

CSS

Cascade Style Sheets (Folha de Estilos em Cascata)

- “Efeito” **Cascata**: É por conta da hierarquia de edição.
 - Exemplo: No CSS, se configurar uma tag <p> pra cor vermelha, todas as tags <p> são formatadas para vermelho simultaneamente, porém, consigo configurar cada uma individualmente também.
- **Mais de 1 arquivo**: Tem como ter vários arquivos CSS, porém, o arquivo base se chama *style.css*, nele, estão configuradas o header, footer, nav (formatações gerais), que são imodificáveis no site.
- **No HTML**: A tag <link rel=”stylesheet” href=style.css /> faz o arquivo buscar o CSS no arquivo de HTML.
 - Rel: função do arquivo que estamos importando, é o link externo.
 - Href: caminho do arquivo, a pasta onde ele está (navega colocando ctrl + espaço). Se o href atributo não estiver presente, a tag <a> não será um hyperlink.
- **Stylesheet**: faz com que o seu site apareça nas primeiras páginas do Google pois está Semanticamente correto.
 - Detalhe: É *extremamente importante* que no HTML5 as semânticas estejam corretas, se não o seu site vai ser rankeado bem no fim das pesquisas (stylesheet).
- **Pré-configurações de margem**: Algumas tags tem formatação padrão, como h1, h2, h3. Elas têm pré-configurações de margem.
- **CSS reset**: Para configurar essas bordas é massante, porém, devemos usar o CSS reset, onde se tira as configurações padrão.
 - Consigo o CSS reset com Chat GPT e outros sites.
 - Há diferenças entre alguns CSS reset, alguns são mais completos que outros, sendo usados para tags mais específicas.
- **Font-size (em)**: Para tamanho de letras devemos usar a unidade de medida “em”, pois ele não se desconfigura com a pixelagem (px) dos equipamentos.
 - O CSS reset deve ter o padrão de 1em, pois ele é o parâmetro, então, podemos alterar as demais escritas.
 - Se eu gostaria que uma determinada escrita fosse 50% maior, por exemplo, eu aumento o tamanho para 1.5 (É IMPORTANTE USAR O “ . ” (ponto) E NÃO “ , ” (vírgula)).

- Quando eu uso #, eu estou configurando um elemento com id.
- **Text-align** só serve para configurar o alinhamento de textos.
- **Color:** As cores devem estar configuradas em hexadecimal, pois, cores são subjetivas a cada ser infeliz.
 - O sistema hexadecimal vai de 0 a 9 e A a F.
 - O QUE SIGNIFICA ESSA MISÉRA? Significa que, no RGB (**red**, **green**, **blue**), os dois primeiros caracteres de um número hexadecimal significam a quantidade de **vermelho**, as duas seguintes a quantidade de **verde**, e as duas últimas, a quantidade de **azul**.

Exemplo:

Vermelho: #ff0000

Verde: #00ff00

Azul: #0000ff

Preto (ausência de todas as cores): #000000

Branco (presença de todas as cores): #ffffff

- **Asterisco** significa que estou selecionando tudo.
- **Pseudo-classes:** Com “--” eu declaro uma cor com o nome que eu quiser usando a pseudo-classe *:root*. Exemplo: o nome da nossa variável vai ser “rosaclaro”.

```
:root{
  --rosaclaro: #facada
}
```

```
em outra tag {
  color: --rosaclaro
}
```

- **Mudando cores com a interação:** Quando temos um link, ele é azul inicialmente. Podemos trocar a cor do URL com as seguintes pseudoclasses:
 - *:hover* (Quando passamos o mouse).
 - *:active* (Quando é clicado).
 - *:visited* (Quando já visitei o link).

Exemplo:

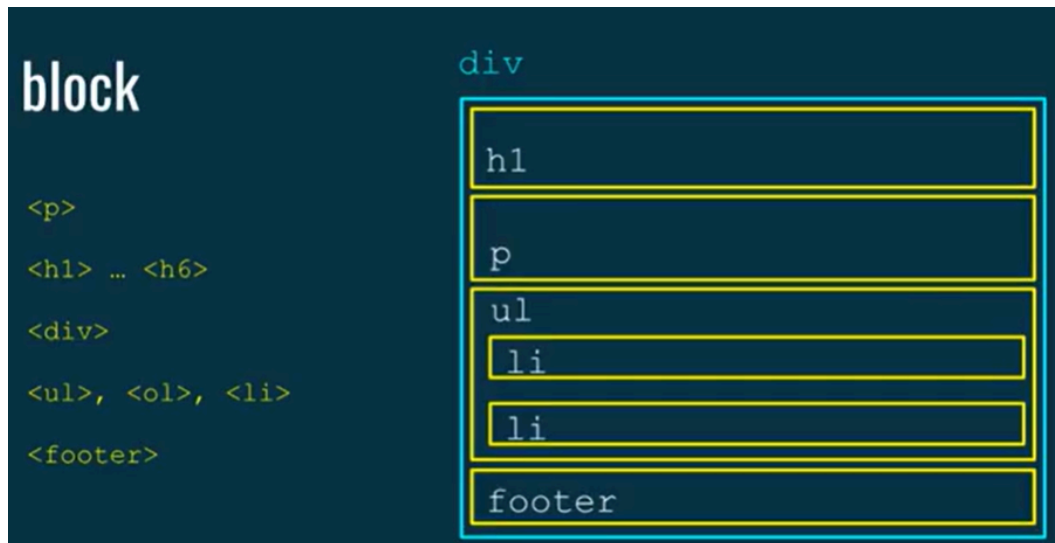
```
a:visited {
  color: #ff0000
}
```

```
a:hover {  
color: #??????  
}
```

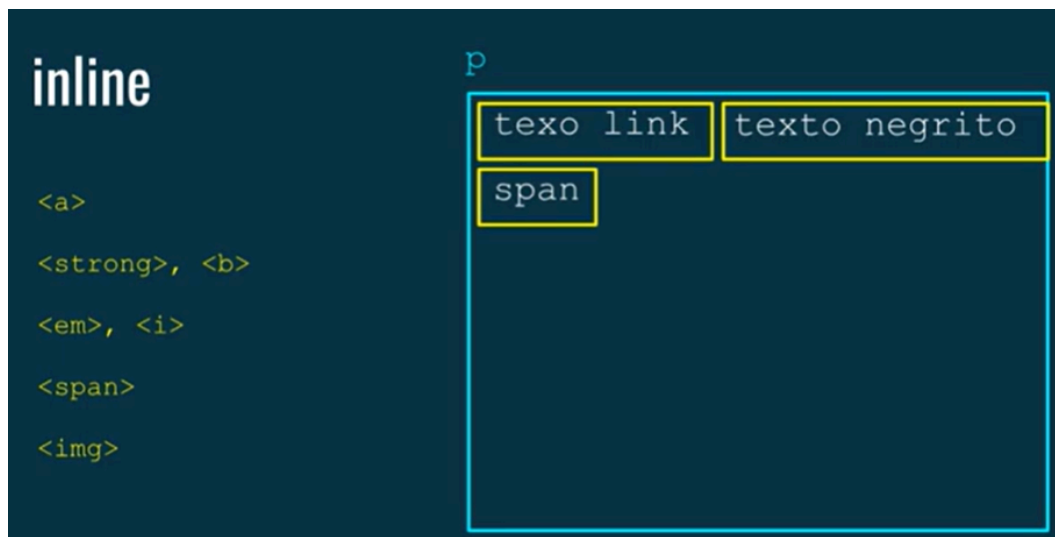
```
a:active {  
color: #??????  
}
```

- **<div>** é uma caixa clicável no site. No css, vamos configurar com width e height.
 - **Background color:** cor de fundo.
 - **Border:** é a borda visível, a moldura do site, seja ela inteira, à esquerda, à direita, em cima ou em baixo.
 - **Margin:** margem ao redor do botão clicável
 - **Padding:** distância entre o conteúdo e a borda
- Para editar o tamanho, cor ou tipo de cada “canto” desse elementos da caixa:
 - 1 valor= dá pra editar todos os lados
 - 2 valores= cima e baixo
 - 4 valores= esquerda e direita
- Para direcionar essa caixa a algum outro link, precisamos colocar **<div>** dentro de **<a>**.
`<div>CLIQUE AQUI</div>`
- **id:** Configura apenas *um* elemento. Exemplo: `<div id= "nome">`
- **class:** Configura *mais de um*. Exemplo: `<li class="itemMenu">`
- **Configurando uma div com um id:**
 - No html:
Vai em uma tag `<div>` e coloca `id="qualquerNome"` (`<div id="qualquerNome">`)
 - No css:
Após configurar no HTML, vá para o css e coloque o nome da id configure o que você quiser.
`qualquerNome{
color: #??????
border: ??px
}`
- **Display:** especifica como um elemento é mostrado numa página. Cada elemento tem um valor de exibição padrão, geralmente sendo:

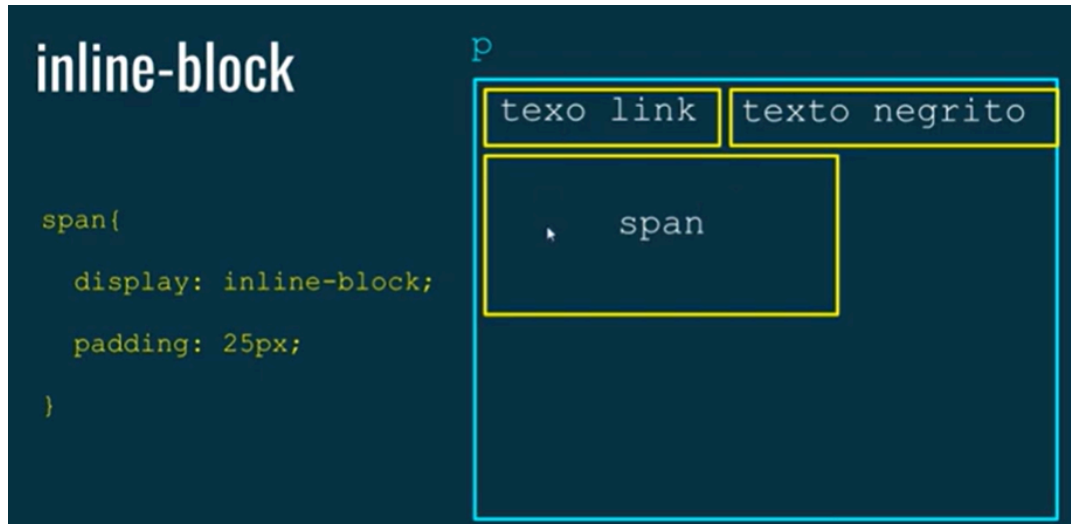
- **block:** ocupa a largura(width) de toda a página e gera quebra de linha pro elemento que usarmos em seguida.



- **inline:** elementos que ocupam apenas a área necessária do seu próprio conteúdo, pode ir aumentando conforme formos colocando mais coisas.



- **inline-block:** podemos alterar a largura(width) de um elemento específico sem alterar outros, basta colocar o nome do elemento desejado no css e alterar como preferir.



- **none:** esconde elementos da página web sem excluir do código. Exemplo: menus escondidos, só aparecem se interagirmos com certo botão/elemento do site.

