

HTML5

Definição e estrutura básica

Em 1991 Tim Berners-Lee criou essa linguagem de marcação para melhorar a comunicação entre ele e seus colegas de trabalho no CERN, desde então já surgiram 5 versões e o HTML se tornou a base da web.

Com o HTML definimos o significado e a estrutura do conteúdo da web e, além de texto, nossas páginas precisam de imagens, vídeos e vários outros formatos e para isso temos os elementos HTML.

Um elemento HTML é formado pela tag de abertura e seus atributos, o conteúdo e uma tag de fechamento. E mais a frente veremos que existem elementos que não tem tag de fechamento.

Com esses elementos podemos agrupar tipos de conteúdo, alterar tamanho e forma de fontes e adicionar diferentes mídias à nossa página na web.

E agora podemos ver como é a estrutura básica de um arquivo HTML.

A primeira linha do documento deve ser o `<!DOCTYPE html>`, apesar de parecer um elemento HTML ela apenas diz ao navegador que ele está lidando com um arquivo do tipo HTML5. Os elementos HTML vêm logo abaixo.

`<html>`

A tag `html` é a raiz do seu documento, todos os elementos HTML devem estar dentro dela. E nela nós informamos ao navegador qual é o idioma desse nosso documento, através do atributo `lang`, para o português brasileiro usamos `pt-BR`.

`<head>`

A tag `head` contém elementos que serão lidos pelo navegador, como os metadados - um exemplo é o `charset`, que é a codificação de caracteres e a mais comum é a UTF-8, o JavaScript com a tag `script`, o CSS através das tags `style` e `link` - veremos a diferença quando falarmos sobre CSS - e o título da página com a tag `title`.

`<body>`

E dentro da tag `body` colocamos todo o conteúdo visível ao usuário: textos, imagens, vídeos.

Prática

Como exercício para esse curso iremos construir um site pessoal, e precisamos começar com a estrutura básica que acabamos de ver.

Vamos criar um arquivo `index.html` e adicionar o `doctype` e os elementos `html`, `head` e `body`.

Depois adicionaremos os elementos `meta` e `title`, no primeiro adicionamos o atributo `charset` com o valor UTF-8 para dizer ao navegador qual é a codificação dos caracteres e no segundo podemos colocar nosso nome.

E por último escreveremos nosso nome dentro do elemento body apenas para enxergarmos isso no navegador.

Semântica

Durante muitos anos o elemento padrão no HTML era a div, construíamos nosso conteúdo todo baseado nela, e assim nascia a sopa de divs.

Mas em 2014 saiu a quinta versão do HTML, e com ela vieram várias mudanças importantes, como performance e acessibilidade, mas nesse curso introdutório vamos focar na semântica.

A semântica nos permite descrever mais precisamente o nosso conteúdo, agora um bloco de texto não é apenas uma div, agora é um article e tem mais significado assim. E temos vários elementos para ressignificar as divs:

<section>

Representa uma seção genérica de conteúdo quando não houver um elemento mais específico para isso.

<header>

É o cabeçalho da página ou de uma seção da página e normalmente contém logotipos, menus, campos de busca.

<article>

Representa um conteúdo independente e de maior relevância dentro de uma página, como um post de blog, uma notícia em uma barra lateral ou um bloco de comentários. Um article pode conter outros elementos, como header, cabeçalhos, parágrafos e imagens.

<aside>

É uma seção que engloba conteúdos relacionados ao conteúdo principal, como artigos relacionados, biografia do autor e publicidade. Normalmente são representadas como barras laterais.

<footer>

Esse elemento representa o rodapé do conteúdo ou de parte dele, pois ele é aceito dentro de vários elementos, como article e section e até do body. Exemplos de conteúdo de um <footer> são informações de autor e *links* relacionados.

<h1>-<h6>

Eles não foram criados na versão 5 do HTML e nem são específicos para semântica, mas servem para esse propósito. São utilizados para marcar a importância dos títulos, sendo <h1> o mais importante e <h6> o menos. Uma dica: use apenas um <h1> por página, pois ele representa o objetivo da sua página.

Prática

Dando continuidade ao nosso site iremos montar sua estrutura. Pensei em adicionarmos um cabeçalho com nosso nome, uma lista de posts (como um blog) e um rodapé para nossos contatos.

Vamos abrir nosso arquivo index.html e começar pelo cabeçalho: criamos um <header> logo abaixo do <body> e colocamos o título da nossa página dentro de um <h1>.

Depois criaremos a lista de postagens: abrimos um elemento section e dentro dele adicionamos outro <header> contendo um <h2>. Notem que eu posso ter mais de um <header> na página.

Para criar nossa postagem adicionamos um <article> com um <header> e um <h3>.

O último passo desta etapa é criar um rodapé para nossas informações de contato: crie um elemento footer antes de fechar o </body>.

Não se preocupe com o layout e com conteúdo da página, nós vamos tratar isso mais a frente.

Textos e links

A criação do HTML foi motivada pela necessidade de compartilhar textos e documentos, e mesmo depois de quase 30 anos, com toda a evolução da web, isso ainda representa uma boa parte do conteúdo da web.

Já falamos anteriormente sobre os elementos h1-h6 e, eles são essenciais para nos indicar visualmente a importância e localização de seções de texto na página, mas para textos maiores e mais densos usamos o elemento p.

O <p> representa um parágrafo, mas ele não suporta apenas texto, podemos adicionar imagens, código, vídeos e vários outros tipos de conteúdo dentro dele.

Um outro elemento interessante e extremamente necessário na web é o <a> - que significa anchor/âncora, ele representa um hyperlink, é ele que interliga vários conteúdos e páginas na web.

O elemento a tem vários atributos, mas vamos focar em dois, o href e o target.

O href representa o *hyperlink* para onde sua âncora aponta, pode ser uma página do seu ou de outro site, um e-mail e até mesmo um telefone, os dois últimos precisam dos prefixos mailto: e tel:, respectivamente.

O target neste momento vai servir para nos ajudar a abrir nossos links em outra aba do navegador usando o valor _blank.

Prática

Vamos adicionar um texto fictício a nossa postagem: logo após o fechamento do </header> vamos adicionar um elemento p e inserir um texto que vamos retirar do site lipsum.com

E em alguma parte deste texto vamos adicionar um *hyperlink* para outra página e um para nosso e-mail.

Criarei um *hyperlink* para meu perfil no *LinkedIn*: adicione o *hyperlink* no atributo href e o valor _blank no atributo target, assim o *link* será aberto em outra aba. E em algum outro lugar do texto adicionarei meu e-mail e um link para ele, desta forma: lucas@vilaboim.com

Imagens

A web também é feita de imagens e para representá-las temos o elemento ``, ele é um daqueles elementos sem tag de fechamento.

O elemento `img` é bem simples, contendo apenas 2 atributos próprios, o `src` e o `alt`.

O `src` é obrigatório e guarda o caminho para a imagem que você quer mostrar na página.

O `alt` não é obrigatório mas é altamente recomendado por melhorar a acessibilidade, ele mostra a descrição da imagem caso ela não carregue e leitores de tela usam esse atributo para descrever a imagem para o usuário saber o que ela significa.

Prática

Vamos adicionar uma imagem ao cabeçalho da página e uma imagem a postagem.

Primeiro vamos colocar as imagens na pasta do nosso projeto. Para a imagem do cabeçalho eu escolhi uma foto minha com 100 *pixels* de largura e 100 *pixels* de altura e para a imagem da postagem eu procurei por *html code* no site Unsplash, escolhi uma das imagens e deixei ela com 960 *pixels* de largura por 322 *pixels* de altura.

Dentro do primeiro `<header>` da página e antes do `<h1>` iremos adicionar um elemento `img` e no atributo `src` colocamos o caminho para a nossa foto, `/lucas-vilaboim.jpg`, e o atributo `alt` deve conter um significado para a imagem, como no meu caso é uma ilustração, colocarei *Ilustração do rosto de Lucas Vilaboim*.

E dentro do `<header>` do `<article>` vamos fazer a mesma coisa, mas agora depois do `<h3>`, e no atributo `alt` colocaremos *Editor de texto mostrando códigos HTML*.

Listas

Os últimos elementos que veremos neste módulo são os relacionados a listas: ``, `` e ``.

Listas servem para agrupar uma coleção de itens, como uma lista de ingredientes ou, como será no nosso caso, uma lista com contatos.

O elemento `ul` cria uma lista não ordenada, onde a ordem dos elementos não é importante, e é representada com pontos, círculos ou quadrados.

O `` serve para criar lista ordenadas, nessas a ordem importa, portanto elas são representadas com números, algarismos romanos ou letras.

E o elemento `li` é um item dentro de uma dessas listas. Um `` pode conter vários tipos de conteúdos, como parágrafos, imagens e até outras listas.