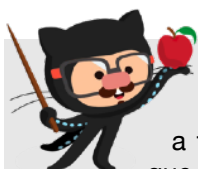


# DevWeb

## Capítulo 06

# Imagens e Favicon

Sabe aquela frase clássica que diz que “uma imagem vale mais do que mil palavras”? Pois é exatamente assim com a maioria dos sites. Tirando algumas pequenas exceções onde o público acessa um site apenas para “ler” seu conteúdo, as imagens são essenciais para a maioria dos casos. São elas que chamam a atenção e devem ser escolhidas com muita dedicação. Vamos nessa?



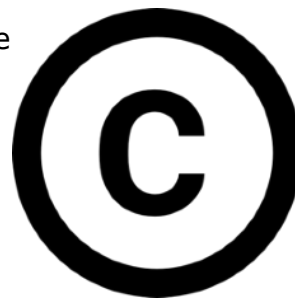
Você tem todo o direito de usar esse material para seu próprio aprendizado. Professores também podem ter acesso a todo o conteúdo e usá-los com seus alunos. Porém todos o que usarem esse material - seja para qual for a finalidade - deverão manter a referência ao material original, criado pelo **Prof. Gustavo Guanabara** e disponível no endereço do seu repositório público <https://github.com/gustavoguanabara/>. Este material não poderá ser utilizado em nenhuma hipótese para ser replicada - integral ou parcialmente - por autores/editoras para criar livros ou apostilas, com finalidade de obter ganho financeiro com ele.



# Pra começar, vamos falar de direitos autorais

Sabe aquela foto que você pegou lá no **Google Imagens** e se encaixou perfeitamente no seu site? Pois é bastante provável que você simplesmente **NÃO POSSA USÁ-LA** e pode até mesmo ser processado(a) por isso.

" - Mas como assim, Guanabara!?" - você pode estar gritando, mesmo que apenas mentalmente, nesse exato momento.



É exatamente o que eu falei, pequeno Gafanhoto! Na verdade não só as imagens, todo conteúdo disponível na Internet pode estar protegido por direitos autorais. Livros em PDF, textos em blogs, matérias em jornais online, fotos, animações, vídeos. Tudo pode ser protegido e você nem fazia ideia disso.



**ANTES CONTINUAR:** Eu não sou profissional do Direito, então faço uma pausa aqui no material para te indicar um vídeo onde temos o Especialista em Direito Digital **Alan Moreira Lopes** falando sobre esse assunto com muito mais propriedade. A partir de 29min ele responde umas perguntas bem comuns.

Canal Revolution: <https://youtu.be/Bkym20Gq0oQ?t=171>

E se nesse momento você está pensando: "*mas quem vai processar um jovem aluno que está aprendendo a criar sites usando imagens do Google?*".

Minha maior preocupação aqui não é te **PROIBIR** de pegar imagens no Google Imagens e usar nos seus sites enquanto você está aprendendo HTML. O meu medo é você achar que esse ato é **100% LEGAL** e comece a criar sites profissionalmente para empresas usando imagens e conteúdos que você não pode.



**OLHA A TRETA:** Um caso que ficou muito famoso na Internet foi com o canal **Nostalgia**. O Felipe Castanhari, dono do canal, vive recebendo avisos de direitos autorais por usar pedaços de vídeos de documentários, filmes, clipes e séries com direito autoral.

Canal Nostalgia: <https://youtu.be/SJacdAbjdZI>

Concordando ou não com isso, o fato é que a lei existe. E ir contra a lei é ficar totalmente vulnerável aos efeitos causados por ela. Eu até sugiro que façamos uma breve discussão sobre isso em sala de aula. Se eu não for seu professor, bata um papo com ele sobre o assunto. Também sugiro que você converse sobre isso com algum advogado especializado nesse ramo digital caso fique alguns dúvida (toda família tem um tio ou vizinho advogado).

Sendo assim, para que você seja profissionalmente capaz de criar sites com imagens que não vão gerar processos, existem quatro maneiras simples:

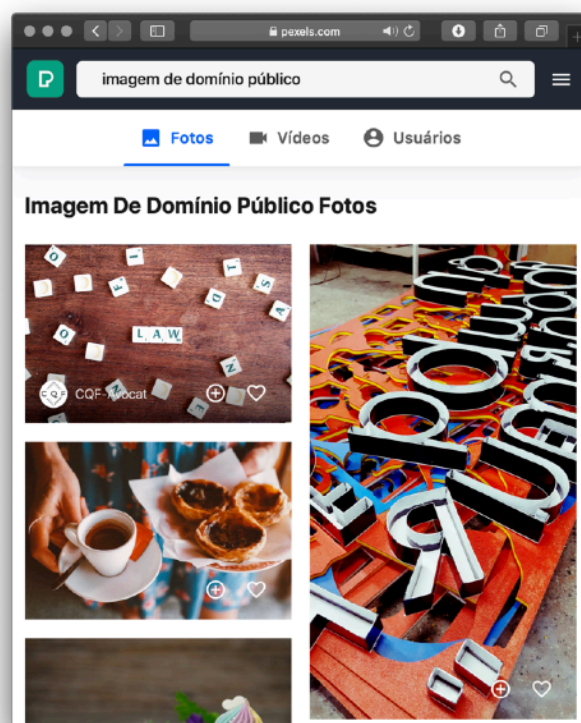
1. Seja o criador da própria arte. Use fotos que seu próprio cliente tirou ou que você mesmo criou.
2. Consiga uma autorização legal de quem é o detentor dos direitos autorais. Isso precisa de documentos específicos para tal.
3. Compre o direito de usar a imagem. Existem vários sites como o **iStockPhoto**, **DepositPhotos**, **Shutterstock**, etc. Só fica de olho aberto e leia as regras, pois na maioria dos casos essa imagem que você comprou só pode ser usada uma única vez ou em um único projeto.
4. Use imagens de domínio público, pois essas você tem a liberdade de colocar no seu site, seguindo as regras estabelecidas pelo autor.

## E onde encontro imagens de domínio público?

Existem vários sites que oferecem imagens de domínio público para serem usados em seus projetos. Mas a dica que te dei antes continua valendo: leia atentamente as instruções para uso dessas imagens. Algumas delas deixam você usar a arte apenas se o artista for creditado em um texto localizado no seu site.

Alguns dos sites que eu mais gosto de visitar quando quero usar imagens com domínio público são:

- [UnSplash](#)
- [Pexels](#)
- [FreePik](#)
- [Rawpixel](#)
- [Pixabay](#)
- [Libreshot](#)
- [Wikimedia Commons](#)



Mas muito **CUIDADO!** Alguns desses sites que disponibilizam imagens de domínio público possuem propagandas de serviços de venda de imagens e acabam te confundindo um pouco.

Vamos mudar de assunto, já que possivelmente você não está se preocupando muito com esse papo de direitos autorais, mesmo ele sendo muito importante para quando você virar um profissional de verdade.

# Vamos falar sobre formatos de Imagem

Existem vários formatos de imagem, cada um com suas características, vantagens e desvantagens. Porém, vamos nos focar aqui nos dois formatos compactados mais usados para a criação de sites: **JPEG** e **PNG**.

O algoritmo de compactação **JPEG** é usado para gerar imagens fotográficas com um tamanho extremamente reduzido. Ele foi criado em 1983 por **Eric Hamilton** e hoje é gerenciado pelo **Joint Photographic Experts Group**. Ele é amplamente utilizado por câmeras digitais modernas e programas de tratamento de imagens.

A grande vantagem do uso de arquivos **JPG** (em formato JPEG) é gerar arquivos muito pequenos e que ocupam pouco espaço em disco. Isso é muito importante, pois quando colocarmos nosso site no ar, ele tem que ser leve e carregar as imagens muito rapidamente. Só toma cuidado para não exagerar na hora de configurar o nível de compactação. Isso pode fazer com que sua imagem fique horrível e toda borrada (dá só uma olhada na imagem abaixo).

O formato **Portable Network Graphics** (PNG) surgiu em 1996, desenvolvido pelo **W3C** (o mesmo órgão que gerencia a linguagem HTML) com o objetivo de substituir o formato **GIF** (que hoje voltou a ser popular graças ao WhatsApp e Instagram). Ele também é um formato compactado, mas não tanto quanto o JPEG.

A principal característica do **PNG** - e que o diferencia do JPEG - é a capacidade de configurar a opacidade de cada pixel (deixá-lo transparente ou com transparência limitada).

Mas para entendermos melhor a explicação acima, vamos a um exemplo visual (já que estamos falando de imagens né?)



As duas primeiras imagens estão compactadas no formato JPEG, mas na primeira eu coloquei o nível de compactação em qualidade 5% e ficou com 20KB e na segunda uma qualidade de 30% e ficou com 120KB. A última imagem está comprimida no

formato PNG, ficou com 300KB e teve o contorno preservado por causa do fundo transparente.

De forma resumida, na hora de escolher o formato de imagem para o seu site, opte sempre pelo formato JPEG com uma compactação entre 30% e 50%. O formato PNG só deverá ser usado quando precisarmos de transparência na foto. Combinado?

## Faltou falar sobre tamanho das imagens

Quando você está começando a desenvolver sites, acaba pensando que para obter o melhor resultado, sua imagem deve ter um tamanho **GIGANTE** para poder ter a maior qualidade.

Realmente, uma imagem com resolução grande (1920x1080) tem mais pontos que uma imagem com resolução pequena (500x280). Só que uma imagem 1920x1080 pode gerar um arquivo de até 3MB, enquanto uma 500x280 dificilmente vai passar de 500KB.



A regra de ouro nesses casos é: **use imagens do tamanho certo!** Vai precisar de uma imagem que vai ter 200 pixels de largura? Gere um arquivo exatamente com esse tamanho! Nada de ficar salvando arquivos gigantes e diminuindo o tamanho da imagem com códigos.

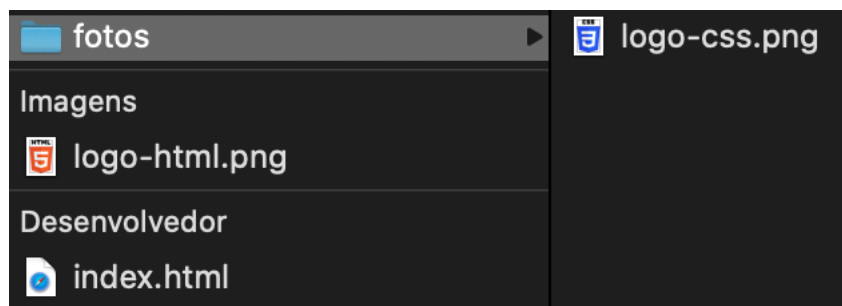
Uma imagem 1920x1080 de 3MB não vai ficar mais leve se você mudar a largura dela no seu CSS. Muito pelo contrário. Seu navegador vai levar um tempão pra carregar o arquivo do seu servidor e vai exibir a imagem minúscula na tela.

E no que isso te prejudica? Os mecanismos de busca como o Google **PENALIZAM** sites lentos e pesados, retirando-os da primeira página de buscas. Quem aqui quer criar um site que não aparece nos resultados do Google?

## Como carregar uma imagem em HTML

Agora que você está especialista em formatos e tamanhos de imagens, vamos gravar algum arquivo em uma pasta chamada **ex003** que você vai criar no seu repositório local. Se você visitar meu repositório em <http://gustavoguanabara.github.io/> e for até a parte de exercícios de HTML, vai ver que na pasta ex003 existem os arquivos logo-html.png e logo-css.png. Faça o download deles pra você ou salve outra imagem qualquer na sua pasta. Só fica de olho no formato dela: JPG ou PNG. Use o VScode para criar também o arquivo index.html e coloque dentro da pasta **ex003**, conforme vou apresentar a seguir.

Eu só te faço um alerta: analise a foto a seguir, onde eu mostro que o arquivo logo-html.png estará na pasta raiz do projeto (mesmo local onde está o arquivo index.html), enquanto o arquivo logo-css.png está localizado dentro da pasta fotos que está na mesma pasta do projeto.



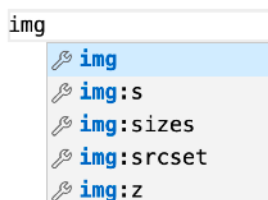
*Dentro da pasta ex003, crie os arquivos acima*

A posição dos seus arquivos vai fazer diferença e você deverá adaptar seu código para poder carregar as imagens sem problemas. Agora vá até o Visual Studio Code e edite o arquivo index.html, criando o código base de sempre e atualizando conforme o que apresento a seguir:

```
8 <body>
9     <h1>Testando carga de Imagens</h1>
10    <p>Abaixo você vai ver uma imagem que está na mesma pasta.</p>
11    
12    <p>Podemos também carregar imagens que estão em outra pasta, contanto
    que ela esteja dentro da pasta atual.</p>
13    
14    <p>Também podemos carregar imagens externas, e para isso devemos ter a
    sua URL completa.</p>
15    
16 </body>
```

O código acima só está começando na **linha 8** por questões de praticidade, pois as **linhas de 1 até 7** são aquelas criadas automaticamente pelo VScode. Não apague essas linhas, apenas vamos analisar o código do <body>.

Na **linha 11**, adicionamos a tag <img>, responsável por carregar imagens que estão na pasta do projeto atual ou em links externos. Essa tag tem como parâmetros básicos o src (que vem de *source*, origem) e alt (que vem de *alternative* ou texto alternativo).

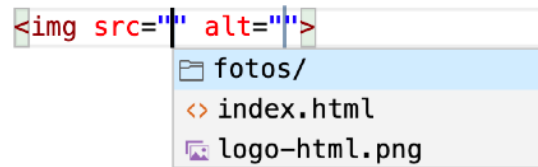


Aqui vai uma **DICA IMPORTANTE**: quando for carregar uma imagem no Visual Studio Code, não digite a tag <img> com os colchetes angulares. No lugar disso, digite apenas img e o próprio editor vai sugerir a inserção da imagem (veja foto ao lado).

Agora pressione a tecla **Enter**. Automaticamente, uma tag padrão será preenchida no seu código, incluindo as propriedades básicas src e alt. Mas a dica não para por aí. Sigamos.



Depois do código aparecer magicamente, o cursor do teclado vai aparecer dentro das aspas do parâmetro `src`. Pressione **Ctrl+Espaço** e mais uma mágica vai acontecer. Como a imagem ao lado mostra, o editor vai te mostrar uma listagem com todos os arquivos disponíveis na pasta do seu projeto. Escolha o arquivo clicando em `logo-html.png` para que você não precise digitar o nome dele. Isso evita muitas falhas causadas por erros de digitação!



Agora volte para a página anterior e analise as linhas 13 e 15. A **linha 13** vai carregar a imagem `logo-css.png` que está dentro da pasta `fotos`, como vimos anteriormente. Note que foi necessário colocar o nome da pasta seguido de uma barra antes de colocar o nome do arquivo. Se isso não for obedecido, simplesmente a imagem não vai carregar!

Por fim, na **linha 15** fizemos a carga de um arquivo que não estava na pasta do projeto. A imagem `js-small.gif` está localizada em outro domínio, o `jsdotgit.com`. Nesse caso, precisamos indicar o caminho completo (**URL**) da imagem.

Você pode obter a URL completa de qualquer imagem, abra um site no Google Chrome, clique com o botão direito do mouse sobre a imagem e escolha a opção *"Copiar endereço da imagem"*. Aí é só voltar no seu código HTML, colocar o cursor do teclado dentro das aspas da propriedade `src` da sua imagem e apertar **Ctrl+V**. Faça uma experiência em casa agora mesmo.

## Para que servem os textos alternativos?

Toda imagem deve ter um texto alternativo, mesmo você achando que isso é muito chato de fazer. Textos alternativos ajudam muito na indexação do seu site em mecanismos de busca e também ajudam muito na **Acessibilidade**, pois se um visitante for deficiente visual, seu navegador vai ser capaz de descrever que tipo de foto está sendo mostrada ali.

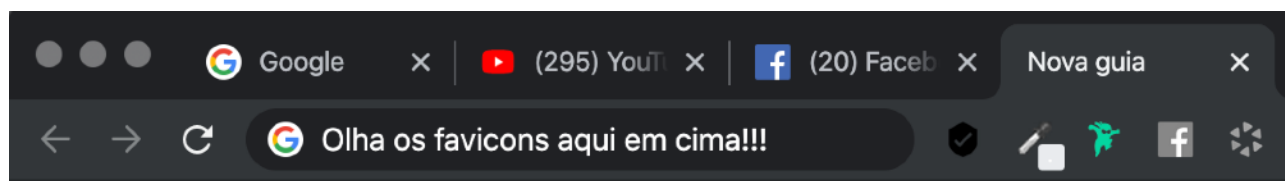
Tente ser objetivo na descrição da sua imagem, mas crie textos que fazem sentido. Tem um monte de gente que coloca o texto "foto" como descrição de uma imagem. Aí não dá!

Os textos alternativos também auxiliam o Google a saber o que tem dentro da sua foto e exibi-la nos resultados de busca do **Google Imagens**.

## Usando Ícone de Favoritos (favicon)

Você já ouviu falar em **favicon**? Talvez você não os identifique pelo nome esquisito, mas com certeza já viu aqueles pequenos ícones que aparecem ao lado dos sites que visitamos, na parte superior do seu navegador. É o mesmo local onde aparecem o texto que você colocou na tag `<title>` do seu site.

Na imagem abaixo, você consegue ver os favicons do Google, YouTube e Facebook. A última guia não tem favicon, porque não tem site nenhum aberto nela.

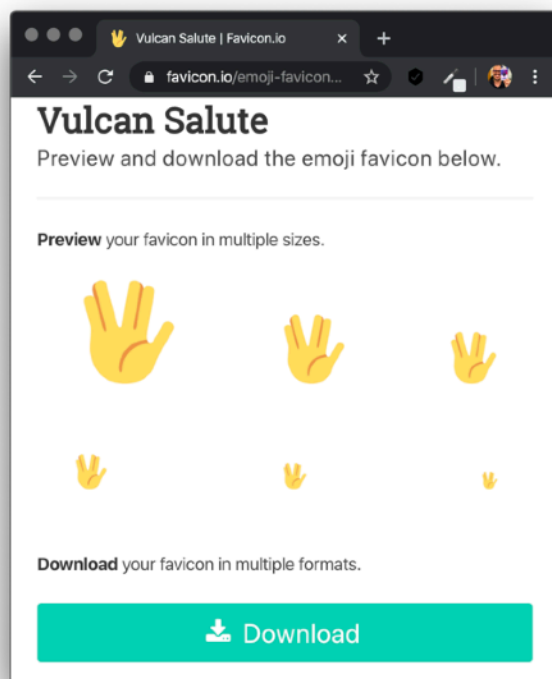


Para usar um favicon no seu site, você precisa ter o arquivo do ícone, que geralmente está no formato **ICO**. Se quiser baixar alguns ícones prontos, recomendo o site [IconArchive](https://iconarchive.com).

Se você tiver o dom e muita paciência, pode desenhar seu favicon no site [favicon.cc](https://favicon.cc).

Agora, se seu objetivo é criar um ícone personalizado para seu site baseado em imagens que você já tem, recomendo o site [favicon.io](https://favicon.io), onde você pode submeter qualquer imagem e baixar um pacote com vários formatos de favicon. Lá você também vai poder criar um ícone de favoritos a partir de um texto ou então baseado em um emoji (veja foto ao lado).

Ao baixar o pacote com os ícones de favoritos, descompacte-os (geralmente é um arquivo ZIP), pegue o arquivo com extensão **.ICO** e coloque na pasta principal do seu projeto. Normalmente, colocamos o nome do arquivo como **favicon.ico**.



Agora que você já tem seu favicon salvo na pasta do seu projeto (recomendo criar o **ex004** no seu repositório local), vamos nos focar no código HTML para carregá-lo.

Crie seu arquivo **index.html** dentro da pasta **ex004**, incluindo seu código base automaticamente. Depois faça uma simples alteração na sua área **<head>**, incluindo uma tag **<link>**, digitando apenas a palavra **link** em uma linha em branco e escolhendo a opção **link:favicon**.

```
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <link rel="shortcut icon" href="favicon.ico" type="image/x-icon">
7   <title>Testando o ícone de Favoritos</title>
8 </head>
```

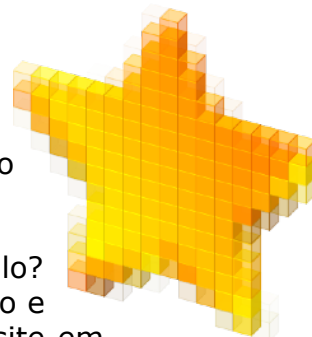
Pronto! É só dar uma olhada no nosso código acima e prestar atenção na **linha 6**. Se você usou a dica que dei acima, não precisou digitar mais nada! Todo o código foi feito automaticamente!



**OBS:** se seu arquivo não se chamar `favicon.ico`, você deve adaptar a linha 6, colocando o nome do seu ícone.

## Quais são os formatos aceitos para Favicon?

Apesar da minha indicação para usar o arquivo no formato **ICO**, existem outros formatos suportados para o seu favicon, como o próprio **PNG** que estudamos anteriormente e até o formato **SVG**, que é vetorizado e seria a melhor das opções.



Mas se o SVG é a melhor das opções, por que não usá-lo? Infelizmente nem todo navegador é compatível com esse formato e acaba causando algumas inconsistências na exibição do nosso site em outras plataformas.

Sendo assim, pelo menos por enquanto, continuo dando o meu conselho: use o formato ICO.

## Vamos aos desafios!

Lá no repositório, além do material em PDF e dos códigos dos exercícios 100% disponíveis, também disponibilizamos alguns **desafios** que devem ser resolvidos. Esses desafios não incluem o código original e você deve tentar chegar à resposta sem copiar nenhum código.



Com todo o conteúdo que vimos até essa aula, você já pode resolver do **desafio d001** até o **desafio003**. Acesse o repositório público, abra a área do curso de HTML+CSS e clique no link de acesso aos desafios. Manda ver! Só não fica pedindo a resposta! Você consegue resolver isso sozinho(a)!

Repositório em: <https://gustavoguanabara.github.io>

## Eu já falei sobre isso no YouTube?

Eu sei que às vezes as pessoas gostam mais de assistir vídeos do que ler livros, e é por isso que eu lanço há anos materiais no canal Curso em Vídeo no YouTube. O link que vou compartilhar contigo tem o conteúdo explicado como você leu aqui, só que de forma mais ilustrada. Reserve um tempo dos seus estudos para assistir esse vídeo todo.



Curso em Vídeo: [https://www.youtube.com/playlist?list=PLHz\\_AreHm4dlAnJ\\_jJtV29RFxnPHDuk9o](https://www.youtube.com/playlist?list=PLHz_AreHm4dlAnJ_jJtV29RFxnPHDuk9o)