

## CSS

### Listas

# Listas

- CSS proporciona propriedades especiais que são projetados para listas. Geralmente é mais conveniente usar estas propriedades sempre que puder.
- Para especificar o estilo para uma lista, use o list-style propriedade para especificar o tipo de marcador.
- O seletor na sua regra de CSS pode selecionar os elementos de item de lista (por exemplo, <li>), ou pode selecionar o elemento primário da lista (por exemplo, <ul>) de modo a que os elementos da lista herdam o modelo.

# Listas não Ordenadas

Em uma lista não ordenada, cada item da lista é marcado da mesma forma.

CSS tem três tipos de marcadores, e aqui está como seu navegador exibe-os:

- disc
- circle
- square

# Listas Não Ordenadas

```
<style>
  li.open {
    list-style: circle;
  }
  li.closed {
    list-style: disc;
  }
</style>
</head>
<body>

<ul>
  <li class="open"> São Paulo </ li>
  <li class="closed"> Rio de Janeiro </ li>
  <li class="closed"> Curitiba </ li>
  <li class="open"> Florianópolis </ li>
  <li class="closed"> Porto Alegre </ li>
</ ul>
```

# Listas Ordenadas

Em uma lista ordenada , cada item da lista é marcado diferentemente para mostrar a sua posição na sequência.

Use a propriedade list-style para especificar o tipo de marcador:

- decimal
- lower-roman
- upper-roman
- lower-latin
- upper-latin

# Listas Ordenadas

```
<style>
  ol.info {
    list-style: upper-latin;
  }
</style>
</head>
<body>

<ol class="info">
  <li>Flamengo</li>
  <li>Palmeiras</li>
  <li>Corinthians</li>
  <li>Vasco da Gama</li>
  <li>Grêmio</li>
</ol>
```

# Contadores

Você pode usar contadores para numerar quaisquer elementos, não somente itens da lista. Por exemplo, em alguns documentos você pode querer numerar cabeçalhos, parágrafos ou sub-títulos.

**Nota:** Alguns navegadores não suportam contadores.

# Contadores

Para especificar a numeração, você precisa de um *contador* com um nome que você especificar. Exemplo: num

Em alguns elementos antes da contagem é começar, reinicie o contador com a propriedade counter-reset e o nome do seu contador.

Em cada elemento que o contador é incrementado, use a propriedade counter-increment e o nome do seu contador.



# Contadores

Para mostrar seu contador, adicione :: before ou :: after para o selector e usar o conteúdo da propriedade.

No valor do conteúdo de propriedade, especifique counter () com o nome de seu contador.

# Contadores

```
<style>
  body {
    counter-reset:num;
  }
  p.numbered:before{
    content: counter(num) ": ";
    counter-increment: num;
    font-weight: bold;
  }
</style>
</head>
<body>

<p class = "numbered"> Introdução </ p>
<p class = "numbered"> Justificativa </ p>
<p class = "numbered"> Objetivo </ p>
<p class = "numbered"> Objetivo Específico</ p>
<p class = "numbered"> Metodologia </ p>
```

# Menu

CSS possibilita definir uma variedade infinita de layouts e efeitos para um menu de navegação. O limite é a sua imaginação!

A tag `<nav>` define um conjunto de links de navegação.

Primeiramente faça um elemento NAV (antigamente usávamos a tag DIV) e atribua um ID. Neste exemplo nossa NAV se chamará “menu”. Dentro desta NAV, faça uma lista com as opções do menu.

# Menu

```
<nav id="menu">
  <ul>
    <li><a href="#">Home</a></li>
    <li><a href="#">Produtos</a></li>
    <li><a href="#">Missão</a></li>
    <li><a href="#">Links</a></li>
    <li><a href="#">Contato</a></li>
  </ul>
</nav>
```

# Menu

- Agora que já fizemos a estrutura do menu, vamos formatá-lo com CSS.

```
<style>
    #menu ul {
        padding:0px;
        margin:0px;
        background-color:#EDED;
        list-style:none;
    }
</style>
```

# Menu

- Para o menu ficar horizontal, temos que fazer as suas opções ficarem uma ao lado da outra, para isso, basta atribuir um `display:inline;` para a tag `li` isso fará todas as opções ficarem em uma linha horizontal:

```
#menu ul li { display: inline; }
```

# Menu

Menu horizontal definindo um novo “visual” que deverá ter o link: cor de fundo, letra, etc.

```
#menu ul li a {  
    padding: 2px 10px;  
    display: inline-block;  
  
    /* visual do link */  
    background-color: #EDED;   
    color: #333;  
    text-decoration: none;  
    border-bottom: 3px solid #EDED;  
}
```

# Construção Layout

Primeiramente, vamos iniciar com os estilos que reconfiguram definições padrão dos navegadores. Cada navegador possui sua própria folha de estilos com suas especificações de fonte, margens, bordas e outros. Ver [reset.css](#)

```
/*
-- Padrões e Reset
*/

* {margin: 0; padding: 0;}
ul {list-style-type: none;}
img, fieldset {border: 0;}
h1, h2, h3, h4, h5, h6 {font-weight: normal;}
```



# Construção Layout

Definir a largura do site para que ele não ocupe sempre toda a largura da tela. Apesar da ideia de flexibilidade ser interessante, deixar o site ocupar toda a largura da tela sempre pode causar muitos problemas de layout e legibilidade, como, por exemplo, linhas muito longas. Por isso, vamos limitar a largura do site a 960 pixels, conforme o layout:

```
/*  
-- Estrutura  
*/  
  
#cabecalho, #conteudo, #rodape {  
    width: 960px;  
    margin: 0 auto;  
}
```

# Construção Layout

No rodapé fazemos o fundo desta ocupar a largura da página  
Simples: no código HTML, adicionamos uma div extra para conter este fundo:

```
<div id="rodape-fora">  
  <div id="rodape">  
    [...]  
  </div>  
</div>
```

No código CSS, adicionamos a cor de fundo na div que acabou de ser criada:

```
/*  
-- Estrutura  
*/  
[...]  
#rodape-fora {  
  background: #808080;  
}
```

# Construção Layout

A primeira coisa que iremos fazer é deixar o cabeçalho com posicionamento relativo, assim criando um contexto de posicionamento na div, como já vimos. Isso irá possibilitar o posicionamento do menu na direita com precisão:

```
/*  
-- Estrutura  
*/  
[...]  
#cabecalho {  
    position: relative;  
}
```

# Construção Layout

Além disso, iremos flutuar a logomarca para a esquerda, para que a logo, sendo bloco, não ocupe toda a largura da div "cabecalho". Se deixarmos a logo com `display: block;` e não limitarmos o seu tamanho, o link irá tomar conta de todo o espaço horizontal do cabeçalho e pode acabar interferindo com a navegação. Uma das formas de limitar a largura de um elemento bloco é flutuando-o, assim este encolhe automaticamente para se adequar ao tamanho do seu conteúdo interno.

```
[...]  
#logo {  
    padding: 10px 0;  
    display: block;  
    float: left;  
}  
  
#logo img {  
    display: block;  
}
```

# Construção Layout

Flutuar um elemento traz o problema e floats: a div "cabecalho" não irá reconhecer o tamanho da logomarca caso a flutuação não seja finalizada. Como temos a logomarca flutuada e o menu será posicionado, o cabeçalho irá sumir a não ser que façamos algo.

```
/*
-- Classes de Uso Geral
*/

| .grupo::before, .grupo::after {
    content: " ";
    display: block;
}

| .grupo::after {
    clear: both;
}
```

# Construção Layout

Vamos usar `position` para colocar o menu à direita do cabeçalho. Estou aqui usando uma medida exata em pixels, mas podemos usar outras unidades, como **em** ou %.

```
/*
-- Estrutura
*/
[...]
#menu-principal {
    position: absolute;
    top: 26px;
    right: 0;
}
```

# Construção Layout

Agora vamos estilizar os itens de lista e os links, conforme vimos anteriormente:

```
/*
-- Estrutura
*/
[...]
#menu-principal li {
    display: inline;
}

#menu-principal a {
    display: block;
    float: left;
    font-size: 18px;
    text-decoration: none;
    color: #333333;
    margin-left: 32px;
}
```

# Construção Layout

Com isso, as regras de posicionamento do menu estão feitas. No entanto, ainda falta um pouco de *feedback* para o usuário: quando passamos o mouse em cima de um dos itens, nada acontece (além da mudança do cursor do mouse) e o item atual do menu não está destacado (o que ajuda o usuário a se localizar no site).

```
[...]
#menu-principal a:hover, #menu-principal a.ativo {
    color: #0f599f;
}

#menu-principal a.ativo {
    font-weight: bold;
}
```



# ATIVIDADE

Crie as outras páginas para o site do IFSUL, obedecendo o mesmo padrão de estilização:

- missao.html
- campus.html
- cursos.html
- contato.html

Lembrando que a página de contato deve possuir um formulário.