

♥ Ebook Solidário

Fluência em

HTML & CSS

Aprenda a construir com menos
tempo e esforço layouts para web

DAVID ARTY





CHIEF
OF DESIGN



CHIEF
OF DESIGN

Sumário

Apresentação.....	10
Capítulo #1 - Ferramentas necessárias para o desenvolvimento.....	22
Capítulo #2 - Afinal, o que é HTML?.....	26
A evolução do HTML:	29
W3C.....	29
Metas do W3C	30
Padrões Web (web standards)	30
Tags HTML	30
Sintaxe HTML	32
Sintaxe das tags em elementos nulos (vazios ou void)	36
Tags entre tags	37
Conhecendo tags estruturais básicas em um arquivo HTML	38
Uma forma inusitada de memorizar estas tags	42
Indentação	43
Árvore do documento (hierarquia)	44
Tags mais usadas em todos os HTML(Utilizando o Princípio 80/20)	47
<div>	49
<p>	52
, e 	55
	59
	71
 	72
<h1> à <h6>	75
<a>	84

Capítulo #3 - HTML 5: a evolução!	103
As diferenças entre o XHTML e o HTML5	107
Definição das principais novas tags do HTML5	112
Diferença entre Section e Article	114
Capítulo #4 - O que é CSS?	116
Folha de	117
Estilos em Cascata	117
O que é o efeito cascata do CSS?	119
Herança	119
Especificidade	120
Importância	120
Como funciona o CSS? Parte 1 - Sintaxe	121
Como funciona o CSS? Parte 2 - Vinculando as folhas de estilos ao documento HTML	128
Inline	128
Incorporadas	129
Externas	130
Como funciona o CSS? Parte 3 - Tipos de seletores	136
Como funciona o CSS? Parte 4 - Propriedades mais usadas	143
Como funciona o CSS? Parte 4.1 - Background	148
Background-color e Background-image	149

Resumo do tópico	164
Como funciona o CSS? Parte 4.2 - Formatando textos	167
Herança	167
Color	173
Line-height	173
Font-family.....	174
Font-size	176
Font-weight	178
Font-style	179
Font-variant	180
Font	182
Text-align.....	184
Text-decoration	186
Outras propriedades para estilizar textos	188
Como funciona o CSS? Parte 4.3 - Considerações importantes	190
Como funciona o CSS? Parte 4.4 (Box Model)	204
Width	209
Height	211
Margin	211
Padding	215
Diferenças entre margin e padding	216
Border	218
Como funciona o CSS? Parte 4.5 - Outras propriedades importantes	229
Position, top, right, bottom e left.....	229

Z-index	243
List-style	249
Float	257
Clear	268
Display	272
Max-width, min-width, max-height e min-height	295
Border-radius.....	305
Box-shadow	313
Text-shadow	317
 Capítulo #5 - Introdução aoDesign Responsivo.....	322
 Capítulo #6 - Proposta para a construção de site	335
Estutura do site	336
Wireframe	337
 Capítulo #7 - Codificação do site Galucho	342
7.1 - Montar a estrutura HTML comum em todas as páginas do site dentro da página modelo	344
O corpo do site	344
O topo	345
.....	345
O menu de navegação	347
A área que abriga o logo do site	347
A área que abriga o slogan	348
O Box de conteúdo	349
O Rodapé	349
 7.2 - Estilizar, com folhas de estilo, a estrutura padrão do site para resoluções	

menores que 480 pixels	352
A “body” e o corpo do site	353
O topo	357
O Box de conteúdo	371
7.3 - Estilizar a estrutura para resoluções de no mínimo 480 pixels	378
O topo	386
7.4 - Estilizar a estrutura para resoluções de até no mínimo 767 pixels	390
O topo	390
O rodapé	396
7.5 - Estilizar a estrutura para resoluções de até no mínimo 991 pixels	398
7.6 - Salvar cópias do modelo para construção das Páginas Home e Portfólio	400
7.7 - Completar o restante dos códigos HTML da Home Page	404
7.8 - Finalizar a estilização da Home Page para resoluções menores que 480 pixels	408
Estilizando o slideshow	409
Estilizando o texto da missão do site	414
Estilizando a galeria de imagens	415
7.9 - Finalizar a estilização da Home Page para resoluções de até no mínimo 480 pixels	417
7.10 - Finalizar a estilização da Home Page para resoluções de até no mínimo 767 pixels	419
7.11 - Finalizar a estilização da Home Page para resoluções de até no mínimo 991 pixels	422

7.12 - Completar o restante dos códigos HTML da página de portfólio	425
7.13 - Finalizar a estilização da página de portfólio para resoluções menores que 480 pixels	431
7.14 - Finalizar a estilização da página de portfólio para resoluções de até no mínimo 481 pixels	437
7.15 - Finalizar a estilização da página de portfólio para resoluções de até no mínimo 767 pixels	438
7.16 - Finalizar a estilização da página de portfólio para resoluções de até no mínimo 991 pixels	439
Capítulo #8 - O mundo começa agora. Apenas começamos	441
Referências Bibliográficas.....	445
Agradecimentos	449
Sobre o Autor	450

Apresentação

Por que ler este E-book?

Forneço-te o conhecimento necessário para você criar suas primeiras páginas de internet e interpretar os códigos mais comumente usados.

Você será capaz de ver o que você planejou sendo renderizado nos navegadores e em diferentes dispositivos do que jeito que sempre quis. E ainda será capaz de executar a manutenção de sites já existentes.

E aí você me indaga: Mas como será isso? De que forma pretendes resolver meu problema? Qual o diferencial deste E-book?

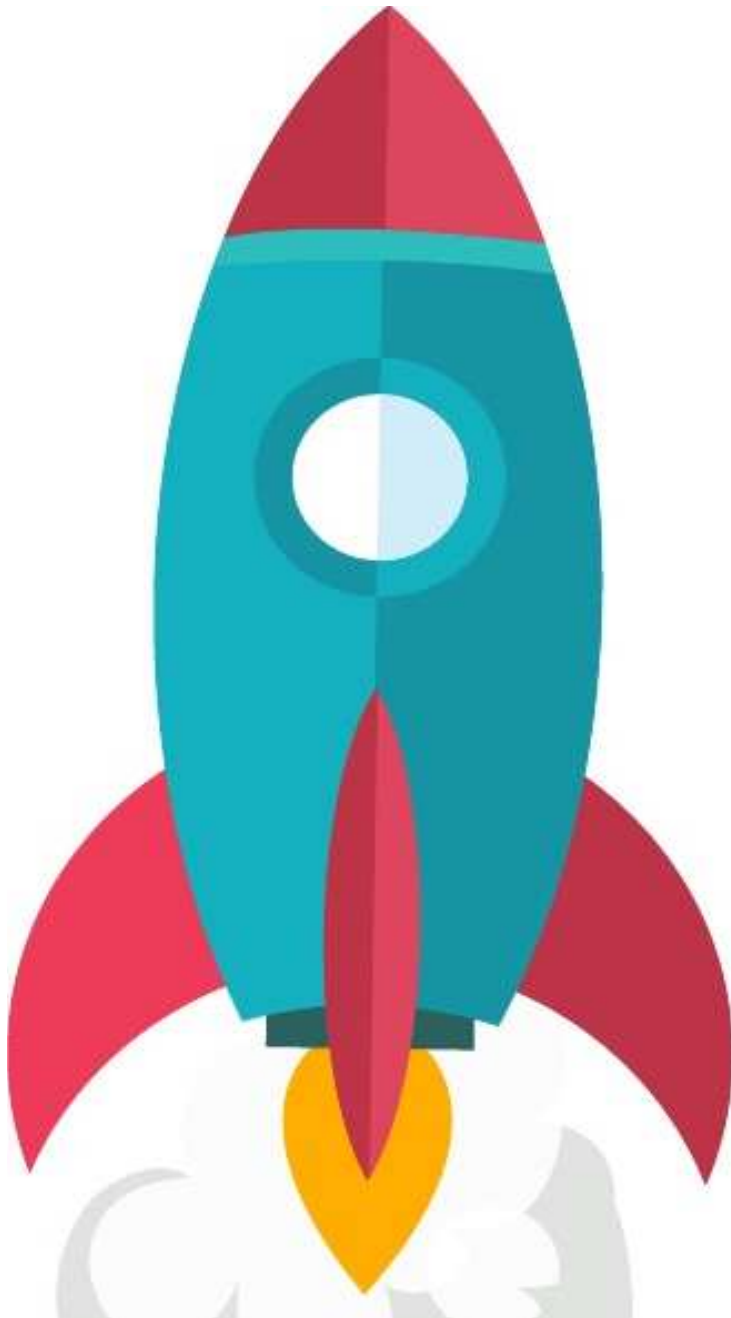
Calma! Logo, logo lhe contarei isso, porém antes quero que reflita algo sobre o comportamento humano em relação à aprendizagem e em relação à motivação.

Aprendizagem

Desde que nascemos absorvermos informações do ambiente externo, muitas delas vindas dos nossos pais e pessoas que nos cercam.

Desde bebezinhos aprendemos aos poucos a nos comunicar, com alguns erros, erros estes corrigidos com o tempo, até que chegamos à fase pré-escolar de forma que já conseguimos nos fazer entender.





Lógico que nesta fase estamos longe da pronúncia perfeita, porém já temos fluência e até então nos comunicamos facilmente sem sermos ainda apresentados à gramática, ou seja, para fazer-se entender você precisa ter fluência na Língua Portuguesa e não necessariamente ser um letrado.

Então anota aí:

Para aprender construir suas primeiras páginas de internet você precisará ter fluência nas linguagens HTML e CSS.

Mas a frente de explico como isso é possível.

Motivação

Sabemos que muitas pessoas tem facilidade em aprender linguagens para construção de sites. Isto é fato.

O problema é que esta tal “facilidade” já se tornou um mito na web que pode parecer absurdo questioná-la. Consequentemente este mito já se impregnou na mente de muitos através da comunidade web.

Isto é uma crença que cada vez mais pertence ao imaginário coletivo de quem trabalha com isso. Basta digitar no Google “HTML é fácil” e temos 167 milhões de resultados.

Fica difícil apostar que não. Concorda?

Só que na verdade muitas pessoas não tem essa tal facilidade.

dade. E como já disse acima, constatei isso ao lecionar aulas em cursos de Web design, no dia a dia, no “feedback” do público do Chief através de pesquisas, e-mails e mensagens.

Entretanto, você, com a certeza de que o mundo inteiro aprende facilmente, gasta sua energia tentando uma, duas, três e mais algumas vezes sem sucesso e cai na armadilha da desmotivação. Daí para não se passar de “burro” é preferível arrumar uma desculpa e desistir, reforçando o mito que aprender HTML é fácil.

Primeiro é que não existe essa de “burro”. Segundo é que as pessoas não são iguais. Sendo assim o que é fácil para um, pode não ser para outros. E para encerrar, se você já teve contato alguma vez com HTML e CSS, é bem provável que você teve a falsa percepção que é tudo muito complicado e o que você sentiu de verdade, na sua percepção, é que essas linguagens são um saco.

Só a verdade não é bem essa. Pelo contrário meu amigo, Galucho! Aprender HTML e CSS é muito legal, desde que a forma que for passada para você não seja tão somente uma receita de bolo mecânica e desmotivadora que ocupe sua memória, não te transmita significados e te canse a ponto de que no dia seguinte você não se lembre de nada.

Vou te mostrar que é possível enxergar e mesclar formas diferentes de entender questões. Pense comigo:

As pessoas podem ser motivadas ao aprendizado por métodos diferentes. Algumas aprendem linguagens apenas repetindo o que um tutor fornece e isso basta para que elas achem o conteúdo fácil. Já outras precisam de simbologias e significados que se relacionem com um conhecimento prévio para que o conteúdo seja absorvido facilmente.

Consegues enxergar isso?

Tendo explicado as questões sobre aprendizagem e motivação, chegou a hora de te falar o que este livro propõe e como a proposta se apresentará.

#2

Afinal, o que é HTML?

**HTML = “HyperText Markup
Language”**

OU

**Linguagem de
Marcação de Hipertexto**

HTML é uma linguagem de marcação que foi inventada na década de 1990 por um cientista chamado Tim Berners-Lee.

A finalidade inicial era tornar possível o acesso e a troca de informações, de documentações e de pesquisas, entre cientistas de diferentes universidades.

O projeto inicial tornou-se um sucesso jamais imaginado por Tim Berners-Lee. Ao inventar o HTML ele lançou as fundações da Internet tal como a conhecemos atualmente.

O HTML se popularizou através do primeiro browser da história: O Mosaic.

Mosaic - browser desenvolvido por Marc Andreessen na década de 1990. Desde então vários desenvolvedores e fabricantes de browsers surgiram utilizando o HTML como base.

O HTML é uma linguagem universal na internet para distribuir informação e conteúdo. Ele é baseado no conceito de Hipertexto, ou seja, ligações/conexões entre elementos como imagens, documentos, vídeos, áudios, palavras, etc.

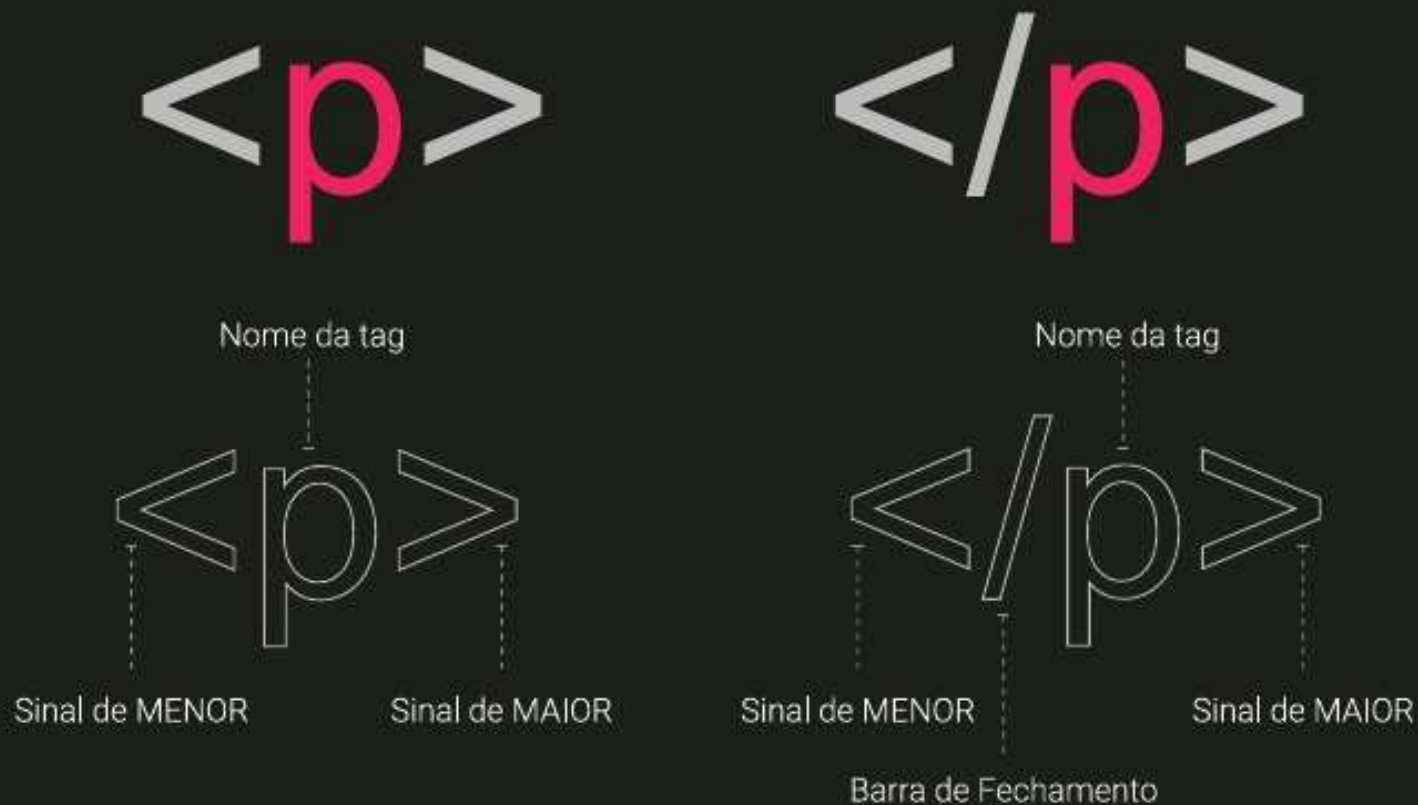
O hipertexto é uma forma não linear de distribuir e dispor os conteúdos no meio digital.

Aquilo que você vê quando abre uma página na Internet é a interpretação que seu navegador faz do HTML.

O HTML é a linguagem que usamos para publicarmos conteúdo (texto, imagem, vídeo, áudio e etc) na Web.

Sintaxe HTML

Já sabemos que todo elemento HTML é formado por tags. As tags tem o seguinte formato:



Começam com um sinal de menor "<", terminam com um sinal de maior ">" e entre esses sinais vai o nome da tag do elemento que queremos inserir em nosso documento.

Para constituir o elemento HTML precisamos de dois tipos de tags mais o conteúdo. Os dois tipos de tags são:

- A tag de abertura que marca o início do elemento: <tag>.
- E a tag de fechamento que marca o fim do elemento: </tag>.

TAG de Abertura

<p>

TAG de Fechamento

</p>

A diferença entre elas é que na tag de fechamento existe uma barra "/".

Tudo que estiver contido entre uma tag de abertura e uma tag de fechamento representa o conteúdo. E este será processado segundo o comando contido na tag.

Para exemplificar vamos usar o elemento HTML que marca parágrafos.

Começamos com a tag de abertura:

#3

HTML 5: a evolução!

HTML



Já sabemos que o HTML apresenta o conteúdo e usa url para vincular arquivos externos como imagens, vídeos, páginas, etc.

Agora vamos um pouco mais adiante e falaremos do HTML5.

Provavelmente você deve ter lido e/ou ouvido falar dele.

O HTML 5 vem sendo desenvolvido e planejado há bastante tempo, praticamente desde 2004 quando começaram os primeiros testes.

O HTML5 é a nova e atual versão do HTML reconhecida pelo W3C. Ele proporcionou inúmeras possibilidades e novas e melhores definições.

Somado ao CSS 3 mais JavaScript, essa nova linguagem possibilitou a criação de coisas que antes só poderíamos fazer com outras linguagens ou métodos.

Novas tags surgiram, novos atributos e a possibilidade de páginas mais interativas, leves, acessíveis e semânticas.

#4

O que é CSS?

**CSS = “Cascading Style
Sheets”**

OU

**Folha de
Estilos em Cascata**

CSS é uma linguagem de estilos que define o layout de documentos escritos com linguagem de marcação, no caso o nosso HTML.

Ele adiciona um estilo a uma página e altera a forma visual de como ela é apresentada.

O CSS foi uma revolução no design para web. Ele executa:

- controle do layout de vários documentos a partir de uma simples folha de estilos;
- controle preciso do layout;
- aplicação de diferentes layouts para diferentes mídias (tela, impressora, celular, etc.);
- avançadas técnicas de desenvolvimento;
- separa a estrutura do design - agilidade e performance.

O CSS é o responsável pelo visual de um site. Através dele aplicamos as cores, determinamos os tamanhos, os pesos (no caso de fontes), bordas, imagens, etc. Com isso, separa-se a linguagem de marcação (HTML) da camada de apresentação (estética, design) do site.

E essa alteração aparecerá em todo o site. E isso independentemente se o meu site tem 1 ou 100 documentos HTML.

Em versões anteriores do HTML existiam elementos que eram específicos para apresentação, como por exemplo, ``, `<i>`, ``, `<center>`, `<big>`, etc. Eles eram utilizados para adicionar certas características visuais ao documento.

Mas imagine que você tem uma tarefa de criar uma página com 50 subtítulos. E lhe foi determinado aplicar a cor azul a todos eles. Logo, para deixá-los com a cor azul você usa ``, aplicando manualmente, em todos os subtítulos da página.

Porém na manhã seguinte seu chefe pede para que você mude todos os subtítulos para a cor vermelha. Sem opção, (claro que você poderia pedir demissão, hehehe) você teria refazer tudo novamente, perdendo bastante tempo nessa refação indo em cada tag para mudar a cor.

Como funciona o CSS? Parte 1 - Sintaxe

O CSS tem a sua própria sintaxe e estrutura. Uma regra CSS é composta por um seletor, uma propriedade e um valor. Vamos conhecê-la?

Logo, para criar uma declaração CSS, seguimos os seguintes passos: Apontamos o seletor;

seletor { propriedade: valor; }

- Abrimos e fechamos chaves {};
- Dentro de chaves determinamos a propriedade e o valor;
- E caso precise aplicar mais de uma declaração (propriedade + valor) utilizamos ponto e vírgula (;) ao final da declaração (após o valor), para separá-las.

Seletor : é o elemento HTML identificado por seu nome, classe, ID, etc; e para o qual a regra será aplicada (por exemplo: p, h1, form, .minhaclasse, :hover, etc.). O seletor funciona como se fosse um "alvo". Eu aponto para o CSS qual tag ele aplicará o estilo que determinei, na regra CSS.

Propriedade : é a característica que será estilizada, de um elemento HTML , no CSS. Existem inúmeras propriedades CSS, por exemplo: font, color, background, etc.

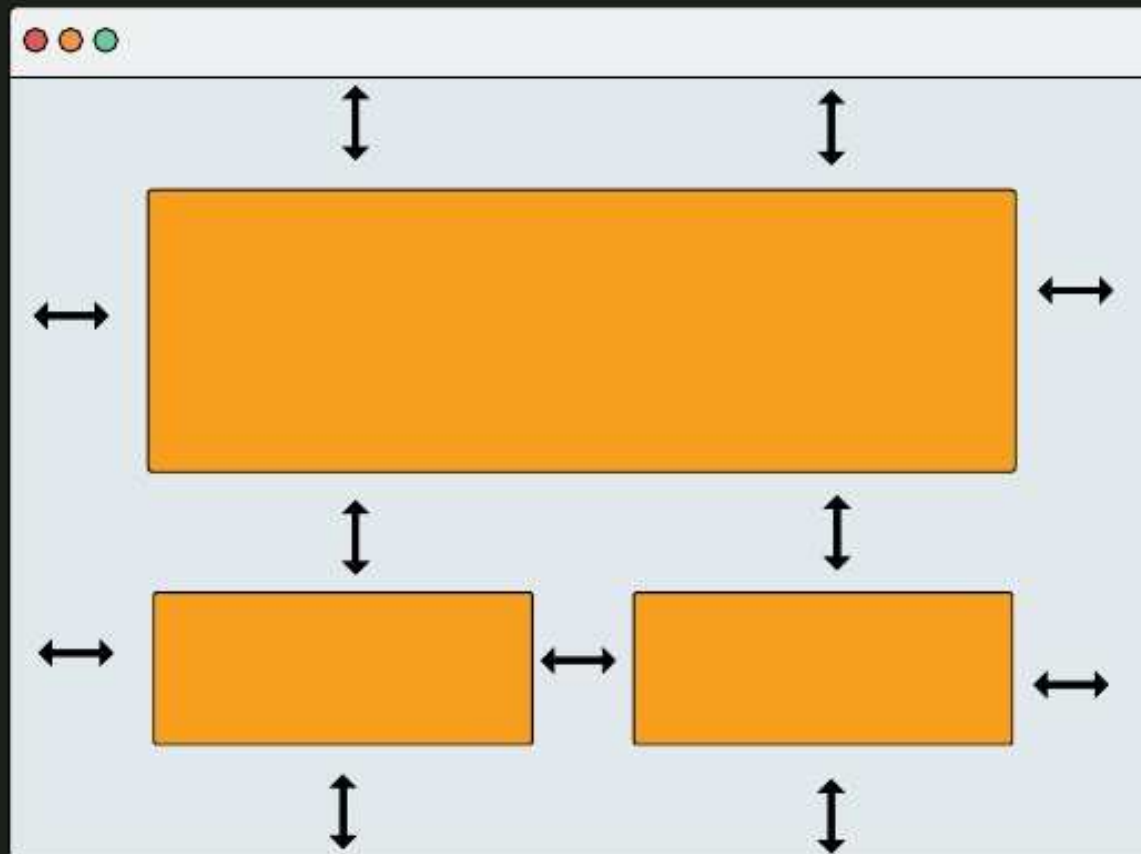
Valor : é a aparência que será assumida pela propriedade CSS . Então, a propriedade define o que iremos estilizar enquanto que o valor define como ou quanto será estilizado.

Por exemplo: Eu quero determinar que o tamanho de um subtítulo, do meu documento HTML, será 14px. A propriedade para mudar o tamanho da fonte é font-size e o valor da propriedade seria 14px. Com isso teríamos o seguinte resultado:

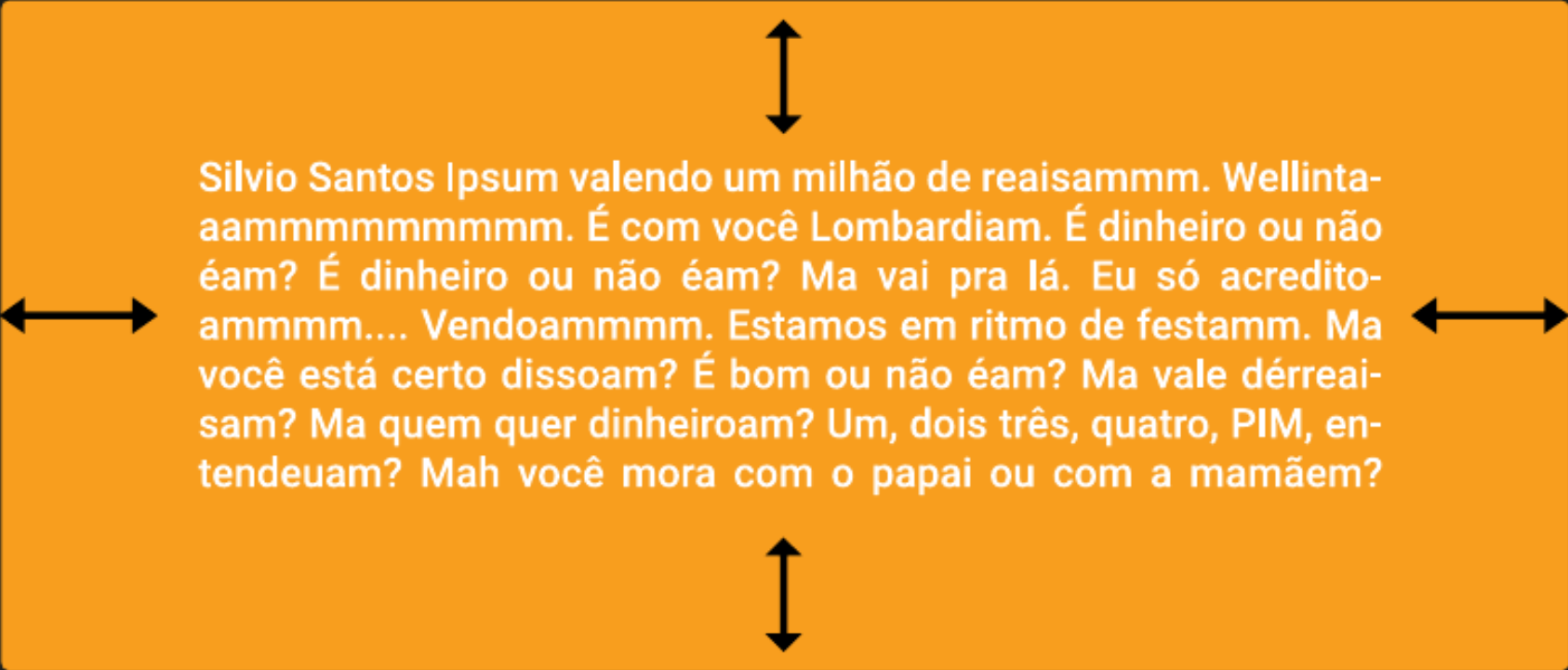
Diferenças entre margin e padding

Quando estamos aprendendo a desenvolver é normal confundirmos estas duas propriedades. Porém com o tempo você verá que são muito diferentes. Para diferenciar tente se lembrar que para afastar boxes vizinhos você aplica margin. E para afastar o conteúdo seja texto, imagem, ou outro, da extremidade do Box, use padding.

A propriedade "margin" faz um Box se distanciar de blocos vizinhos e também da extremidade do navegador.



Enquanto que a propriedade "padding" faz o conteúdo dentro do bloco se distanciar da sua extremidade.



Silvio Santos Ipsum valendo um milhão de reaisammm. Wellintaammmmmmmmm. É com você Lombardiam. É dinheiro ou não éam? É dinheiro ou não éam? Ma vai pra lá. Eu só acreditoammm.... Vendoammm. Estamos em ritmo de festamm. Ma você está certo dissoam? É bom ou não éam? Ma vale dérreaisam? Ma quem quer dinheiroam? Um, dois três, quatro, PIM, entendeuam? Mah você mora com o papai ou com a mamãem?

Nota: Quando utilizamos o valor zero para as propriedades margin e padding no seletor universal (*) estamos "limpando" os valores em elementos que já vem padronizados pelos navegadores.

E fazemos isso, pois "resetando" podemos estabelecer os valores da forma que a gente bem entender sem entrar em conflito com valores criados pelos navegadores, sacou?



#5

Introdução ao Design Responsivo

Um site projetado com media queries estabelece diferentes estilizações para diferentes tamanhos de tela. Quando este site é acessado por um dispositivo as medias queries fazem algumas consultas ao navegador:

“ E ae navegador? Qual sua largura? Altura? Resolução? Que time que você torce? ”
(Não! Esta ultima não! Hahaha)

Após esta checagem com o navegador ele compara as informações que tem declaradas em seus parâmetros e retorna os estilos que se adaptam melhor as informações passadas pelo navegador.

Vamos explicar melhor isso com os códigos. Para usar as media queries você precisa saber as resoluções de tela (altura e largura em pixels) dos dispositivos mais comuns. Usamos estas larguras para estabelecer “breakpoints” que são pontos de quebras que determinam quando devemos alterar as regras CSS.

Hoje em dia esta cada vez mais difícil saber o quanto de resoluções diferentes existem, porque toda hora se lança um novo aparelho e cada vez mais parrudo. Mas para fins de estudos vamos nos ater a estas larguras a seguir como breakpoints:

- **320 pixels**
- **480 pixels**
- **600 pixels**
- **768 pixels**
- **1024 pixels**
- **1200 pixels**

Teoricamente os dois valores abaixo atendem bem a alguns smartphones:

320 pixels
480 pixels

#6

**Proposta para a
construção de site**

Chegamos em um estágio onde é fundamental repassar o conteúdo fornecido em um contexto mais próximo da realidade que você irá encontrar.

Por isso propomos a construção de um site onde usaremos seus conhecimentos prévios e recém adquiridos sobre HTML, juntamente com os tópicos sobre CSS que foram apresentados e incorporados ao seu conhecimento, conforme o site for produzido.

Você perceberá que o esquema 80/20 faz sentido. Porém, eventualmente entrará um ou outro elemento novo que foge ao esquema, mas que aumentará seu leque de conhecimento.

O site será um portfólio fictício do Designer "Galucho". E contará com elementos suficientes para divulgação dos seus trabalhos como uma página inicial, uma área de apresentação dos trabalhos e outra para dados de contato e integração com as redes sociais.

Esse seria o "briefing básico" para o nosso projeto já que não aprofundaremos em questões como objetivo de comunicação e de marketing, público alvo, estética, entre outras.

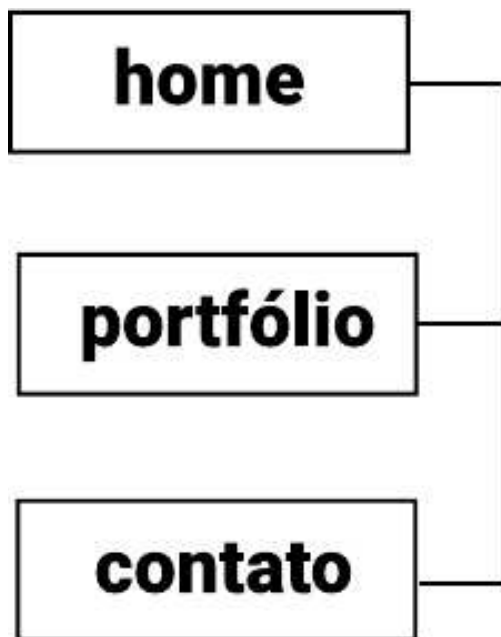
Reforçamos que este é um livro sobre HTML/CSS. Porém para termos um site funcional e visualmente agradável atentaremos a alguns aspectos que garantem a usabilidade, navegabilidade e uma boa disposição dos elementos da interface.

Você sempre pensará em todos os aspectos do Design, entretanto para esses exercícios adotaremos um template que foi reformulado para este fim didático e de acordo com os aspectos citados acima.

Alguns profissionais projetam suas interfaces direto no código. Sugerimos que você defina a estrutura de páginas e navegação, crie um esboço, depois transporte as ideias para seu editor visual favorito (Photoshop e Fireworks são indicados) para depois transportar o layout para codificação.

Veremos a hierarquia do site através de um diagrama e depois o Wireframe do nosso site.

Estrutura do site



Antes de começar a um criar site, é importante pensar como será a estrutura dele (qual o caminho de tal página, qual ação do usuário o levará para tal página, onde ficará cada diretório, definir a hierarquia do conteúdo, etc).

Para ajudar nessa tarefa, podemos usar um diagrama simples. Você pode criar um diagrama para definir a hierarquia e organização do seu do seu site.

Você pode fazer com lápis e papel, em um quadro branco usando postit ou em softwares e serviços online.

O importante é você planejar a sequência lógica do seu site, estruturando-o da melhor maneira possível.

Reservar um tempo, antes de sair fazendo as coisas no Photoshop, para planejar e estruturar o seu site, mesmo que ele seja simples, pode te facilitar a vida tanto na criação quanto no desenvolvimento das páginas.

Você pode achar essa parte chata e trivial, mas com certeza se algo der errado durante a execução do projeto, você lembrará dela. Não que ela seja um “escudo anti-bugs”, não é isso, mas um bom planejamento pode facilitar a sua vida te ajudando a “encontrar possíveis erros” antes mesmo deles se apresentarem.

Então antes de sair fazendo, pare e pense no seu projeto como um todo!



**Codificação do
site Galucho**

Agora começaremos a montar efetivamente o site do Galucho com os códigos HTML e CSS. A primeira etapa é construir uma página modelo que possuirá somente os elementos comuns entre todas as páginas. Por exemplo: o topo, com a integração das redes sociais, o menu e o slogan, e também o rodapé, com as informações de contato.

Essa página modelo não será usada no site, entretanto ela serve de modelo para construir as outras páginas do site. O intuito é duplicá-la para não perdermos tempo construindo os mesmos elementos em cada página. Sacou? Com isso bastará a você incluir em cada página os elementos que não são comuns entre as páginas. Você ganhará muito tempo com isso.

Separaremos o desenvolvimento nas seguintes etapas:

1. Montar a estrutura HTML comum em todas as páginas do site dentro da página modelo;
2. Estilizar, com folhas de estilo, esta estrutura para resoluções menores que 480 pixels;
3. Estilizar a estrutura para resoluções de até no mínimo 480 pixels;
4. Estilizar a estrutura para resoluções de até no mínimo 767 pixels;
5. Estilizar a estrutura para resoluções de até no mínimo 991 pixels;
6. Salvar cópias do modelo para construção da Home Page e da página de portfólio;
7. Completar o restante dos códigos HTML da Home Page;
8. Finalizar a estilização da Home Page para resoluções menores que 480 pixels;
9. Finalizar a estilização da Home Page para resoluções de até no mínimo 480 pixels;
10. Finalizar a estilização da Home Page para resoluções de até no mínimo 767 pixels;
11. Finalizar a estilização da Home Page para resoluções de até no mínimo 991 pixels;
12. Completar o restante dos códigos HTML da página de portfólio;
13. Finalizar a estilização da página de portfólio para resoluções menores que 480 pixels;
14. Finalizar a estilização da página de portfólio para resoluções de até no mínimo 480 pixels;
15. Finalizar a estilização da página de portfólio para resoluções de até no mínimo 767 pixels;
16. Finalizar a estilização da página de portfólio para resoluções de até no mínimo 991 pixels;

Então , vamos começar! Vamos nessa? :)