

gustavoguanabara / html-css

Public

<> Code

Issues 229

Pull requests 16

Actions

Projects 1

Security

Insights

master

html-css / aulas-pdf / 11 - Imagens dinâmicas, áudios e vídeos.pdf

guanabara-eteot

Update 11 - Imagens dinâmicas, áudios e vídeos.pdf

✓

2 contributors

3.75 MB

https://github.com/gustavoguanabara/html-css/blob/master/aulas-pdf/11 - Imagens dinâmicas%2C áudios e vídeos.pdf

1/9

CURSO DE INFORMÁTICA PARA INTERNET

MÓDULO 01 ■ CAPÍTULO 11 — □ ×

DevWeb

IMAGENS DINÂMICAS, ÁUDIO E VÍDEO




M01C11

veja em vídeo

IMAGENS DINÂMICAS, ÁUDIO E VÍDEO



Fotos, áudios e vídeos são essenciais para a construção dos sites hoje em dia. Ilustrar bem o seu conteúdo é imprescindível para deixar clara a mensagem que queremos passar. Mas é preciso tomar cuidado com formatos e tamanhos, pois muita gente abrirá os sites em seus celulares e dispositivos móveis. Vamos falar sobre esse assunto no capítulo que iniciamos agora.



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.

Seu site deve se adaptar ao tamanho da tela

Como eu disse ali na introdução do capítulo, atualmente podemos acessar sites em todo tipo de dispositivo: smartphones, tablets, notebooks, computadores desktop, aparelhos de smart TV e muito mais. Pensando nisso, você que está começando a desenvolver sites vai precisar adaptar seu conteúdo a tamanhos de tela diferentes.



A mídia que mais sofre com esse problema de tamanho de exibição são as imagens. Uma TV pode exibir imagens gigantes, mas se usarmos essa mesma foto para ser apresentada em um celular, teremos que redimensionar forçadamente a imagem com CSS.



Porém, essa prática não vai fazer com que o tamanho (em bytes) diminua também. Isso acaba aumentando o consumo de dados em dispositivos móveis e deixando seu site muito pesado, e ninguém tem paciência para acessar site lento.



CUIDADO! Sites lentos diminuem a **taxa de retenção** dos usuários, que ficam menos tempo acessando e podem prejudicar a indexação da sua página em mecanismos de busca como o **Google**. Veja no vídeo a seguir, um especialista em SEO (otimização para mecanismos de busca) falando sobre lentidão de sites, principalmente via 3G e 4G do celular.

Portal SEO: <https://youtu.be/jWnMfvSdo1E>



#DevWeb - Capítulo 11

prof. Gustavo Guanabara

Página 3 de 14

Imagens Flexíveis



Nosso primeiro passo no caminho de adaptar nosso conteúdo ao tamanho da tela vai ser aprender a gerar imagens de tamanho diferentes e a fazer o navegador carregar a imagem certa para cada situação. Para isso, devemos conhecer as tags `<picture>` e `<source>`.

Para esse exemplo, criamos as três imagens ao lado: a menor tem 300x300px, a média tem 700x700px e a maior tem 1000x1000px. Usamos o programa **GIMP**, que é um editor de imagens 100% gratuito. Essas imagens serão carregadas pelo navegador de acordo com o tamanho da janela atual. Para isso, criamos o seguinte código base:

```
<picture>
  
</picture>
```

Note que colocamos a tag `` exatamente como aprendemos no **capítulo 06** do nosso material. A novidade aqui é que inserimos essa imagem dentro da tag `<picture>`, que vai concentrar as outras fontes de imagem. Por padrão, a imagem **foto-g.png** (1000x1000px) será carregada.

O problema vai começar a surgir quando a janela do navegador chegar perto dos 1000 pixels de largura, pois a foto não vai mais caber lá. Vamos agora adicionar uma linha para resolver esse problema:

```
<picture>
  <source media="(max-width: 1050px)" srcset="foto-m.png" type="image/png">
  
</picture>
```

Note que a tag `<source>` possui três atributos:

- `type` vai indicar o *media type* da imagem que usamos (veja mais informações sobre *media types* no capítulo 10)
- `srcset` vai configurar o nome da imagem que será carregada quando o tamanho indicado for atingido
- `media` indica o tamanho máximo a ser considerado para carregar a imagem indicada no atributo `srcset`.



ATENÇÃO! Você pode até colocar o valor exato de 1000px na propriedade `max-width`, mas vai perceber que um valor ligeiramente acima vai gerar resultados mais interessantes.

Agora, recarregue seu código e mude o tamanho da janela do navegador. Você vai perceber que a imagem muda automaticamente conforme aumentamos ou diminuimos o tamanho da tela.

Vamos continuar e acrescentar mais um `<source>` à nossa imagem:

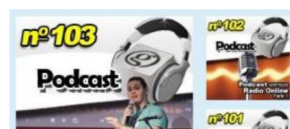
```
<picture>
  <source media="(max-width: 750px)" srcset="foto-p.png" type="image/png">
  <source media="(max-width: 1050px)" srcset="foto-m.png" type="image/png">
  
</picture>
```

É importante que existe uma ordem entre os `<source>`, e nessa nossa configuração, os itens mais acima sejam os menores tamanhos para `max-width` e que os seguintes sejam maiores, de forma crescente. O último item dentro de `<picture>` deve ser a imagem padrão.

Faça seus testes, modifique as ordens, entenda os resultados!

Vamos falar sobre áudio

Quando comecei a produzir conteúdo para Internet, os vídeos ainda não eram uma realidade palpável, pois a Internet brasileira era muito lenta e cara. Iniciativas como Volog e YouTube estavam começando a surgir, mas ainda mostravam vídeos muito curtos e com qualidade sofrível.



mostravam vídeos muito curtos e com qualidade ruim.

E foi em 2006 que comecei a produzir um podcast sobre tecnologia que chegou a ser bem conhecido e ter um número significativo de downloads por cada episódio.



Me lembro bem de que na época era bem complicado colocar conteúdo em áudio em um site. Precisava da adição de bastante JavaScript, manipulação de plugins específicos e um conhecimento bem consistente em RSS. A HTML não me ajudava em nada. Mais tarde, com a evolução dos vídeos e a melhoria da banda larga nacional, acabei optando por focar meus esforços no formato vídeo, mas tenho um carinho especial pelo áudio até hoje.



QUER ME OUVIR? Eu também participei de dois episódios de um dos maiores podcasts do Brasil: o **Nerdcast**. Dei meus pitacos sobre as profissões de Professor e de Programador. Você pode acessar o site abaixo e conferir.

https://jovemnerd.com.br/?podcast_guest=gustavo-guanabara

Com a HTML5, veio também a facilidade em compartilhar áudio nos nossos sites e sem depender necessariamente de JavaScript ou plugins extras. A partir de agora, basta uma tag `<audio>` e alguns `<source>` para fazer seu site ser capaz de tocar qualquer áudio.

