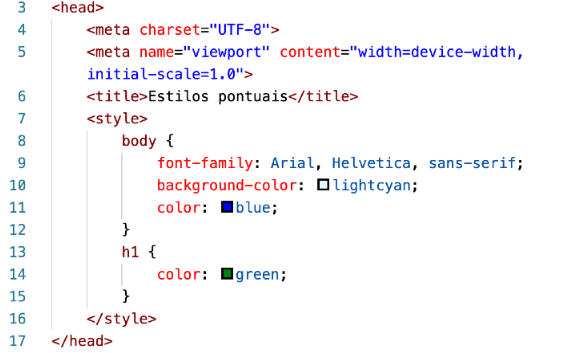
Tags usadas no HTML:

* <title> Título em cima do site </title>
* <h1> Título principal</h1> (assunto principal)
* <h2> Título secundário </h2> (sub-assunto)
* <p> parágrafo </p> (para a criação de um  
  parágrafo simples).
* <a> ancora </a> (cria-se um hiperligação nas páginas web, arquivos, endereços de emails, ligações na mesma página ou endereços na URL).
* <a target> ancora com target </a> (‣ blank vai abrir o link em uma nova janela em branco  
  ‣ \_self vai abrir o link na janela ou frame atual (padrão).  
  ‣ \_top vai desfazer todos os frames e abrir o destino no navegador completo  
  ‣ \_parent similar ao uso do \_top em uma referência à janela mãe  
  ‣ nome-do-frame caso esteja usando frames, indicar o nome da janela a abrir)

<a rel> ancora com rel </a> (Existe um recurso bem interessante para links que é indicar qual é a natureza do destino usando o atributo rel. Esse atributo aceita vários)  
valores, entre eles vou citar:  
‣next indica que o link é para a próxima parte do documento atual  
‣prev indica que o link é para a parte anterior do documento atual  
‣author indica que é um link para o site do autor do artigo atual  
‣external indica que é um link para outro site que não faz parte do site.

atual  
‣ nofollow indica que é um link para um site não endossado, como um link pago).

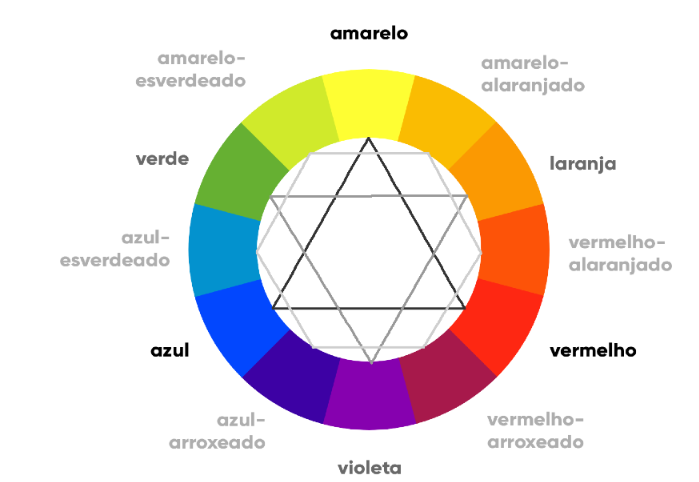
* <img> imagem (responsável por carregar imagens que estão na pasta do projeto atual ou em links externos).
* <br> quebre a linha (A única quebra que  
  REALMENTE aconteceu foi após a palavra “específico” no segundo parágrafo).
* <strong> negrito </strong> (Significa que o termo delimitado possui força dentro da frase. Logo, ele aparecerá em negrito).
* <em> ênfase </em> ênfase (significa que queremos dar ênfase [do Inglês emphasis] ao termo. Logo, ele aparecerá em itálico).
* <mark> marca texto </mark> (para delimitar o texto que queremos demarcar, como se estivéssemos usando uma caneta marcador).
* <small> texto menor </small> (que deixa o texto menor> segue firme e forte na vida da linguagem).
* <q>citar</q> (já vai receber automaticamente as aspas, mas não terá nenhum deslocamento).
* <blockquote>Citação mais longa</blockquote> (Também podemos criar citações mais longas (em bloco) e que tenham um parágrafo só para si. Nesse caso, colocaremos tudo dentro de <blockquote> e </blockquote> o texto ganha um recuo automaticamente. Podemos também colocar um link para o texto original, usando o parâmetro cite dentro da tag).
* <abbr>Abreviações</abbr> (Sempre que você quiser escrever uma sigla, mas deixar claro ao usuário (e aos mecanismos de busca) o significado dela use a tag <abbr>).
* <bdo> ODITREVNI OTXET (Para começo de conversa, BDO significa bi-directional override. Ao usar essa tag, coloque também o parâmetro dir para indicar uma das duas direções possíveis: rtl = da direita para a esquerda (right-to-left)  
  ltr = da esquerda para a direita (left-to-right) ).
* <ol> <li> Listas Ordenadas </li></ol> (A HTML chama de ordered lists todas aquelas listas onde a ordem dos itens é algo muito importante).
* <ul><li>Listas não Ordenadas</ul></li> (são aquelas onde a ordem dos itens não influenciará no significado da lista
* <dl></dl> <dt> <dd>Lista de Definições. (Toda lista de definições está dentro de uma tag <dl> </dl> (definition list) Cada termo é um <dt> (definition term) e cada descrição é um <dd> (definition description). Assim como os itens da lista, essas duas últimas tags possuem fechamento opcional, segundo a referência oficial da HTML5).
* <pictures> foto </pictures> (gerar imagens de tamanho diferentes e a fazer o navegador carregar a imagem certa para cada situação. Para isso, devemos conhecer as tags <picture> e <source>
* <audio> arquivo de áudio </áudio> (A partir de agora, basta uma tag <audio> e alguns <source> para fazer seu site ser capaz de tocar qualquer áudio).
  + Vamos analisar os principais atributos da tag <audio> antes de mais nada:
  + metadata vai carregar apenas as informações sobre o arquivo (tamanho, tempo, informações de direitos, etc)
  + none não vai carregar absolutamente nada até que o usuário clique no botão play ou um script inicie a reprodução
  + auto (padrão) vai carregar o arquivo de áudio inteiro assim que a página for carregada, mesmo que o usuário nunca aperte o play
    - O atributo controls vai apresentar o player na tela. Caso não seja colocado na tag <audio>, o controle será transparente e o usuário não poderá interagir com ele.
    - O atributo autoplay, quando inserido, vai iniciar a reprodução do áudio assim que a página for carregada.
    - O atributo loop vai fazer com que o áudio seja repetido eternamente assim que terminar a sua reprodução
  + Css Inline (estilo é feito na mesma linha que o css)
    - <body style=” background-color; font-family; font- size”>
    - <h1 style= “color; background color; font-size”></h1>
    - <h2 style=color; font-size></h2>
    - <p style “ text-align: justify”> </p>
  + Css Interno (Para aplicar estilos de forma mais dinâmica e prática, podemos adicionar uma tag <style> dentro da área <head> do nosso documento HTML local. Volte lá no seu VSCode, e adicione o código dentro de <head>



* CSS external style
  + Manter as folhas de estilo fora do código HTML, além de uma maior organização faz com que tudo seja reaproveitado de maneira mais eficiente nas outras páginas do nosso site. Para isso, utilizamos a tag <link> especialmente configurada para trabalhar com arquivos externos de estilo. Essa tag deve ser colocada dentro da área <head> do seu documento HTML



* Cores
  + Você provavelmente já olhou para um belo site ou para uma peça de propaganda bem produzida, teve aquela sensação de que tudo está em perfeita harmonia, mas não sabe explicar o porquê do seu cérebro perceber toda essa beleza e te fazer se sentir bem.
  + Pois saiba que boa parte de toda essa percepção que temos é por conta das cores e da simetria geométrica que aconteceu durante o planejamento desse site/propaganda.



* + Cores primárias: estão destacadas com o texto mais escuro: amarelo, vermelho e azul.
  + Cores secundárias: que são o laranja (amarelo+vermelho), o violeta/roxo (azul+vermelho) e o verde (azul+amarelo)
  + Cores terciárias: Da junção de uma cor primária com uma secundária
  + Amarelo-esverdeado (amarelo+verde)  
    Amarelo-alaranjado (amarelo+laranja)  
    Vermelho-alaranjado (vermelho+laranja)  
    Vermelho-arroxeado (vermelho+roxo)  
    Azul-arroxeado (azul+roxo)  
    Azul-esverdeado (azul+verde)
* Temperatura e Harmonia
  + Olhando o círculo cromático, também conseguimos classificar as cores por sua temperatura. Dá só uma olhada na imagem a seguir:

  
Além da classificação por temperatura, podemos classificar as cores por esquemas harmônicos.

1. Cores complementares:



São aquelas que apresentam o maior contraste entre si. Elas estão localizadas do lado imediatamente oposto do círculo cromático.

Se pegarmos qualquer cor primária, a sua cor complementar é  
sempre uma cor secundária. De forma similar, qualquer cor terciária tem uma outra cor terciária como complementar.  
Quando juntamos duas cores complementares, sempre obtemos o cinza.

1. Cores análogas:



Diferente das cores complementares, que estão do lado  
extremo oposto no círculo cromático, as cores análogas são  
aquelas que são imediatamente vizinhas entre si.

Por serem cores consecutivas, as cores análogas possuem um baixo contraste entre elas, mas criam uma bela harmonia quando combinadas em um mesmo design.

1. Cores análogas mais uma complementar



Dá pra notar que essa aqui é uma combinação dos dois tipos anteriores, não é?

Essa técnica quebra um pouco o ritmo semelhante das cores análogas, adicionando uma cor que cria um grande contraste com as três análogas.

1. Cores análogas relacionadas



Nesse tipo de harmonia, escolhemos duas cores análogas (consecutivas) e depois pulamos uma terceira cor (em qualquer direção) e escolhemos a quarta.

Com essa técnica, conseguimos um resultado parecido com o  
das cores análogas simples, mas com um pouco mais de contraste sem ter que escolher uma cor complementar.

1. Cores intercaladas



Um tipo menos usado de harmonia, já que às vezes não funciona tão bem assim. Vamos escolher a primeira cor e depois mais duas com intervalo constante entre elas.

Na imagem ao lado, criei um exemplo onde o intervalo é constante entre as cores selecionadas.

1. Cores triádicas



Técnica bastante utilizada e que garante uma grande riqueza de cores, onde escolhemos três pontos equidistantes no círculo cromático.

Esse esquema gera sempre um triângulo equilátero e cria uma opção que sempre possui um ótimo contraste entre as cores.

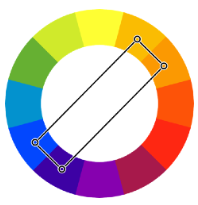
1. Cores em quadrado



Bastante semelhante ao esquema triádico, mas permite selecionar quatro cores com um contraste razoável entre as cores escolhidas.

Esse esquema gera sempre um quadrado perfeito com os pontos selecionados.

1. Cores tetrádicas



Com essa técnica, vamos escolher dois pares de cores complementares, que não serão necessariamente análogas ou consecutivas. Isso vai nos garantir dois pares de cores, com bastante contraste entre si.

1. Monocromia



Uma harmonia bem diferente das anteriores, que usa apenas uma cor e varia apenas a sua saturação e o seu brilho. Essa combinação geralmente gera pouquíssimo contraste entre as cores escolhidas, mas acaba gerando um resultado visual bem agradável aos olhos, conhecido como “degradê”.

**Usando Gradientes em CSS**

Podemos gerar gradientes e aplicarmos a componentes visuais usando folhas de estilo. Vamos usar um exemplo simples no nosso exercício atual. Vá até o documento e modifique a declaração do nosso seletor body.



Pode parecer esquisito no início, mas uma gradiente é considerada pelo navegador como se fosse uma imagem, por isso usamos a propriedade background-image na declaração CSS. A função linear-gradient é auto-explicativa e gera um gradiente  
linear angular. O primeiro parâmetro da função indica o ângulo de inclinação de 90 graus (90deg) e as seguintes indicam as cores do degradê a ser criado. Você pode indicar quantas cores quiser e o navegador vai saber se virar pra gerar seu degradê personalizado. Experimente na sua casa outros valores de ângulo também, incluindo negativos (45deg, -90deg, 25deg,...) e note as diferenças.

Também é possível gerar os chamados gradientes radiais, que também são meio auto-explicativos. Veja o exemplo:



Altere o tipo de gradiente do body para usar o formato radial circular e veja o resultado. Você também pode personalizar ainda mais seu degradê colocando uma porcentagem ao lado da cor como red 10%, yellow 40%, green 50%. Experimente!

Tipografia?

Para entender mais sobre as fontes, precisamos estudar os fundamentos básicos da tipografia, que é uma arte antiga que estuda técnicas de escrita (do Grego, graphía) para a apresentação de forma impressa (do Grego, týpos).

Fontes de tipográfica:

Nem toda a fonte tem família tipográfica gigante, as vezes tem um tipo de fonte, mas as vezes tem família tipográfica. Nem toda fonte tem (Light, extranegrito) e toda fonte tem normal e extranegrito

Vamos falar de tamanhos

Além da família, podemos configurar tamanhos e estilos extras de qualquer componente textual do nosso documento HTML5.

Medidas Absolutas: cm (centímetros), in (polegadas), pt (pontos), pc (paicas), px (pixels)

Medidas relativas: Em (Altura M), ex (altura x), rem (root em) vw (viewport), vh( altura da viewport), % (porcentagem)

16 px= geralmente 1em

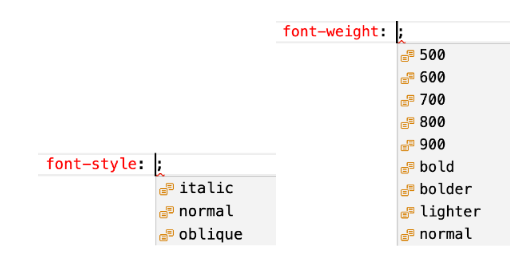
Recomendação oficial: Use px e use em

Outros estilos

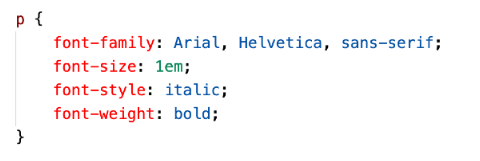
Weight: peso

Wigth: largura

Height: altura



Me dá uma mãozinha ✋?



Podemos usar a shorthand font que vai simplificar tudo:



Tipos de formatos ()

* opentype(otf)
* truetype(ttf)
* embedded-opentype
* truetype-aat (apple- advanced typography)
* svg