

MÓDULO VII

Introdução ao CSS3 e Estilização Básica

MÓDULO VII

A regra CSS e sua sintaxe

Uma regra CSS segue uma sintaxe própria que define como será aplicado estilo a um ou mais elementos da marcação HTML de uma página. Um conjunto de regras CSS formam uma folha de estilos.

Uma regra CSS, na sua forma mais elementar, compõe-se de três partes: um seletor, uma propriedade e um valor e tem a sintaxe conforme mostrado a seguir:

```
seletor { propriedade : valor; }
```

MÓDULO VII

O significado de cada uma das partes da regra CSS é conforme descrito a seguir:

1. **seletor:** é um elemento da marcação HTML identificado pelo seu nome (por exemplo: `<p>`, `<h1>`, `<form>`), pelo nome de uma classe aplicada ao elemento da marcação HTML (por exemplo: `.topo`, `.principal`, `.menu`), pelo nome de um identificador ID aplicado ao elemento da marcação HTML (por exemplo: `#tudo`, `#auxiliar`, `#rodape`) ou por qualquer outro identificador CSS de elementos ou trechos da marcação HTML, genericamente denominados de seletores CSS.
2. **propriedade:** é a propriedade do elemento HTML ao qual será aplicada a estilização definida no valor (por exemplo: tamanho da fonte, cor do texto, altura do elemento).
3. **valor:** é a característica específica a ser assumida pela propriedade (por exemplo: letra tipo arial, cor azul, fundo verde, altura igual a 300px)
4. Ao conjunto propriedade: valor denominamos declaração CSS.

MÓDULO VII

Na sintaxe de uma regra CSS, escreve-se o seletor e a seguir a propriedade e o valor separados por dois pontos e colocados entre chaves { }.

É possível escrever mais de uma declarações CSS em uma regra CSS com o objetivo de estilizar-se várias propriedades de um mesmo seletor. Neste caso deve-se usar ponto-e-vírgula para separar as declaração CSS constantes da regra CSS. O ponto-e-vírgula é facultativo quando a regra CSS for constituída de uma só declaração CSS e também facultativo após a última declaração CSS quando houver mais de uma declaração CSS.

É recomendável que se use, sempre, o ponto-e-vírgula após cada declaração CSS, pois no caso de futuramente ter que se acrescentar mais declarações na regra CSS não se corra o risco de esquecer eventual ponto-e-vírgula inicialmente omitido.

MÓDULO VII

Observe os exemplos a seguir que esclarecem a sintaxe da regra CSS.

```
p {  
    font-size: 12px; /* ponto-e-vírgula é recomendável */  
}
```

```
body {  
    color: #000;  
    background-color: #fff;  
    font-weight: bold; /* ponto-e-vírgula é recomendável */  
}
```

MÓDULO VII

No exemplo a seguir, o **seletor** é o "documento todo" (elemento *body*), a **propriedade** é a cor do fundo do documento e o **valor** é a cor branca.

```
body {  
    background-color: #ffff;  
}
```

Se o valor for uma palavra composta, deverá estar entre aspas duplas " ", ou simples ':

```
h3 {  
    font-family: "Comic Sans MS";  
}
```

MÓDULO VII

Para aumentar a **legibilidade** das folhas de estilo, é de boa prática usar linhas distintas para escrever cada uma das declarações — propriedade e seu valor —, bem como separar o valor da propriedade com uso de um espaço em branco, como mostrado a seguir.

```
p {  
    text-align: right;  
    color: #f00;  
}
```

Isto não é obrigatório.

A regra CSS a seguir tem o mesmo efeito da regra anterior e ambas as sintaxes estão corretas.

```
p {text-align:right;color:#f00;}
```

MÓDULO VII

NOTAS:

1. Aumentar a legibilidade da folha de estilo conforme descrito é útil e indispensável na fase de desenvolvimento do projeto. Terminado o projeto deve-se "minificar" a folha de estilos, que é a técnica de se escrever a folha de estilos sem os espaços em branco entre as partes que a compõem e sem os espaços em branco entre as diferentes regras CSS, obtendo uma folha de estilos escrita em uma linha contínua.

Isso visa a diminuir o número de bytes do arquivo da folha de estilo e aumentar a performance diminuindo o tempo de carregamento da página. Existem ferramentas para minificar folhas de estilos.

MÓDULO VII

2. A razão de se recomendar o uso de ponto e vírgula na última declaração ou mesmo quando só há uma declaração é que durante a fase de desenvolvimento da Folha CSS quase sempre estaremos acrescentando novas declarações e a última declaração quase nunca é a última naquela fase. Assim, esta prática certamente nos poupará revisões por ter esquecido um ponto e vírgula.

MÓDULO VII

Agrupamento de Seletores

Agrupando seletores pode-se fazer com que uma regra CSS seja aplicada a mais de um seletor. Para agrupar seletores separe-os com uma vírgula. No exemplo abaixo agrupamos todos os elementos cabeçalho de níveis 1 e 2 os parágrafos e o elemento com a classe `.box`. A cor do texto de todos eles será verde.

```
h1, h2, p, .box {  
    color: green;  
}
```

MÓDULO VII

Seletor classe

Mas você não está restrito somente aos elementos HTML (tags) para aplicar regras de estilo.

Você pode "inventar" um nome e definí-lo como valor a ser atribuído ao atributo *class* (classe) do elemento HTML. O nome "inventado" será o seletor para aplicar declarações CSS. E o mais interessante das classes, é que elas podem ser aplicadas a **qualquer elemento** HTML. E mais ainda, você pode aplicar estilos diferentes para o mesmo tipo de elemento da HTML, usando classes diferentes para cada um deles.

A sintaxe para o seletor classe é mostrada a seguir. Elemento HTML mais um nome qualquer que você "inventa" precedido de . (ponto):

```
elemento.nomedaclasse {  
    propriedade: valor;  
}
```

MÓDULO VII

NOTA: Para o nome que você "inventa" evite usar números e caracteres especiais. Tanto quanto possível use só letras de a-z e de A-Z. Há restrições quanto ao uso de números e caracteres. Minha experiência e conselho: use somente letras e os caracteres - (traço) e _ (sublinhado).

A HTML5 acabou com a restrição dos caracteres e na sintaxe dela você pode usar qualquer caractere exceto espaço em branco, contudo meu conselho anterior continua válido.

Por exemplo: suponha que você precisa de dois estilos para parágrafos em seu documento: um parágrafo com letras na cor preta e um parágrafo com letras na cor azul. Crie duas classes conforme mostrado a seguir.

```
p.cor-um {  
    color: #000;  
}
```

```
p.cor-dois {  
    color: #0ff;  
}
```

MÓDULO VII

Ao escolher nomes para classes não use nomes que lembrem a apresentação. Prefira nomes que lembrem a estrutura. Para o exemplo mostrado nomear as classes *.cor-preta* e *.cor-azul* é uma péssima escolha. Se no futuro você resolver alterar a cor dos elementos azuis para verde vai ficar com uma regra CSS sem sentido.

MÓDULO VII

No seu documento HTML as classes seriam aplicadas conforme mostrado a seguir:

```
<p class="cor-um"> este parágrafo será na cor preta.</p>
<p class="cor-dois">este parágrafo será na cor azul.</p>
```

Em CSS 1 não é válido atribuir mais de uma classe para um elemento HTML. O exemplo abaixo está errado:

```
<p class="cor-um" class="cor-dois">
    Aqui o texto do parágrafo.
</p>
```

MÓDULO VII

As CSS 2 alteraram este conceito, permitindo declarar mais de uma classe, desde que o nome das classes sejam separados por um espaço.

```
<p class="cor-um fonte-dois">  
    Aqui o texto do parágrafo.  
</p>
```

Nesse caso aplicam-se ao parágrafo as regras CSS definidas nas duas classes. Havendo conflito de declarações aplica-se o efeito cascata. No nosso caso o parágrafo será na cor e tamanho de fonte definidos nas regras CSS para as duas classes (cor-um e fonte-dois).

Ao nomear uma classe você talvez queira que ela seja aplicável a qualquer elemento HTML. Neste caso basta que se omita o nome do elemento antes da classe.

MÓDULO VII

Por exemplo: a regra CSS a seguir pode ser aplicada a qualquer elemento HTML ao qual você deseja atribuir cor azul:

```
.cor-tres {  
    color: blue;  
}
```

No exemplo a seguir tanto o cabeçalho `<h2>` como o parágrafo `<p>` serão na cor azul:

```
<h2 class="cor-tres">  
    Este cabeçalho é na cor azul.  
</h2>
```

```
<p class="cor-tres">  
    Este parágrafo é na cor azul.  
</p>
```

MÓDULO VII

Seletor ID

O seletor ID difere do seletor classe, por ser **ÚNICO**. Um seletor ID de determinado nome só pode ser definido a UM e somente UM elemento HTML dentro do documento.

Você pode "inventar" um nome e com ele criar um **ID** que será o seletor para definir a regra CSS. Um determinado **ID só pode ser aplicado a UM elemento** HTML.

A sintaxe para o seletor ID é mostrada a seguir. Um nome qualquer que você "inventa" precedido do sinal # ("cerquilha", "tralha" ou "jogo-da-velha");

```
#meuID {  
    propriedade: valor;  
}
```

MÓDULO VII

Nota: Para o nome que você "inventa" evite usar números e caracteres especiais.

Tanto quanto possível use só letras de a-z e de A-Z . Há restrições quanto ao uso de números e caracteres.

Minha experiência e conselho: use somente letras e os caracteres - (traço) e _ (sublinhado).

A HTML5 acabou com a restrição dos caracteres e na sintaxe dela você pode usar qualquer caractere exceto espaço em branco, contudo meu conselho anterior continua válido.

MÓDULO VII

Inserindo comentários nas CSS

Você pode inserir comentários nas CSS para explicar seu código, e principalmente ajudá-lo a relembrar de como você estruturou e qual a finalidade de partes importantes do código. Daqui há alguns meses a menos que você seja um privilegiado, terá esquecido a maior parte daquilo que você levou horas para "bolar". O comentário introduzido no código, será ignorado pelo navegador. Um comentário nas CSS começa com o sinal /* e termina com */. Observe o exemplo a seguir:

```
/* este é um comentário*/  
p {  
    font-size: 14px;  
    /* este é outro comentário*/  
    color: #000;  
    font-family: Arial, Serif;  
}
```

MÓDULO VII

Segue um resumo com as principais propriedades de estilo da linguagem CSS.

font-family: Define a família da fonte utilizada. Exemplos:

```
font-family: Arial, Verdana, Geneva, sans-serif;
```

```
font-family: Georgia, "Times New Roman", Times, serif;
```

MÓDULO VII

font-style: Define a propriedades de estilos que podem ser: normal, italic ou oblique. Exemplos:

```
font-style: italic;
```

```
font-style: normal;
```

MÓDULO VII

font-weight : Define a propriedade de intensidade de uma fonte na sequência de valores de ‘100’ a ‘900’, cada número indica uma fonte mais intensa (escura) que o valor anterior. Exemplos:

font-weight: normal; (ou valores de 100 a 500)

font-weight: bold; (ou valores de 600 a 900)

MÓDULO VII

font-size: Define o tamanho da fonte. Exemplo:

```
font-size: 12pt;
```

letter-spacing: Controla o espaçamento entre as letras de um texto. Exemplo:

```
letter-spacing: 5px;
```

MÓDULO VII

word-spacing: Controla o espaçamento entre as palavras de um texto. Exemplo:

```
word-spacing: 10px;
```

line-height: Controla a altura entre as linhas do texto de um parágrafo. Exemplo:

```
line-height: 25px;
```

MÓDULO VII

text-align: Controla o posicionamento horizontal do conteúdo de um elemento.

Os valores possíveis são: left, right, center e justify. Exemplo:

```
text-align: center;
```

MÓDULO VII

text-decoration: Define um efeito decorativo no texto.

Podendo entre eles ser: **none** (sem decoração);

underline (sublinhado);

line-through(linha cortando o texto);

e **blink** (efeito piscante). Exemplo:

```
text-decoration: underline;
```

MÓDULO VII

text-transform: Controla os efeitos de capitalização do texto.

Com ela, podemos definir a caixa das letras (minúsculas e maiúsculas). Os valores possíveis são:

- **capitalize** – transforma o primeiro caracter de cada palavra em maiúscula
- **uppercase** – transforma todas as letras de todas as palavras em maiúsculas
- **lowercase** – transforma todas as letras de todas as palavras em minúsculas
- **none** – cancela algum valor que tenha sido herdado

```
text-transform: uppercase;
```

MÓDULO VII

text-indent: Define o tamanho da endentação para a primeira linha do texto contida em um bloco, ou seja, o deslocamento para a direita de um parágrafo. Exemplo:

```
text-indent: 20px;
```

MÓDULO VII

color: Define a cor do texto de um texto. Exemplos:

```
color: red;
```

```
color: rgb(255,0,0);
```

```
color: #FF0000;
```

MÓDULO VII

font: Propriedade para definir em um único local vários estilos para a fonte. A sintaxe geral para font: [style] [variant] [weight] [size] [/line-height] [family]. Exemplo:

```
font: normal small-caps bold 14px "Comic Sans MS", sans-serif;
```

Nota: Os valores size e family são obrigatórios.

Os demais são facultativos (se você os omitir será adotado o valor **default** ou herdado do elemento pai).

MÓDULO VII

width: Define o comprimento (largura) de um elemento. Exemplo:

```
width: 100px;
```

height: Define a altura de um elemento. Exemplo:

```
height: 50px;
```

MÓDULO VII

border: Define bordas para um elemento. Exemplo:

```
border:1px solid #CCC;
```

```
border-bottom:5px solid #97B2B9;
```

MÓDULO VII

background: Define as propriedades relacionadas ao fundo de exibição. Exemplo:

```
background:url ("images/fundo-topo.png") no-repeat;
```

margin: Controla as margens de um elementos. Se forem indicados quatro valores, eles dizem respeito, respectivamente, às margens superior, direita, inferior e esquerda. Se for fornecido apenas um valor, ele é aplicado às quatro margens.

```
margin: 15px;
```

```
margin: 10px 1px 5px 20px;
```

MÓDULO VII

A última atribuição do exemplo acima é equivalente a:

margin-top: 10px;

margin-right: 1px;

margin-bottom: 5px;

margin-left: 20px;

MÓDULO VII

padding: Controla os espaçamentos de um elementos. Se forem indicados quatro valores, eles dizem respeito, respectivamente, aos espaçamentos superior, direito, inferior e esquerdo.

Se for fornecido apenas um valor, ele é aplicado aos quatro espaçamentos.

```
padding: 5px;
```

```
padding: 10px 5px 5px 10px;
```

MÓDULO VII

float: Esta regra faz com que o box seja retirado de sua posição no fluxo do documento e flutuado para a direita ou esquerda. O espaço original ocupado pelo box não será deixado livre, mas preenchido pelo elemento que segue no fluxo do documento. Exemplo:

```
float: left;
```

```
float: right;
```

MÓDULO VII

Tabela de cores HTML: aplicando cores nas tags HTML

Durante o desenvolvimento de um website, existem diversos fatores que interferem diretamente na qualidade do mesmo, e em como o usuário receberá as informações do conteúdo. Uma das coisas mais importantes para a criação de um layout é a escolha certa das cores a utilizar. Por esta razão, ter à mão ferramentas que facilitam no processo como uma tabela de cores HTML é essencial.

Através das linhas de código **HTML e CSS**, podemos aplicar cores tanto para fontes como para seus elementos. Podemos até mesmo aplicar cores às bordas e sombreamentos. São muitas possibilidades que nos permite criar layouts mais complexos e com diversas características.

MÓDULO VII

Com os diversos tipos de display, grids, flexbox, floats, é possível criar inúmeras possibilidades visuais só utilizando o HTML e CSS, assim como aplicar diversos backgrounds, animações e até mesmo vetores.

Assim, é importante entender quais são os padrões de cores que o HTML suporta, e como são formados. Após compreender esse tema, você poderá então começar aplicar as cores através do **estilos em HTML** e **CSS**. Para tanto, disponibilizamos uma **tabela de cores HTML**, com os nomes das cores, o código hexadecimal e o valor RGB prontos para usar diretamente no seu estilo!

MÓDULO VII

Tabela de cores HTML

Na tabela de cores HTML, você poderá encontrar diversos exemplos de cores com seu código hexadecimal, assim como o nome em inglês e o código RGB, já prontos para copiar diretamente para o seu código HTML/CSS:

Nome da Cor	Código hexadecimal	Código RGB
Black	#000000	(0,0,0)
grey11	#1C1C1C	(28,28,28)
grey21	#363636	(54,54,54)
grey31	#4F4F4F	(79,79,79)
DimGray	#696969	(105,105,105)

MÓDULO VII

Já imaginou como seria um website apenas com cores preto e branco e sem a inserção de multimídia?

Antes do surgimento do protocolo **HTTP** e da linguagem **HTML**, a navegação de sites era apresentada através de uma tela similar ao do terminal, com fundo preto e fonte branca ou verde, onde a navegação era através do teclado. Um ótimo exemplo disso é o The Gopher Project.

Em 1989 o britânico **Tim Berners-Lee** dava início a um projeto que mais tarde seria conhecido como HyperText Markup Language, o **HTML**, a principal linguagem quando o assunto é desenvolvimento web.

Em 1991 surge o primeiro **website**, que ainda está online.

MÓDULO VII

Com o HTML, passou a ser possível a inserção de multimídia e uma melhor manipulação de fontes, cores e posicionamento de elementos. Por consequência, conforme foram lançando novas versões, o HTML foi recebendo novas funcionalidades, até chegar no que conhecemos hoje, o **HTML5**.



MÓDULO VII

As cores no HTML e CSS

No **HTML** e **CSS**, as cores podem ser definidas através de seu nome (*em inglês*), através do código hexadecimal, ou através do seu valor **RGB**. Com as novas versões do CSS, é possível também utilizar os valores **RGBA**, **HSL** e **HSLA**. Esses valores são intensidades de cores ou propriedades (como saturação e opacidade) que juntos formam uma outra cor.

Então, após compreender como as cores funcionam, é interessante estar sempre com a tabela de cores **HTML com fácil acesso para consultar sempre que precisar**.

MÓDULO VII

Cores Hexadecimais

O código Hexadecimal para cores consiste em seis letras ou números precedidos do “#”.

Os dois primeiros elementos representam a intensidade de vermelho, o terceiro e quarto elementos representam a intensidade de verde e os dois últimos a intensidade de azul. Desta forma, o resultado final será a combinação dessas três cores em diferentes intensidades, formando a cor desejada.

Os valores variam entre 00 (mais escuro) até FF (mais claro). Por exemplo, temos o código #FFFFFF, que representa a cor branca, pois tem a intensidade máxima das três cores. Do mesmo modo, #FF0000 representa a cor vermelha, com intensidade máxima de vermelho e nenhuma intensidade para verde e azul. Na tabela de cores HTML, é possível encontrar mais de 100 códigos hexadecimais das cores mais utilizadas.

MÓDULO VII

Valores Hexadecimais

Valores **hexadecimais** usados em tabela de cor **HTML** não são especialmente complicados, mas podem ser difíceis para se lembrar. Como o prefixo “**hexa-**” indica, estas notações incluem 6 algarismos totais, o que pode ser letras ou números. Os 6 dígitos de uma notação **hexadecimal** ou **hex** em uma tabela de cor

HTML indicam a quantidade de vermelho, verde e azul, ou **RGB** presentes em cor particular.

O vermelho é indicado pelos 2 primeiros dígitos, verde pelos 2 dígitos do meio, e a quantidade de azul é com base nos 2 dígitos finais. Um valor de “**00**” em notação **hex** é o menor valor possível, e em tabela de cor **HTML** indica nenhuma quantidade dessa cor. Isto quer dizer que uma notação **hex** de “**oooooooo**” não tem vermelho, verde, ou azul, e assim, é exibido como preto. O valor mais alto possível neste tipo de notação **hex** é “**FF**”, que indica tanto quanto de uma cor possível.

MÓDULO VII

Uma notação hex de “**FFFFF**F” em uma tabela de cor **HTML** indica branco, o valor mais alto possível de todas as 3 cores combinadas. E vários tons e sombras diversos podem ser gerados por meio desta notação **hex**, que normalmente são mostrados em tabela de cor **HTML**. Ao invés de mostrar cada combinação possível, no entanto, estas tabelas geralmente mostram uma gama limitada ou paleta que é funcional e pode guiar alguém usando **HTML**. Muitas tabelas mostram **216** cores que são relacionadas como “seguras para web” quando monitores tinham uma paleta potencial mais limitada. Existem milhões de cores que podem ser geradas usando notação **hex**, embora o diferencial entre “**0010FF**” e “**0011FF**” se mostra muito difícil para notar.

MÓDULO VII

Paleta de Cor e Usuário Não importa qual método prefere, códigos de cor **RGB** e **hex** são baseados em misturar e intensificar as 3 cores para criar muitas cores diferentes. No entanto, cores na web misturam juntas diferentemente do que pode esperar se misturasse tinta de cor diferente.

Sliding Scale Sliding scale da cor é o que a maioria das representações das paletas de cor usa em lugar de um círculo de cor. E segue através de todas as cores do arco-íris, começando em vermelho, e continuando pelo laranja, amarelo, verde, azul, roxo e rosa até que retorna para o vermelho. O valor **RGB** de vermelho é responsável pelos tons quentes de cor, como vermelho, rosa e laranja. Quanto mais alto este valor, mais quente a cor será. O vermelho é a primeira cor encontrada na sliding scale, com um valor de **rgb(255, 0, 0)**, ou **#FF0000**. Conforme trabalha pela escala, mais do valor verde **RGB** é adicionado para criar laranja, então amarelo. Amarelo com valor de **rgb(255, 255, 0)**, ou **#FFFF00**, é a cor que vermelho e verde têm em comum.

MÓDULO VII

A partir deste ponto, o valor vermelho é reduzido para oferecer diferentes tons de verde.

Eventualmente vai atingir verde em sliding scale, com um valor de **rgb(0, 255, 0)** ou **#00FF00**.

A partir dali, o valor azul **RGB** é adicionado para eventualmente criar ciano. Ciano com um valor de **rgb(0, 255, 255)** ou **#00FFFF**, é a cor que verde e azul têm em comum. Similar a antes, o valor verde é então reduzido para criar tons diferentes de azul. Para o final de sliding scale, pode começar adicionando o

valor vermelho **RGB** de volta para criar cores como roxo e rosa. Uma vez que atinge magenta, com um valor de **rgb(255, 00, 255)**, ou **#FF00FF**, o valor azul é então reduzido para eventualmente alcançar a cor vermelho original que começou. **Valores RGB e Hexadecimal** Na conclusão, ambos valores **RGB** e **hexadecimal** podem ser usados para mostrar cores em página web.

MÓDULO VII

Não há certo ou errado para escrever valores de cor, mas cada um tem suas vantagens e desvantagens a depender da tarefa que está tentando completar. Além disso, independente de qual convenção de nome de cor usar, pode facilmente manipular o tom de cor por ajustar os valores que está utilizando.

Não há uma única técnica que pode oferecer um código para cada e toda cor. Por vezes precisa de um tom mais brilhante, ou um mais turvo, que não pode ser obtido por simplesmente mudar um ou dois valores. Não ter medo de ajustar todos os 3 valores, os resultados podem ficar melhores do que estava desejando antes. Quando precisamos aplicar cores HTML (cores em um documento HTML) também necessitamos saber os valores, os códigos hexadecimais deles. Isso porque o formato HTML suporta apenas esse formato para cores. Ou seja, precisamos entender como funcionam as cores HTML de um site em um sistema hexadecimal.

MÓDULO VII

RGB

Do mesmo modo, valor RGB significa **R**ed, **G**reen, **B**lue (*vermelho, verde, azul*) e sua teoria é a mesma, a intensidade dessas três cores formarão a cor final. Os RGB é declarado com três valores com escala entre 0 e 255. Dessa forma temos o código **rgb(255,255,255)** para a cor branca, **rgb(255,0,0)** para a cor vermelha e **rgb(0,0,0)** para a cor preta. Na tabela de cores **HTML**, é possível encontrar mais de 100 valores **RGB** das cores mais utilizadas.

MÓDULO VII

RGBA

No sistema **RGBA**, é utilizado o mesmo código do sistema **RGB** porém acrescenta-se um novo valor, o **Alpha**. O valor alpha significa a opacidade da cor, que varia de uma escala entre decimais de 0.0(*transparente*) e 1.0 (*opaco*). É muito utilizado para a sobreposição de elementos, sem perder a visibilidade do elemento inferior, dando a sensação de transparência. Por exemplo: para a cor preta com 50% de opacidade, utilizamos o **rgba(0,0,0,0.5)**.

MÓDULO VII

Antes de tudo, você sabe como as cores em **RGB** se formam? **RGB** significa **Red, Green, Blue** (vermelho, verde e azul), que são as cores primárias usadas na formação de todas as outras cores.

No mundo das telas digitais (TVs e computadores, por exemplo) esse é o padrão. E para que as diversas cores que vemos em um monitor se formarem, elas precisam ser definidas pelo código hexadecimal.

Mas como? Comece considerando que cada cor tem **256** tons, no código hexadecimal essa cor é definida pelo seguinte:

MÓDULO VII

- sinal de sustenido, ou jogo da velha (#);
- seis dígitos sendo: os dois primeiros referentes a intensidade da cor primária vermelha, os dois seguintes a intensidade da cor verde e os dois últimos dígitos a intensidade do azul.
- números de **0** a **9**.
- letras de **A** a **F**

Ou seja, o código possui 2 dígitos para cada cor e cada cor pode ter **256** tonalidades.
Nesse sentido as cores primárias ficam assim:

- vermelho: **#FFoooo**
- verde: **#ooFFoo**
- azul: **#ooooFF**

MÓDULO VII

Mas também existem aqueles códigos hexadecimais com oito espaços.

Nesse caso, é porque eles representam a aplicação de configuração das cores **HTML** como transparência e opacidade.

Geralmente **FF** é a opacidade máxima e **00** totalmente transparente.

Veja só um exemplo de como ficaria o código caso você aplicasse um verde semi transparente como cor

HTML do seu site: **#7700FF00**.

MÓDULO VII

A tabela **hexadecimal** é uma tabela numérica que utiliza a base **16**, em vez da base **10** utilizada na matemática convencional. A tabela é usada em áreas como informática, eletrônica e programação, e é especialmente útil para representar valores binários de uma forma mais compacta e fácil de entender.

A tabela **hexadecimal** é composta pelos seguintes caracteres:

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E e F. Cada um desses caracteres corresponde a um valor decimal de **0** a **15**. Por exemplo, o valor **hexadecimal** “A” corresponde ao valor decimal **10**, e o valor **hexadecimal** “F” corresponde ao valor decimal **15**.

A tabela hexadecimal é usada para representar valores binários de uma forma mais compacta e fácil de entender. Por exemplo, um número binário de **8** bits (ou seja, um número que pode ter até **8** dígitos binários) pode ser representado por dois dígitos **hexadecimais**.

Para converter um número binário em hexadecimal, o número binário é dividido em grupos de **4** bits, e cada grupo é convertido em um único dígito hexadecimal.

MÓDULO VII

A tabela hexadecimal também é usada para representar cores em computadores e outros dispositivos eletrônicos. As cores são representadas por combinações de valores hexadecimais que correspondem às intensidades de vermelho, verde e azul que compõem a cor. Por exemplo, a cor branca é representada pelo valor hexadecimal #FFFFFF, que corresponde às intensidades máximas de vermelho, verde e azul. Existem várias tabelas de cores, sendo as mais comuns a tabela **RGB**, a tabela **HEX** e a tabela **CMYK**.

MÓDULO VII

Tabela **RGB**:

- Vermelho: (255, 0, 0)
- Verde: (0, 255, 0)
- Azul: (0, 0, 255)
- Amarelo: (255, 255, 0)
- Ciano: (0, 255, 255)
- Magenta: (255, 0, 255)
- Branco: (255, 255, 255)
- Preto: (0, 0, 0)

Tabela **CMYK**:

- Vermelho: (0%, 100%, 100%, 0%)
- Verde: (100%, 0%, 100%, 0%)
- Azul: (100%, 100%, 0%, 0%)
- Amarelo: (0%, 0%, 100%, 0%)
- Ciano: (100%, 0%, 0%, 0%)
- Magenta: (0%, 100%, 0%, 0%)
- Branco: (0%, 0%, 0%, 0%)
- Preto: (0%, 0%, 0%, 100%)

Tabela **HEX**:

- Vermelho: #FF0000
- Verde: #00FF00
- Azul: #0000FF
- Amarelo: #FFFF00
- Ciano: #00FFFF
- Magenta: #FF00FF
- Branco: #FFFFFF
- Preto: #000000

MÓDULO VII

Além dessas cores básicas, existem muitas outras cores disponíveis na tabela **HTML**, incluindo tons de cinza, cores pastel e cores vivas. Você pode encontrar uma lista completa de cores **HTML** em sites de referência ou utilizando softwares de edição de imagens.

Você sabia que as cores são uma parte fundamental do design do seu site? Elas podem transmitir emoções e sentimentos para seus visitantes, além de dar uma identidade visual única à sua marca.

E uma das formas mais precisas de escolher as cores do seu site é utilizando as cores hexadecimais.

As cores **hexadecimais** são representações numéricas de cores que podem ser usadas em seu site.

Elas são formadas por seis caracteres, que vão desde 0 até F, e representam as quantidades de vermelho, verde e azul em uma cor.

MÓDULO VII

Por exemplo, **#FFoooo** é a representação **hexadecimal** da cor vermelha pura. Ao utilizar as cores **hexadecimais** em seu site, você tem um controle absoluto sobre as cores utilizadas e pode garantir que as cores correspondam exatamente à sua marca. Além disso, as cores **hexadecimais** são facilmente reproduzíveis em diferentes dispositivos, garantindo que seu site tenha uma aparência consistente em todos os lugares. Outra vantagem de usar as cores **hexadecimais** é que elas são amigáveis aos mecanismos de busca. Os robôs do Google e outros motores de busca usam as cores **hexadecimais** para entender a estrutura do seu site e podem interpretar a cor de fundo e outras cores importantes.

Então, se você quer um site profissional, moderno e dinâmico, usar as cores hexadecimais é uma escolha inteligente.

Você terá controle total sobre a aparência do seu site e poderá garantir que ele transmita as emoções e sentimentos certos para seus visitantes. Além disso, as cores **hexadecimais** ajudam na otimização para mecanismos de busca, o que significa que você terá uma maior visibilidade na web.

MÓDULO VII

HSL

Diferentemente, o sistema **HSL** representa **Hue** (*tom*), **Saturation** (*saturação*) e **Lightness** (*luminosidade*). O valor do tom é medido em ângulos expresso em graus. O valor de saturação é medido em porcentagem, sendo 0% um leve sombreado de saturação e 100% uma saturação total da cor. O valor de luminosidade também é expresso em porcentagem. Por exemplo: um valor igual a 100% resulta em cor branca e 0 em cor preta, sendo 50% o valor mais usual.

MÓDULO VII

HSLA

Assim como no **RGBA**, no sistema **HSLA** segue o mesmo padrão do **HSL**, acrescentando o valor **Alpha**, que representa a opacidade. Também varia numa escala entre 0.0 e 1.0.

MÓDULO VII

O que é CMYK?

CMYK (Cyan, Magenta, Yellow, Key/Black)

- **CMYK:** Utilizado para impressão em papel, como em revistas, folhetos, cartazes e materiais gráficos. O modelo **CMYK** é um sistema **subtrativo**, onde as cores são criadas pela combinação de pigmentos (tinta).
Quando todos os pigmentos são misturados na máxima intensidade, resultam em preto.
- Já a combinação de todas as cores na menor intensidade resulta em branco (ou, mais precisamente, no branco do papel).

MÓDULO VII

Em resumo:

- **RGB** é ideal para **dispositivos digitais**, pois utiliza luz para formar cores.
- **CMYK** é usado em **impressão**, pois utiliza pigmentos para criar as cores.

MÓDULO VII

GUIA EMOCIONAL DAS CORES

OTIMISMO	clareza calor
AMIGÁVEL	alegria confiança
EXCITAÇÃO	juventude coragem
CREATIVO	imaginativo sabedoria
CONFIANÇA	seguro força
PACÍFICO	crescimento saúde
EQUILÍBRIO	neutro calmo



MÓDULO VII

o significado das cores

- **vermelho:** energia, desejo, paixão, raiva, calor, amor;
- **rosa:** amor, felicidade, romantismo, charme, delicadeza, feminilidade;
- **amarelo:** sabedoria, conhecimento, imaginação, verão, desonestidade, covardia, inveja;
- **laranja:** energia, equilíbrio, calor, entusiasmo, extravagância;
- **verde:** cura, calma, natureza, saudável, renovação, juventude.
- **azul:** espiritualidade, paz, tranquilidade, calma, harmonia;
- **roxo:** misterioso, transformação, sabedoria, poder, sensibilidade, intimidade
- **marrom:** excitação, terra, casa, confiabilidade, conforto, estabilidade
- **preto:** sofisticação, mistério, medo, profundidade tristeza, remorso;
- **branco:** respeito,, pureza, limpeza, paz, juventude.

MÓDULO VII

Aplicando cores nos elementos HTML

No **HTML**, podemos customizar as cores nos elementos através do seu estilo, podendo ser feito na própria tag, com a utilização da propriedade style ou através de uma folha de estilo, seja ela interna ou externa.

Modificando as cores inline das tags HTML

Para modificar o estilo de um elemento inline, basta acrescentar dentro da tag do elemento o atributo style e assim realizar as modificações.

MÓDULO VII

Por exemplo: se tivermos um elemento <p> e queremos alterar sua fonte para a cor azul, bastaria utilizar o código:

```
<p style="color:blue">Azul</p>
```

Como resultado, teremos:

Azul

MÓDULO VII

Mas e se quisermos modificar também o background do elemento? Então podemos utilizar a propriedade background-color para isso!

Dessa forma, vamos aplicar ao nosso elemento p um fundo vermelho:

```
<p style="color: blue; background-color: red">Azul</p>
```

Como resultado, teremos:

Azul

MÓDULO VII

Modificando as cores através de uma folha de estilos CSS

Através das tags **<script></script>**, podemos inserir um código **CSS** dentro dela, ou podemos também aplicar um estilo externo. Dessa forma, fica muito mais fácil para alterar os estilos dos elementos.

Por exemplo, vejamos a seguinte estrutura em HTML:

```
<body>
    <p>Olá mundo</p>
    <p>Olá mundo</p>
    <p class="vermelho">Eu sou vermelho</p>
    <p class="azul">Eu sou Azul</p>
    <p class="bgciano">Eu tenho um Background ciano!</p>
</body>
```

MÓDULO VII

Como resultado inicial, teremos:

Olá mundo

Olá mundo

Eu sou vermelho

Eu sou Azul

Eu tenho um Background ciano!

MÓDULO VII

Repare que nada aconteceu, pois ainda não aplicamos nenhum estilo.
Mas se quisermos aplicar a cor púrpura a todos os elementos **p**, basta utilizar o **CSS**:

O resultado será:

```
p {  
    color: #ff00ff;  
}
```

Olá mundo

Olá mundo

Eu sou vermelho

Eu sou Azul

Eu tenho um Background ciano!

MÓDULO VII

Aplicando agora o estilo para as classes, teremos um novo resultado.

```
p{  
    color: #FF00FF;  
}  
  
p.vermelho{  
    color:red;  
}  
  
.azul{  
    color:rgb(0,0,255);  
}  
  
.bgciano{  
    background-color: #00FFFF;  
}
```

Olá mundo

Olá mundo

Eu sou vermelho

Eu sou Azul

Eu tenho um Background ciano!

MÓDULO VII

Da mesma forma, você também pode aplicar essa técnica para diversos elementos e propriedades, gerando cores para os textos e background do seu site. Contudo, para criar um layout bonito, é importante entender como as cores funcionam e interagem entre si. Mas se você não puder se aprofundar no estudo da Teoria das Cores, há ferramentas que podem auxiliar.

MÓDULO VII

Ferramentas para utilizar em conjunto com a tabela de Cores HTML e CSS

Vale a pena utilizar algumas paletas de cores, veja abaixo algumas sugestões:

- **Material UI Colors:** Neste site é possível encontrar as cores padrões mais utilizadas no Material Design, são cores vibrantes porém sofisticadas e modernas, muito utilizadas para produção de websites hoje em dia. Ao clicar em cima da cor, a ferramenta já copia o código hexadecimal automaticamente para você, podendo colar diretamente no arquivo HTML e CSS! Vale a pena dar uma olhada;
- **Color Hunt:** Neste Site, é possível encontrar diversas paletas de cores, feitas tanto por profissionais como por amadores, você pode selecionar as que mais te agradem e utilizar;
- **Adobe Color:** Esta ferramenta pertence a Adobe, é muito útil para gerar combinações de cores, ao clicar em uma cor, ela já seleciona as cores que se enquadram na mesma, vale muito a pena utilizar essa ferramenta!

MÓDULO VII

EXERCÍCIOS...

MÓDULO VII

FIM...