Layouts Responsivos: Trabalhando com Layouts Mobile

1. **Aula 1 - Ferramentas, Unidades e Imagens:**
   1. Unidade de medida “rem” para fontes é a mesma coisa que “em” e as outras, porém, a em é unidade de medida relativa ao font-size tag mãe, já o rem é a mesma coisa, mas ao invés da ser da tag mãe ele vai ser da que possui o maior nível hierárquico de todos, sendo ela a tag HTML, o elemento raiz do nosso html. ‘Em’ significa ephemeral, aquilo que transita, variável. O “Rem” é root ephemeral, ou seja, é variável a partir da raiz.
   2. A vantagem de usar o rem é que diferentes browsers usam diferentes tamanhos de fonte no root e, utilizando essa medida, o nosso site se adapta automaticamente.
   3. O em é melhor ser utilizado quando queremos variar apenas um bloco, que se distoa do resto.
   4. **Tipos de imagens:** 
      1. PNG: Pouco compacta, mantém a resolução da imagem, mas é grande.
      2. JPG: Bem mais compactos que PNG diminuindo a qualidade, além de não conseguir utilizar efeitos de transparência, como o fundo de PNG, por exemplo.
      3. GIF: Tamanho varia dependendo do tamanho da animação e da resolução, porém, GIF’s possuem qualidades extremamente baixas, tendo uso bem limitado.
      4. SVG: Criada através de instruções do computador, ou seja, o computador recebe a instrução e monta o desenho na tela. Ex.: “desenhe uma maça” e o computador desenha uma maça. Isso é feito por cálculos, portanto, independente do tamanho da tela, os cálculos sempre são os mesmos. A principal vantagem é o peso, sendo muito mais leve que PNG e JPG, pois, o PNG guarda quantidade e posição de pixels, já o SVG, guarda apenas a instrução dos cálculos. Além disso, se você der um zoo na imagem PNG, acabará vendo artefatos e imagem embaçada, já o SVG, como é formada através de cálculos, não perde a resolução. Qualidade bem inferior para imagens realistas e com detalhes.
      5. Em resumo: SVG é o formato mais indicado para usar em logotipos, ícones e ilustrações simples. Fora isso, o JPG é o mais indicado a menos que precise de efeitos de transparência, nesse caso usa-se PNG ou GIF’s caso queira imagens animadas.
2. **Aula 2 – Layout Mobile e Variáveis CSS:**
   1. Mobile first: Pensar primeiro nos dispositivos mais limitados durante a produção de um site, pois, mesmo que ele evolua para um site desktop, continuará totalmente funcional, diferente daqueles que foram pensados primeiro em desktop.
   2. O <meta name=”viewport” contente=”width=device-width, initial-scale=1.0”> é uma tag extremamente importante para a utilização de designs responsivos.
   3. Nós podemos pegar valores de propriedades e salvá-las em variáveis, como as hexadecimais de cores em nomes que façam sentido para a gente, dessa forma, ao invés de decorar o hexadecimal de cada cor, nós podemos salvá-los nos nomes, facilitando o uso.
      1. Para fazer isso, basta colocar “--” antes do nome para a cor, “:” depois do nome e “;” depois do hexadecimal. Tudo isso dentro da Tag mais genérica que temos: “:root{}”.
      2. Para utilizar as variáveis nós colocamos: var(--nome).
      3. A vantagem é poder alterar as cores todas de uma vez só, uma vez que você utilize a mesma variável de cor para vários itens. Ex.: colocando a variável “–branco” para todas as letras, no momento em que você alterar o código hexadecimal dessa cor para outra, todas as letras mudam de cor ao mesmo tempo, sem a necessidade de você ter que ir alterar item por item e ficar verificando se está tudo certo.
3. **Aula 3 – Particularidades de Telas:**
   1. Ter o hábito de já ir planejando classes para as tags que utilizaremos para codar a página durante o desmembramento de uma determinada área é uma boa prática, uma vez que agiliza o trabalho.
   2. Quando temos um painel e/ou figura legendada, nós temos uma tag <figure> que, dentro dela, tem uma <img> juntamente com a <figc> sendo ela a caption da figure, ou seja, a sua legenda.
   3. No width podemos colocar 100vw quando queremos que algum elemento use 100% da viewport width, ocupando a largura da tela do celular inteira.
   4. É uma boa prática sempre utilizar medidas de padding, margin dentre outros, individualmente, para não ocorrer o risco de sobrescrever as configurações por acaso.
   5. Podemos colocar uma propriedade *max-width: px;* em imagens e afins para que conforme a tela vai aumentando, ela pare de aumentar junto na quantidade de pixels determinada no CSS, sem ter a necessidade de criar um @media(max-width: 600px){.classeImagem{width: 100vw;}, para que dessa forma ele se adapte para 100% da viewport width somente quando a tela estiver menor que 600px. Diminuindo o código e facilitando a vida.
   6. Utilizar unidades relativas como: %, rem, dentre outros é extremamente bom e vantajoso, mas existem casos como o acima, onde unidades fixas sejam o ideal, uma vez que essa imagem pode acabar ficando MUITO grande e/ou perdendo qualidade de imagem se ocupar 100% da largura de tela de um computador, por exemplo, o que não ocorre se a pessoa estiver vendo em um celular.
4. **Aula 4 – Trabalhando a Semântica HTML:**
   1. Quando trabalhamos com imagens precisamos fazer a seguinte análise: “O conteúdo do site é entendível sem essa imagem?” Se sim, então essa imagem é apenas estilização e pode ser removida completamento do HTML, sendo colocada no site utilizando CSS.
   2. Para poder fazer isso, podemos criar marcadores específicos para cada imagem/ícone a ser utilizado em seus tópicos específicos, colocar um *background-image: url(caminhdoDaImagem);*, e adicionar essa classe ao seu respectivo tópico no HTML.
   3. Porém, fazendo isso teremos o problema de que a imagem colocada irá se repetir durante todo o fundo daquele elemento. Para concertar isso, podemos utilizar o *background-repeat: no-repeat;* para que mostre a imagem apenas 1 vez e ajustar sua posição através do *background-position: top 1.5rem left 2.5rem...;* e assim por diante até estar no lugar certo.
   4. Na parte de endereços de lugares, possuímos tags específicas para utilizarmos. Visualmente não tem diferença nenhuma de um <p>, mas semanticamente faz diferença e ajuda no ranqueamento do site em buscadores como google.
   5. Levando isso em consideração, todas as informações de contato das pessoas precisam estar dentro da tag <address>. Nada impede que utilizemos mais de uma, mas o ideal é que tenhamos apenas 1 tag para todas as informações de contato.
   6. Podemos fazer com que o celular abra o discador do celular da pessoa já com o número discado, basta colocar o número desejado no href do link do telefone, começando com [tel:**+55**DDDXXXXXXXXXX](tel:+55DDDXXXXXXXXXX). Fica assim: <a href=”<tel:+5511991237332>”></a>.
   7. A mesma lógica serve para e-mails, podendo abrir diretamente o e-mail da pessoa com o endereço colocado: <a href=”<mailto:contato@email.com>”></a>.
   8. Podemos alocar vídeos no nosso site como imagens utilizando a tag <vídeo src=”localVideo” controls> o parâmetro “controls” serve para adicionar controles de play, pause, dentre outros no nosso vídeo para que possa ser utilizado.
5. **Aula 5 – Evoluindo do Mobile ao Desktop:**
   1. Utilizamos os media query (*@media screen and (max/min-width: celular/tablet/pcpx){config}*) para fazer alterações específicas para determinado tipo de dispositivo, seja ele tela, como no exemplo acima onde colocamos que precisa ter *screen* e um tamanho mínimo ou máximo determinado pela pixelagem da tela do dispositivo que aquelas configurações irão se aplicar, ou colocar como *print* caso seja para imprimir um boleto e você queira criar a condição de retirar tudo o que tem na tela, menos o boleto para ser impresso.
   2. Para saber mais sobre breakpoints e valores genéricos para dispositivos (referência em inglês): <https://css-tricks.com/snippets/css/media-queries-for-standard-devices/>
   3. Podemos alterar a visibilidade de um elemento colocando *display: none;* nele, deixar assim para versões mobile e colocar de volta usando media query e colocando novamente o *display: block/inline/inline-block/flex;*.