array();
            var imc = new Array();
            var i;
            for(i=0;i<tam;i++){
                aux = prompt("Digite o peso da pessoa "+[i+1]+":");
                peso.push(aux);
                aux = prompt("Digite a altura da pessoa "+[i+1]+":");
                altura.push(aux);
                imc[i] = peso[i]/(altura[i]*altura[i])
                imc.push(aux);
            }

            for(i=0;i<tam;i++){
```

```

document.write("Peso da pessoa "+[i+1]+": <b>"+peso[i]+"</b>
</br>");
document.write("Altura da pessoa "+[i+1]+": <b>"+altura[i]+"</b>
</br>");
document.write("IMC da pessoa "+[i+1]+": <b>"+imc[i]+"</b> </br>");
if(imc[i] < 18.5 ){
    document.write("Abaixo do Peso");
}else
    if(imc[i] >= 18.6 && imc[i] < 25){
        document.write("Peso Ideal");
    }else
        if(imc[i] >= 25 && imc[i] < 30){
            document.write("Levemente acima do peso");
        }else
            if(imc[i] >= 30 && imc[i] < 35){
                document.write("Obesidade I");
            }else
                if(imc[i] >= 35 && imc[i] < 40){
                    document.write("Obesidade
II");
                }else
                    if(imc[i] >= 40){
                        document.write("Obesidade M3rbida");
                    }else{}
                document.write("</br></br>");
            }
        }
    }
}
</script>
</body>
</html>

```

Aula 11 – Eventos.

Objetos

Nessa p3gina est3o os conceitos j3 vistos e alguns conceitos novos, como o bot3o

VIDA SAUD3VEL
HOME SOBRE CONTATO

C3LCULO DE IMC

Nome	Peso	Altura	IMC	Resultado
Adamastor da Silva	89.5	1.75		
Eleanor Camila de Moraes	59.4	1.65		
Joaquim de Menezes Garcia	114.0	1.70		

Calcular IMC

Vamos aos conceitos j3 vistos:

O cabeçalho foi feito com o Header, mas dessa vez não está na div interface, observe que ela ocupa a página inteira. E "VIDA SAUDÁVEL" é uma h1.

O Menu, foi feito exatamente como antes, mas com uma pequena maleabilidade, foi usado uma % no lugar de valor fixo (ainda não trabalhamos com design responsivos – vamos deixar para Eng de Interfaces).

Não esqueça as configuração do header, body e da interface (em css).

A background do header tem a mesma cor do botão dos menu.

E a tabela ocupa 80% da página.

Na tabela foi usado th na linha de título

Foram criadas classes para peso, altura e imc, no CSS, feito alinhamento à direita

Agora vamos aos conceitos novos

- No HTML, em cada td de cada nome dar uma id, exemplo no Adamastor:
<td class= "nome" id="nome1"> Adamastor da Silva </td>
- E assim para cada um dos nomes, pesos, alturas e imcs
- As classes são para podermos configurar cada um de uma forma diferente, por exemplo: nomes são alinhados à esquerda e números à direita.
- Os id são para identificá-los no js, então teremos nome1, nome2, e assim por diante. Assim como, peso1, peso2,... , altura1, altura2,..., imc1, icm2,
- Outro item novo é o botão.

Configurando o Botão

- Outro item novo é o botão
- O botão é inserido após a tabela, com o seguinte código html.
<button id="calculoDelmc" class="botao">Calcular IMC</button>
- Com a seguinte configuração em CSS

```
button{
  color: #ffffff;
  padding: 5px 20px 5px 20px;
  border-button: 3px solid;
  font-size: 14px;
  background-color: #1780AD;
  width: 200px;
}
button:hover{
  color:#c0c0c0;
}
```

Você pode configurar o botão da maneira que quiser.

Agora já inserimos o botão em HTML, configuramos sua aparência em CSS, precisamos ainda determinar o seu comportamento em Javascript.

Para isso, primeiro vamos inserir o conceito de DOM.

DOM significa – Document Object Model

Toda página HTML tem uma hierarquia – como se fosse uma árvore hierárquica (nossos tabs).

E podemos acessar cada um desses elementos.

Vimos na aula da semana passada como acessar a página para podermos escrever nela usando document.write.

Hoje vamos buscar elementos por meio de um id, por isso id são tão importantes

Também podemos buscar elementos usando nomes de classes ou de tags.

Capturando um elemento pelo Id

Usamos o document.getElementById para resgatar o valor do elemento da página web.

No nosso caso,

```
peso=document.getElementById("peso1").textContent;
altura=document.getElementById("altura1").textContent;
```

- Lembrando-se, que no HTML, estava:

```
<tr>
  <td class="nome" id="nome1" > Adamastor da Silva </td>
  <td class="peso" id="peso1"> 89.5</td>
  <td class="altura" id="altura1">1.75</td>
  <td class="imc" id="imc1"></td>
  <td class="res" id="res1"></td>
</tr>
```

- Faz-se o cálculo do IMC, usa-se o toFixed, como já foi feito.
- E para inserir o novo valor na tabela:

```
imc1 = document.getElementById("imc1");
imc1.textContent=imc;
```

- O código js ficará

```
function calculo(){
  var peso;
  var altura;
  var imc, imc1;
  peso=document.getElementById("peso1").textContent;
  altura=document.getElementById("altura1").textContent;
  imc = peso/(altura*altura);
  imc = imc.toFixed(2);
  imc1 = document.getElementById("imc1");
  imc1.textContent=imc;
  if(imc<18.5){
    res1=document.getElementById("res1");
    res1.textContent ="Peso abaixo do normal";
  }
  else{
    if(imc<25){
      res1=document.getElementById("res1");
      res1.textContent ="Peso normal";
    }
    else{
      if(imc<30){
        res1=document.getElementById("res1");
        res1.textContent ="Sobrepeso";
      }
      else{
        res1=document.getElementById("res1");
        res1.textContent ="obeso";
      }
    }
  }
}

var botao = document.getElementById("calculoDeImc");
botao.addEventListener("click", calculo);
```

-

Observação

- É claro que o botão não é mágico, ele precisa ser ativado, quem faz isso são as duas últimas linhas.

```
var botao = document.getElementById("calculoDeImc");
botao.addEventListener("click", calculo);
```

- Lembrando de novo que no HTML deve haver um id com o nome "calculoDeImc" para o botão.

```

|
<p></p>
<button id="calculoDeImc" class="botao">Calcular IMC</button>

```

Agora vamos calcular para todos de uma vez

- Para calcularmos para todos de uma vez usamos vetores. Por isso, declaramos classes para cada um deles.
- Para obtermos vetores usamos

```

peso=document.getElementsByClassName("peso");

```

- E para obtermos o tamanho do vetor

```

n=peso.length;

<tr>
  <td class="nome" id="nome1" > Adamastor da Silva </td>
  <td class="peso" id="peso1"> 89.5</td>
  <td class="altura" id="altura1">1.75</td>
  <td class="imc" id="imc1"></td>
</tr>

<tr>
  <td class="nome" id="nome2"> Eleanor Camila de Moraes </td>
  <td class="peso" id="peso2"> 59.4</td>
  <td class="altura" id="altura2">1.65</td>
  <td class="imc" id="imc2"></td>
</tr>

```

Aula 12 – Formulário

- Queremos um formulário do tipo

NOVO PACIENTE

Nome:

Peso:

Altura:

- Para criar um formulário em html usamos as tags:

<form> </form> que delimitam o formulário

- Delimitam o conjunto de campos do formulário

<fieldset> </fieldset>

Nome do formulário (opcional) <legend> </legend>

Identificando cada campo de uma linha do formulário

```

<P>
  <label for="nome">Nome:</label>
  <input class="campo" id="campoNome" type="text" placeholder="digite o nome do novo cliente">
</P>

```


Iremos criar cada campo em uma linha:

A tag **label** indica o rótulo do campo (nome que irá aparecer para o usuário)

A tag input (não tem fechamento – pode-se usar `</>` para indicar fechamento) possui os subitens

type="text"

placeholder = "texto a ser exibido no caixa de texto"

Definimos a class para input para podermos formatar no CSS e id para podermos manipular na JS.

Fazemos isso para cada um dos itens do formulário.

Ficando o formulário dessa forma

VIDA SAUDÁVEL

HOME

SOBRE

CONTATO

CÁLCULO DE IMC

Nome	Peso	Altura	IMC	Resultado
Ademastor da Silva	89.5	1.75		
Eleanor Camila de Moraes	59.4	1.65		
Joaquim de Menezes Garcia	114.0	1.70		

Calcular IMC

NOVO CLIENTE

Nome:

Peso:

Altura:

Adicionar

Adicionando um novo paciente e Calculando o IMC

VIDA SAUDÁVEL

HOME

SOBRE

CONTATO

CÁLCULO DE IMC

Nome	Peso	Altura	IMC	Resultado
Ademastor da Silva	89.5	1.75		
Eleanor Camila de Moraes	59.4	1.65		
Joaquim de Menezes Garcia	114.0	1.70		

Calcular IMC

NOVO CLIENTE

Nome:

Peso:

Altura:

Adicionar

VIDA SAUDÁVEL

HOME

SOBRE

CONTATO

CÁLCULO DE IMC

Nome	Peso	Altura	IMC	Resultado
Ademastor da Silva	89.5	1.75		
Eleanor Camila de Moraes	59.4	1.65		
Joaquim de Menezes Garcia	114.0	1.70		
Edna	48.4	1.47		

Calcular IMC

NOVO CLIENTE

Nome:

Peso:

Altura:

Adicionar

VIDA SAUDÁVEL

HOME

SOBRE

CONTATO

CÁLCULO DE IMC

Nome	Peso	Altura	IMC	Resultado
Ademastor da Silva	89.5	1.75	29.22	Sobrepeso
Eleanor Camila de Moraes	59.4	1.65	21.62	Peso Normal
Joaquim de Menezes Garcia	114.0	1.70	39.45	Obeso
Edna	48.4	1.47	22.40	Peso Normal

Calcular IMC

NOVO CLIENTE

Nome:

Peso:

Altura:

Adicionar

É claro que não é tão simples assim, primeiro temos que criar a estrutura toda em HTML.

- Vamos lá, vocês conseguem criar com o que foi explicado??
- Usar o `<form>` e `</form>` para criar o formulário
- Usar o `<fieldset>` e `</fieldset>` dentro do form para criar os campos e ficar o retângulo delimitador.
- Dentro dos fieldset criar o `<legend>` para dar nome ao formulário

Ainda dentro do fieldset criar cada um dos campos como

- Faça isso para os demais campos: peso e altura. Colocando a mesma classe e os ids "campo-peso" e "campo-altura".

Botão

```
<button id="adicionaCliente" class="botao">Adicionar</button>
```

Falta css e js.

- Temos que tratar os dados do formulário.
- Usamos querySelector, quando não sabemos exatamente o que selecionar, para não nos preocuparmos com o que selecionar se é um id ou uma class ou tag, nesse caso estamos buscando um botão com id adiciona paciente:

```
var botaoForm = document.querySelector("#adicionaCliente");
```

-
- Depois adicionamos o evento a ser executado:

```
botaoForm.addEventListener("click", function(e){
    e.preventDefault();
```

-
- e.preventDefault(); //impede que o código seja carregado em uma nova página.

Finalizando:

- Capturamos cada uma das strings do formulários e armazenamos em uma variável (nome, peso e altura).
- Criamos a variável novoCliente como uma linha de uma tabela da classe cliente (acho que havíamos criado como paciente, mas achei que o contexto vida saudável não condizia com o tema, então troquei para cliente).
- Depois capturei a tabela e inseri o novo cliente na tabela e zerei os valores dos campos do formulário.

```
botaoForm.addEventListener("click", function(e){
    e.preventDefault();

    var nome = document.querySelector("#campoNome");
    var peso = document.querySelector("#campoPeso");
    var altura = document.querySelector("#campoAltura");

    var novoCliente = "<tr class='cliente'>" +
        "<td class='nome'>" + nome.value + "</td>" +
        "<td class='peso'>" + peso.value + "</td>" +
        "<td class='altura'>" + altura.value + "</td>" +
        "<td class='imc'></td>" + "<td class='res'></td>" +
        "</tr>";

    var cliente = document.querySelector("table");
    cliente.innerHTML += novoCliente;

    nome.value = "";
    peso.value = "";
    altura.value = "";
});
```

12-1 Forms

Para não deixar nenhum campo em branco

- Acrescentar no início do módulo
- Se um dos campos estiver vazios
 - Não deixar executar

```
var botao = document.getElementById("calculaDeIMCs");
botao.addEventListener("click", calculo);

var botaoForm = document.querySelector("#adicionaCliente");

botaoForm.addEventListener("click", function(e){
    e.preventDefault();
    var nome = document.querySelector("#campoNome");
    var peso = document.querySelector("#campoPeso");
    var altura = document.querySelector("#campoAltura");

    if (nome.value=="" || peso.value=="" || altura.value==""){
        alert("Um dos campos está vazio!!!");
    }
    else
    {
        var novoCliente = "<tr class='cliente'>" +
            "<td class='nome'>" + nome.value + "</td>" +
            "<td class='peso'>" + peso.value + "</td>" +
            "<td class='altura'>" + altura.value + "</td>" +
            "<td class='imc'></td>" + "<td class='res'></td>" +
            "</tr>";

        var cliente = document.querySelector("table");
        cliente.innerHTML += novoCliente;

        nome.value = "";
        peso.value = "";
        altura.value = "";
    }
});
```

- Trocar de text para number
- Dessa forma eles só aceitarão números.
- Não deixarão os usuários digitarem valores não numéricos.
- O tipo number pode ser utilizado com outros campos

 - Pode-se usar também o stepdown
- O tipo range
- O tipo date
- O tipo datetime
- O tipo color

IMC CODIGO

```
<!DOCTYPE html>
<html>
<head>
  <title>Calculo de IMC</title>
  <link rel="stylesheet" type="text/css" href="estilo.css">

</head>
<body>

<div class="cabecalho">
  <h1>VIDA SAUDÁVEL</h1>
  <span>
    <a href=""></a>
    <a href=""></a>
    <a href=""></a>
  </span>
</div>

  <form>
    <fieldset>
      <legend>Nova Pessoa</legend>
      <p>
        <label>Nome: </label>
        <input type="text" id="campoNome" placeholder="Digite o
Nome">
      </p>
      <p>
        <label>Peso: </label>
        <input type="text" id="campoPeso" placeholder="Digite o
Peso">
      </p>
      <p>
```

```

        <label>Altura: </label>
        <input type="text" id="campoAltura" placeholder="Digite a
Altura">
    </p>
    <button id="adicionaPessoa" class="botao"
onclick="adicionarNovaPessoa(event)" >Adicionar</button>
</fieldset>
</form>

<div class="menu">

    <table border="1" class="tabela">
        <thead>
            <th>Nome</th>
            <th>Peso</th>
            <th>Altura</th>
            <th>IMC</th>
            <th>Resultado</th>
        </thead>
        <tbody>
            <tr>
                <td class="nome" id="nome1">Adamastor da Silva</td>
                <td class="peso" id="peso1">89.5</td>
                <td class="altura" id="altura1">1.75</td>
                <td class="imc" id="imc1"></td>
                <td class="res" id="res1"></td>
            </tr>
            <tr>
                <td class="nome" id="nome2">Eleanor Camila de Moraes</td>
                <td class="peso" id="peso2">59.4</td>
                <td class="altura" id="altura2">1.65</td>
                <td class="imc" id="imc2"></td>
                <td class="res" id="res2"></td>
            </tr>
            <tr>
                <td class="nome" id="nome3">Joaquim de Menezes
Garcia</td>
                <td class="peso" id="peso3">114.0</td>
                <td class="altura" id="altura3">1.70</td>
                <td class="imc" id="imc3"></td>
                <td class="res" id="res3"></td>
            </tr>
        </tbody>
    </table>

    <button id="calculoDeImc" class="botao" onclick="calculo()">Calcular IMC</button>

    <script type="text/javascript">

        var tamanho = document.getElementsByClassName("peso").length;

```

```

function calculo(){

    var peso = Array();
    var altura = Array();
    var imc
    var imcl;
    var i = 0;
    var auxPeso;
    var auxAltura;
    var tamanho;

    peso = document.getElementsByClassName("peso");
    altura = document.getElementsByClassName("altura");

    tamanho = altura.length;

    for(i = 1; i <= tamanho; i++){
        auxPeso = peso[i-1].textContent;
        auxAltura = altura[i-1].textContent;

        imc = auxPeso/(auxAltura*auxAltura);
        imc = imc.toFixed(2);

        imcl= document.getElementById("imc"+i);
        imcl.textContent=imc;

        resl = document.getElementById("res"+i);
        if(imc<18.5){
            resl.textContent="Peso abaixo do Normal";
        }else if(imc < 25){
            resl.textContent="Peso Normal";
        }else if(imc <30){
            resl.textContent="Sobrepeso";
        }else{
            resl.textContent="Obeso";
        }
    }
}

function adicionarNovaPessoa(e){
    e.preventDefault();

    var nome = document.querySelector("#campoNome");
    var peso = document.querySelector("#campoPeso");
    var altura = document.querySelector("#campoAltura");
    tamanho ++;

    var novoCliente = "<tr>" +

```

```

nome.value + "</td>" +
"<td class='nome' id='nome'+tamanho+'>" +
peso.value + "</td>" +
"<td class='peso' id='peso'+tamanho+'>" +
altura.value + "</td>" +
"<td class='altura' id='altura'+tamanho+'>" +
"<td class='imc' id='imc'+tamanho+'></td>" +
"<td class='res'
id='res'+tamanho+'></td>" + "</tr>";

```

```

var cliente = document.querySelector("table");
cliente.innerHTML += novoCliente;

```

```

        nome.value = "";
        peso.value = "";
        altura.value = "";
    }
</script>
</div>

```

```

</body>
</html>

```

IMC CSS

```

.cabecalho{
    background: #1780AD;
    padding: 1px;
    text-align: center;
}

.menu{
    margin-left: 2%;
    margin-right: 2%;
    margin-top: 2%;
}

.tabela{
    border-collapse: collapse;
    width: 80%;
}

.nome{

}

.peso, .altura, .imc{
    text-align: right;
}







```

```
button{
  color: #ffffff;
  padding: 5px 20px 5px 20px;
  border: 3px solid;
  font-size: 14px;
  background: #1780AD;
  width: 200px;
}
```

```
button:hover{
    color: #c0c0c0;
}
```

file:///C:/Users/Leopoldo/Desktop/Aula%2012%20-%20Formulário/imc.html

☆



VIDA SAUDÁVEL

Meu

Nova Pessoa

Nome:

Peso:

Altura:

Adicionar

Nome	Peso	Altura	IMC	Resultado
Adamastor da Silva	89.5	1.75		
Eleanor Camila de Moraes	59.4	1.65		
Joaquim de Menezes Garcia	114.0	1.70		

Calcular IMC