PROGRAMAÇÃO BÁSICA

NATANAEL ANTONIOLI

2 FUNDAMENTOS DO CSS

Programação Básica - Fundamentos do CSS

Por: Fábrica de Noobs

Índice

1) Prefácio

Se você está lendo isso, então é porque meu projeto deu certo e decidi publicar o curso. Antes de irmos propriamente ao conteúdo, vou falar um pouco sobre o objetivo dele.

Todo o conteúdo deste curso é voltado ao público iniciante, para aquele que está interessado no HTML/CSS não para sair trabalhando com ele amanhã, mas para ter uma noção básica de como a coisa funciona, e usá-la no futuro ou em alguma outra área – em determinados conteúdos de hacking, pode ser importante, por exemplo – além disso, o curso é voltado para o contexto atual, onde já temos programas que automatizam boa parte do processo, tornando o ato de programar algo menos monótono.

Já vi alguns cursos que obrigavam o aluno a programar no bloco de notas, e repetir a mesma coisa centenas de vezes – isso quando não pediam para copiar códigos – é exatamente o oposto que você encontrará aqui.

Pretendo futuramente gravar todo o conteúdo desse curso no canal Fábrica de Noobs, mas por enquanto será apenas por texto.

Recomendo que não fique preso somente ao conteúdo dele, vá por conta própria, faça experiências, procure por outros exercícios.

Pois bem, é isso. Agradeço por ter se interessado pelo meu curso, não sou muito experiente em dar esse tipo de aulas, então sinta-se à vontade para qualquer tipo de sugestão e crítica.

Bons estudos!

2) Introdução ao CSS

Caso você tenha uma noção de HTML (ou principalmente, caso tenha lido o curso anterior), deve ter percebido que a determinação de atributos (cor, tamanho e fonte, por exemplo) é algo um tanto “repetitivo” de se fazer em HTML puro.

Basicamente, a função do CSS é facilitar tal tarefa. Através de generalizações, pode-se definir atributos que servirão para mais de uma tag, ou para todas as tags com determinada identificação.

```
@charset "utf-8";
/* CSS Document */
h1 {
    font: Cambria;
    font-size:12px;
    color:red;
}

h2 {
    color:green;
}
```

No exemplo acima, definimos que todas as tags <h1> apresentariam fonte Cambria, em 12px e cor vermelha; ao passo que todas as <h2> seriam verdes.

Basicamente, a função do CSS é essa. Mais adiante, veremos outras formas de fazer essa generalização.

O script deve ser salvo em um arquivo de extensão .css, chamado também de folha de estilos ou stylesheet.

3) Programas Recomendados

Os programas que podem ser usados no CSS são os mesmos que apresentei no curso de HTML.

- **Notepad++**: é um editor de texto um pouco mais bem trabalhado, com cores e tabulações melhores. Recomendo usá-lo no início, para entender o básico que é normalmente automatizado em outros programas.
- **Adobe Dreamweaver**: indico para quem já conhece os fundamentos – ou seja, para você daqui a alguns capítulos – e quer agilizar a programação dos códigos. Sua grande vantagem é que não é necessário escrever os elementos por inteiro, eles já aparecem intuitivamente. Isso significa maior facilidade na programação, mas requer um pouco mais de prática. O programa é pago, porém você pode baixá-lo pirata aqui http://fabricadenoobs.wix.com/home#!Adobe-Dreamweaver-CSS/c7i/55972ff50cf2604da3499ddd, ou caso tenha baixado o curso completo, ele vem como material complementar.
- **Brackets**: semelhante ao Dreamweaver, em versão grátis. Não conta com tantos recursos e possui uma mecânica um tanto diferente, mas não deixa de ser interessante testá-lo.

4) Vinculação CSS/HTML

Uma das questões que você deve ter levantado é “como fazer com que o script em CSS seja relacionado a página em HTML?”.

Tal vinculação é feita a partir de algumas tags HTML a serem inseridas dentro da tag <head> no documento. Usamos a tag <link> com os atributos rel e type, igualados a stylesheet e text/css, respectivamente. Por fim, usamos href para indicar a folha de estilo CSS. Veja no exemplo, onde tal folha é chamada de *estilo.css*.

```
<link rel="stylesheet" type="text/css" href="estilo.css">
```

Acima, vinculamos uma página em HTML a uma folha de estilos CSS chamada de estilo.css. É importante que, ao publicar, ambas estejam no mesmo diretório.

Também é possível vincular o script em CSS na mesma página que o HTML, a chamada folha de estilo incorporada, o que não entrarei em detalhes, já que se trata de um curso básico.

Exercício 1 – Utilizando seus conhecimentos, vincule a página *index.html* com a folha de estilo *stylesheet.css*.

A partir desse momento, você sabe como vincular ambas as páginas. Sendo assim, nos próximos exercícios, já partiremos do pressuposto de ter uma página HTML criada e vinculada para formatar.

5) Sintaxe

Em CSS, organizamos nosso script com uma sintaxe bastante simples. Ela é dividida em Seletor, Declaração, Propriedade e Valor. Veja no exemplo:

```
h1 {
        color:red;
}
```

Sendo assim, temos que:

- O seletor é responsável por identificar a tag HTML que se deseja generalizar. No exemplo, é a tag <h1>.
- A declaração corresponde a todo o conteúdo entre chaves, que corresponde aos critérios estabelecidos para a tag de seu seletor.
- A propriedade corresponde a qual critério da tag deseja-se mudar. Pode ser a cor, a fonte, o tamanho da fonte, por exemplo. No caso, nos referimos ao atributo color. Tais atributos são praticamente os mesmos que usávamos dentro do atributo style, em HTML.

- O valor é a característica da propriedade que queremos alterar. No caso, vermelho, ou red.

Uma das características de um seletor, é que, sua propriedade se aplica as demais tags dentro dele. Assim, se inserirmos um seletor para a tag <body>, todas as outras tags dentro dela possuirão as mesmas características declaradas.

Por fim, percebemos que a folha de estilos não é nada mais que uma sucessão de seletores, identificando e caracterizando cada parte de nosso código HTML.

Outro ponto importante é sempre utilizarmos chaves para fechar cada seletor, e terminarmos cada declaração com ponto-e-vírgula.

Por fim, veja novamente esse exemplo e tente identificar quais formatações são usadas e onde elas se aplicam.

```
@charset "utf-8";
/* CSS Document */
h1 {
    font: Cambria;
    font-size:12px;
    color:red;
}

h2 {
    color:green;
}
```

6) Seletores

Para podermos formatar a página HTML com mais praticidade, foram criados vários tipos de seletores, que vão desde os mais genéricos, até específicos.

O primeiro que usaremos são os **seletores de tipo**. Foi tal seletor que usamos nos exemplos até o momento. Ele é responsável por dar um estilo para todas as suas tags. Assim, ao usarmos um seletor de tipo h1, estamos dizendo que todo o conteúdo dentro das tags <h1> da página deve possuir a determinada formatação.

Para usarmos um seletor de tipo, basta escrever a tag desejada, sem nenhum outro caractere adicional. Veja no exemplo:

```css
p {
    color:red;
    font-size:16px;
    font:Cambria;
}

h2 {
    color:green;
    font-size:13px;
    font:Arial;
}
```

Aqui, todas as tags <p> e <h2> terão suas respectivas formatações.

Exercício 2 – Faça com que os elementos da página *index.html* possuam as formatações especificadas.

Você deve se lembrar que, em HTML, era possível termos uma tag dentro de outra tag. Caso contrário, veja no exemplo:

```html
<body>
<h1> Título em vermelho </h1>
<a> Parágrafo em azul </a>

<div>
    <h3> Subtítulo em Verde </h3>
    <a> Parágrafo em amarelo  </a>
</div>

</body>
```

Aqui, temos um parágrafo que deve vir em azul, e outro que deve sair em amarelo. Porém, se usarmos apenas o seletor a, não teremos como diferenciar os parágrafos.

Note que o primeiro <a> está sozinho, enquanto que o segundo está dentro da tag .

Podemos nos aproveitar disso, usando um **seletor descendente**. Tal seletor pode ser usado para especificar uma sequência de tags, sendo a primeira “exterior” e a última, a mais “interior”. Veja no exemplo:

```css
a {
    color:green;
}
div a {
    color:yellow;
}
```

Quando usamos apenas a como seletor, dizemos para ele aplicar tal formatação a todas as tags <a> da página, independentemente de como estejam.

Já quando criamos outro seletor, dessa vez com div a, fazemos com que a formatação declarada se aplica somente as tags <a> que estão dentro das tags .

Lembre-se que, se temos dois seletores conflitantes, o que for mais específico irá permanecer.

Mas e se tivermos várias tags iguais e de mesma representação, que necessitam de formatações diferentes, como no exemplo abaixo?

```html
<body>
<h1> Título em Verde Escuro </h1>
<h1> Título em Roxo </h1>
</body>
```

É aqui que entram os **seletores de classe e ID**.

Podemos especificar uma classe para cada tag <h1> do código. Fazemos isso inserindo o atributo HTML class dentro da tag, e o igualando a um valor qualquer, que será usado na folha de CSS:

```html
<body>
<h1 class="verde"> Título em Verde Escuro </h1>
<h1 class="roxo"> Título em Roxo </h1>
</body>
```

Feito isso, devemos agora inserir os seletores no código CSS:

```css
.verde {
    color:#073309;
}

.roxo {
    color:#3B177C;
}
```

Como pode-se perceber, anunciamos uma classe em CSS utilizando o ponto depois de seu nome.

Também podemos usar a mesma classe em vários elementos, ou usar várias classes em um mesmo elemento. Por exemplo:

```
<body>
<h1 class="verde grande" > Título em Verde Escuro </h1>
```

```
.verde {
    color:#073309;
}
.grande {
    font-size:36px;
}
```

Assim, se quisermos usar mais de uma classe na mesma tag, basta lista-las dentro do atributo class com um espaço entre elas.

Existem também os **seletores de ID**. Eles funcionam praticamente da mesma forma que os seletores de classe, com algumas diferenças cruciais: seletores de ID são únicos.

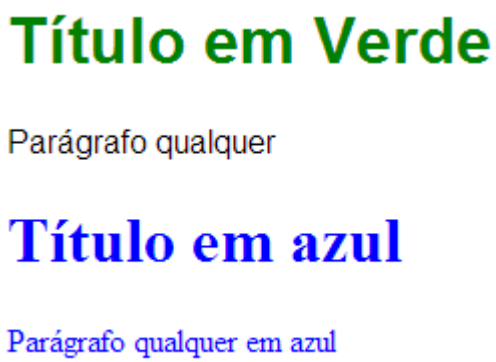
O mesmo seletor de ID só pode aparecer uma vez em todo o documento, e uma tag só pode possuir um único seletor de ID.

Sendo assim, tais seletores são mais usados para formatar áreas maiores do documento. Veja no exemplo:

```
<div id="arial">
    <h1 class="verde"> Título em Verde </h1>
    <p> Parágrafo qualquer  </p>
</div>

<div id="azul">
    <h1> Título em azul </h1>
    <p> Parágrafo qualquer em azul </p>
</div>
```

```css
#arial {
    font-family:Arial;
}

#azul {
    color:blue;
}

.verde {
    color:green;
}
```

Aqui, dizemos ao editor que a primeira divisão terá o id=''arial'', ao passo que a segunda, terá o id=''azul''. Na folha de estilo, criamos um seletor de ID com o uso da hashtag. Verifique o resultado:



Lembre-se que é possível construir uma folha em CSS usando somente ID's, classes, ou nenhum dos dois. Vai da preferência do programador.

Exercício 3 – Abra a página *index.html*. Nela, há uma série de comentários indicando como sua formatação final deve ficar. Usando CSS, faça com que isso aconteça, da forma que prefrir.

7) Conclusão

Por fim, terminamos nosso curso de CSS. O que vimos é apenas uma introdução, o básico que você precisa para entender como o CSS e HTML funcionam. Caso queira se especializar no assunto, recomendo procurar outros cursos para se aprofundar.

Porém, caso preferia continuar a entender outras linguagens de programação, recomendo esperar o próximo curso – que será sobre lógica de programação. É o

primeiro passo para começar a compreender linguagens realmente de programação, e não apenas de marcação, como vimos até agora.