**Introdução a criação de websites com HTML5 e CSS3**

**CSS3**

**Objetivos**

1. O que são seletores
2. Conceitos básicos
3. Principais seletores

**Requisitos básicos**

* Editor de texto >> VSCODE
* Navegador de internet >> Chrome, **Firefox**, Edge etc.
* Ter assistido as aulas de introdução ao HTML 5

**Definição e seletores**

Após a criação do HTML a necessidade de formatar as páginas ficou evidente, assim, em 1996, foi criada a linguagem de estilo que conhecemos por CSS.

A sintaxe é bem simples e pode ser explicada com a frase "você cria regras de estilo para elementos ou grupos de elementos".

Vamos usar um elemento HTML que vimos anteriormente, a âncora <a>, para exemplificar.

Uma regra CSS é representada por um seletor ou um grupo de seletores, no nosso caso é o <a>, então dentro de um par de chaves adicionamos as declarações, no exemplo acima estamos alterando cor e tamanho da fonte dessa âncora, as declarações são formadas por uma propriedade e um valor.

Percebam que podemos colocar vários seletores em uma regra separando-os por vírgula.

Texto

Descrição gerada automaticamente

E há um último detalhe nesse exemplo: a pseudo-classe. Elementos HTML sofrem alterações causadas pela interação do usuário, como mover o mouse por cima ou clicar nesse elemento.

O a:hover do exemplo significa que a âncora também terá essa aparência quando o usuário passar o mouse por cima de um hyperlink.

**ID x Classe**

No exemplo anterior criamos uma regra que altera um elemento HTML diretamente, mas isso significa que todos os elementos <a> ficarão com aquela aparência, e normalmente temos sites mais complexos que precisam de várias regras diferentes para elementos iguais.

Para ficar mais tangível vamos relembrar um pouco o site que começamos a fazer no módulo passado, ele tinha vários elementos header, mas não vamos querer que o header principal tenha a mesma formatação que o header de uma postagem, é aí que entram os IDs e Classes.

Texto

Descrição gerada automaticamente

O seletor que vimos no primeiro exemplo é um seletor de tipo, pois ele representa um elemento HTML, e com IDs e Classes podemos representar qualquer tipo de elemento, mas há algumas diferenças entre eles:

ID: é representado pelo símbolo # (hash) seguido de um nome para esse ID.

Classe: a classe é representada de forma parecida do ID, mas é precedida por um ponto em vez do hash.

Texto

Descrição gerada automaticamente

E a diferença mais importante entre eles é a forma como devem ser usados: o ID só pode ser usado uma vez em uma página HTML enquanto a classe não tem restrições.

**Exercício prático**

Vamos adicionar algumas classes no nosso site e alterar alguns elementos, mas antes precisamos adicionar um arquivo CSS a nossa página.

No módulo de HTML descobrimos que podemos adicionar CSS de duas formas, com o elemento style, e assim suas regras ficarão no arquivo HTML, ou podemos criar um arquivo CSS e adicioná-lo na página através do elemento link, e é essa forma que usaremos.

Crie um elemento link dentro do head do seu arquivo e adicione os atributos rel="stylesheet" e href="style.css", o rel denota o tipo de arquivo que estamos incluindo na página e o href é o caminho para o arquivo. E na mesma pasta do arquivo HTML crie um arquivo chamado style.css.

Agora sim vamos ao CSS, adicione um ID #title ao h1 da página, pois queremos que ele seja único, e depois adicione as classes .subtitle e .post\_title ao h2 e h3, respectivamente.

No arquivo CSS vamos mudar a cor desses três títulos, e depois alterar o tamanho da fonte do título da postagem.

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

**Box-model**

Quando estamos criando o layout de um site o navegador representa cada elemento HTML como uma caixa retangular, isso é o box-model. E com CSS nós alteramos a aparência dessa caixa (largura, altura, cor de fundo, etc.). Essa caixa é composta por 4 áreas: o conteúdo, o padding, a borda e a margem.

Diagrama

Descrição gerada automaticamente com confiança média

* As margens (margin) são espaçamentos entre elementos;
* As bordas (border) ;
* O padding é um espaçamento entre as bordas e o conteúdo, a diferença para as margens é que declarações de imagem de fundo funcionam nele;
* O conteúdo (content) é o que o seu bloco representa, um texto, uma imagem, um vídeo;

**Exercício prático**

Para enxergamos o box-model vamos adicionar cores e bordas a alguns elementos.

Primeiro adicionaremos uma cor de fundo para a visualização ficar mais fácil, usaremos a propriedade background com o valor #fcfcfc no elemento body.

Depois vamos adicionar uma classe ao <article>, pode ser .post, e então vamos colocar a cor branca de fundo com a propriedade background e o valor #FFF. Agora conseguimos enxergar o content do box-model.

Vamos adicionar um padding de 10 pixels neste mesmo article. Perceberam o espaçamento que surgiu em volta do nosso conteúdo?

Agora adicionamos um borda mais escura a ele com a propriedade border. Vou falar mais detalhadamente sobre border mais a frente, mas por enquanto vamos deixar essa borda com 3 pixels de largura, o contorno sólido e a cor azul.

E por último vamos adicionar uma margem do lado de fora do post com a propriedade margin e o valor 10 pixels.

E agora inspecionando o nosso elemento conseguimos todas aquelas camadas citadas antes: o conteúdo em azul, o padding em verde, as bordas em marrom e as margens em laranja.

E já que começamos a falar sobre bordas e cor de fundo, no próximo vídeo vamos nos aprofundar nessas propriedades.

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

**Estilizando elementos**

Agora que entendemos o box-model podemos focar em deixar nosso site mais bonito, então vamos repassar pelas propriedades já citadas:

**Padding e Margin**

Anteriormente usamos o padding e o margin da forma mais básica, com apenas um valor, mas eles são mais poderosos que isso. Se quisermos atribuir tamanhos diferentes para cada lado do box nós podemos, e vamos ver três formas de fazer isso.

Texto

Descrição gerada automaticamente

A primeira é colocando um valor para as partes superior e inferior e depois para os lados esquerdo e direito.

Texto

Descrição gerada automaticamente

O valor de 10 pixels se refere ao eixo Y, ou partes superior e inferior, e os 5 pixels se referem aos lados esquerdo e direito.

A segunda forma é dando valores para cada lado do box.

Texto

Descrição gerada automaticamente

Então começamos pelo topo com 15 pixels, passamos o lado direito com 10 pixels, depois para a parte inferior com 5 pixels e por último o lado esquerdo com 0, e sempre nessa ordem.

Uma boa dica também é que quando o valor for 0 não precisamos não precisamos colocar a unidade.

A terceira forma é com as propriedades específicas para cada lado, até agora tínhamos visto atalhos para essas propriedades.

Texto

Descrição gerada automaticamente

Essa opção é mais usada quando temos o mesmo valor para 3 lados, e o quarto precisa ter um valor diferente, então usamos o padding com apenas um valor e uma dessas opções para representar o lado diferente.

**Background**

A propriedade background também é um atalho para várias propriedades, mas isso vocês podem absorver aos poucos, e uma boa opção de leitura é a documentação do MDN.

Texto

Descrição gerada automaticamente

Por enquanto veremos apenas como mudar a cor de fundo.

E aqui temos 3 formas de colocar uma cor de fundo, e ainda existem outras.

Texto

Descrição gerada automaticamente

A primeira é pelo nome da cor em inglês, a segunda é pelo código hexadecimal e a terceira é usando apenas o atalho background.

**Border**

Vimos que a propriedade border pode ter 3 valores: a largura, a cor e o estilo, mas existem algumas particularidades nisso.

A largura pode ser usada com várias unidades, como px, em e mm. A cor pode ser atribuída pelo nome ou por um código hexadecimal, assim como fizemos com o background, e o estilo é representada por palavras-chave, vamos ver algumas delas:

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

**solid**: mostra uma borda simples e reta;

**dotted**: são bolinhas com um pequeno espaçamento entre elas;

**dashed**: forma uma linha tracejada.

E aproveitando que mostrei esse código temos que falar sobre como separar a estilização dos lados de uma borda.

Texto

Descrição gerada automaticamente

E se você não quiser usar a propriedade border existem as propriedades específicas para cada aspecto de uma borda, são elas border-width para a largura, border-color para a cor e border-style para o estilo.

Texto

Descrição gerada automaticamente

Aqui temos o mesmo código anterior de duas formas diferentes, a primeira com o atalho border e a segunda com cada propriedade específica.

Texto

Descrição gerada automaticamente

E depois disso podemos juntar os lados com os aspectos de uma borda e criar uma regra mais específica ainda.

**Border-radius**

E a última propriedade é o border-radius, ele permite arredondar os cantos de um elemento. Podemos usar várias unidades, mas as mais comuns são os pixels e a porcentagem.

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

Colocando apenas um valor mudamos todos os cantos do elemento, mas seguindo aquela mesma ordem que vimos no padding e margin - topo, direita, inferior e esquerda - conseguimos alterar cada canto separadamente.

**Exercício prático**

Neste exercício vamos deixar o nosso site um pouco mais bonito usando as propriedades que acabamos de ver.

Vamos aumentar o padding para 15 pixels e colocar uma margem de também de 15 pixels só na parte de baixo do post.

Quando olhamos para os textos percebemos que os espaçamentos estão diferentes do restante do post, então vamos padronizar isso.

No título do post vamos retirar todas as margens para depois colocar apenas uma margem inferior de 15 pixels. E no corpo do post precisamos adicionar uma classe e remover todas as margens para depois adicionar uma margem superior de 15 pixels.

Podemos manter o background branco, mas vamos diminuir a largura das bordas para 2 pixels e mudar a cor para a mesma do texto - #505050 - e por último adicionaremos um border-radius, 5 pixels são suficientes. Podemos adicionar esse mesmo de valor de border-radius na imagem, para isso vamos acrescentar uma class a imagem antes.

Texto

Descrição gerada automaticamente

Interface gráfica do usuário

Descrição gerada automaticamente

**Estilizando textos**

Já sabemos que podemos mudar cor e tamanho de algumas fontes, e agora vamos nos aprofundar nisso.

**font-family**

Com o font-family podemos alterar a fonte dos nossos textos, como uma fonte da internet ou uma que esteja instalada no nosso computador, mas vamos nos ater às fontes seguras, chamadas de web safe fonts.

Texto

Descrição gerada automaticamente

Essas fontes são chamadas assim pois são encontradas em quases todos os sistemas e podem ser usadas sem preocupação.

**font-size**

O font-size nos ajuda a mudar o tamanho do texto, existem algumas unidades de medida para ele mas por enquanto os pixels são suficientes para nós.

Texto

Descrição gerada automaticamente

**font-style**

Usamos o font-style para tornar um texto itálico, na maioria das vezes você usará apenas o valor *italic* para ele, mas se precisar tirar o itálico de um texto você pode usar o valor *normal*.

Texto

Descrição gerada automaticamente

**font-weight**

**Texto

Descrição gerada automaticamente**

Em fontes simples o “normal” e o “bold” resolvem.

- normal – é o peso comum, de acordo com o que foi desenhada

- bold – é o negrito

**text-transform**

Tela de celular com texto preto sobre fundo branco

Descrição gerada automaticamente

Trabalha o texto com letras maiúsculas e minúsculas

- uppercase – coloca todo o texto em caixa alta

- lowercase – coloca todo o texto em caixa baixa

- capitalize – coloca a primeira letra de cada palavra em letra maiúscula

**text-decoration**

**Texto

Descrição gerada automaticamente**

Serve para dar destaque ao texto, colocando linhas

- underline – coloca uma linha abaixo da palavra

- overline – coloca uma linha acima da palavra

- line-through – coloca uma linha no centro, cortando a palavra

**Exercício prático**

Vamos fazer algumas mudanças nos nossos textos.

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Interface gráfica do usuário

Descrição gerada automaticamente

**Estilizando listas**

**list-style-type**

Texto

Descrição gerada automaticamente

- square – altera o marcador de uma lista não ordenada para um quadrado

- upper-roman – altera o marcador de uma lista ordenada para algarismo romano em maiúsculo

- “\1F44D” – altera o marcador de uma lista não ordenada para um símbolo, nesse caso um emoji de um “joinha”

**list-style-image**

Texto

Descrição gerada automaticamente

**Observação:**

O “CSS” funciona com uma cascata:

- temos o elemento “ul” ou “ol” (lista não ordenada / ordenada), dentro dele temos o elemento “li” (item) e dentro do elemento “li”, temos o elemento “a” (âncora), então estilizamos os elementos seguindo esse fluxo.

**Exercício prático**

Vamos melhorar a aparência da nossa lista.

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

**Dimensão e alinhamento**

**Propriedades de dimensões e alinhamento**

Texto

Descrição gerada automaticamente com confiança média

- width – ajusta altura do elemento

- height – ajusta largura do elemento

- max-width – ajusta a largura máxima que um elemento pode ter, se a janela do navegador for menor que largura máxima, o conteúdo se ajustará.

- max-height – ajusta a altura máxima que um elemento pode ter

- margin – valor “auto” para alinhar um objeto automaticamente

- text align – alinha textos, podemos alinhar a esquerda com o valor “left”, a direita com o valor “right”, ao centro com o valor “center” e justificar o texto com o valor “justify”

**Exercício prático**