**Módulo 1: Primeiros passos HTML+CSS**

Nesse 1° módulo aprenderemos conceitos básicos, preparação do ambiente, semântica da HTML5, textos, ligações, multimídia, estilos.

**Capítulo 1**

Aula 1: 22:48 18/01/2022

* Locais para pegar os materiais: github.com/gustavoguanabara e gustavoguanabara.github.io.

Aula 3: 23:00 19/01/2022

* Aula com indicações de livros para se aprofundar no assunto.

Aula 4: 23:00 19/01/2022

* A internet surgiu pela ARPANET para proteção de dados militares;
* O TCPIP é responsável por dividir os dados em pacotes no servidor e na máquina cliente (a que solicita esses dados), o TCPIP unifica essas informações;
* Assistir vídeo sobre envio de dados pela internet "https://youtu.be/TNQsmPf24go";
* HTML: linguagem de marcação para hipertextos;
* O WWW (World Wide Web = Rede de alcance mundial) é uma sub rede dentro da internet que englobam os servidores especializados em http, porém ele consegue se conectar com os demais servidores;

**Capítulo 2**

Aula 1: 22:52 20/01/2022

* Dígitos binários são representados por 0 e 1, que também tem o nome de bit. A cada 8 bits, temos 1 byte, que é a porção mínima para que se represente algum dado. Um exemplo de informação com o byte 01000001 é a letra A. Para ver as combinações de bytes que geram cada informação, podemos acessar a tabela Código Multibyte UTF-8;
* O computador só entende as ondas binárias (quadradas);
* A cada 1024 bytes temos 1 KB -> A cada 1024 KB temos 1 MB -> A cada 1024 MB temos 1 GB -> A cada 1024 GB temos 1 TB -> A cada 1024 TB temos 1 PB -> A cada 1024 PB temos 1 EB -> A cada 1024 EB temos 1 ZB -> A cada 1024 ZB temos 1 YB;
* MB são megabytes, que são utilizados para se referir armazenamento. Ex.: HD de 100 MB;
* mb são megabits, que são utilizados para se referir a transmissão. Ex.: a internet tem 50 mb;
* O computador entende ondas quadradas e o telefone entende ondas senoidais, e para fazer a comunicação desses aparelhos, utilizamos o modem, que é um aparelho de modulação (transforma ondas quadradas em senoidais) e desmodulação (transforma ondas senoidais em quadradas);
* Site www.iplocation.net para descobrir o IP da nossa máquina e do servidor de algum site;
* DNS (sistema de nome de domínio) são servidores que ligam os nomes aos números (IPs), semelhante a uma agenda telefônica. Assim, quando pesquisamos algum site na internet, primeiro passa pelo DNS, ele informa o IP do servidor do site para o modem e depois somos encaminhados para os servidores para ter acesso às informações;
* Rotas são os caminhos que os roteadores traçam para o envio de solicitações e informações.

Aula 2: 23:22 24/01/2022

* Domínio é um nome único que identifica seu site, é pago anualmente e para criação dele é necessário escolher uma TLD;
* Hospedagem é um espaço para armazenar seus arquivos, então deve ser escolhido pela qualidade do serviço prestado, memória, recursos, é pago mensalmente (importante escolher com cautela);
* URL: localizador de recurso único. Esse endereço serve para apontar para apenas um lugar;
* Dentro de uma URL existem partes EX.: www.github.com/gustavoguanabara. São elas: Domínio que poder ser o .github e dentro desse domínio há uma TLD que dividido em GTLD que é o .com, mas esse é só um exemplo, há vários outros como por exemplo .gov. Também há o ccTLD que identifica os países no site, como por exemplo o .br. Fora desses domínios ainda há o WWW que representa o sub-domínio. Após a barra temos o caminho;
* Https significa protocolo com segurança, o s no fim é que simboliza que o site é seguro por SSL (serviço de criptografia de dados).

**Capítulo 3**

Aula 1: 22:35 26/01/2022

* A HTML (linguagem marcação hipertextos) é focada no conteúdo do site que são textos, vídeos, imagens, tabelas, linhas, listas; As CSS (folhas estilo cascata) são focadas no estilo que são cores, sombras, tamanhos, posicionamentos e o JS (Java Script) é focado em interatividade que são menus, animações, popups, validações;
* Para poder descobrir o que é JS e CSS em um site, utilizar a extensão Web Developer.

**Exemplo de conteúdo em HTML:**

* <h1> Exemplo de título </h1>. Significado por partes: <> serve para abertura de tag, h1 é uma tag de título, o "exemplo de título" é o título e o </> serve para fechar a tag;
* Para digitar um parágrafo podemos utilizar a HTML <p> Exemplo de parágrafo </p>;
* Apesar de a grande maioria das tags terem fechamento, nem todas tem, um exemplo pode ser a <img src="foto.png" alt="exemplo de foto">. Src (origem) e Alt (texto alternativo) são parâmetros, e as frases dentro das aspas são valores;

**Exemplo de conteúdo em CSS:**

h1{

**font-family:** Arial;

**font-size:**20pt;

**color:** blue;

}

* Tudo o que estiver digitado dentro da tag é uma declaração que irá decidir o estilo da página (toda declaração acaba com ponto e vírgula (;);
* O que vem escrito antes dos dois pontos (:) é uma propriedade e o que vem depois é um valor;
* CSS tem seletores > dentro dos seletores há declarações > cada declaração é um par de propriedade e valor.

**Estrutura básica documento HTML:**

Diagrama

Descrição gerada automaticamente

* O HTML e CSS funciona da seguinte forma: dentro do servidor há um HTML e CSS, quando é digitado uma URL, é feita uma cópia desse HTML e CSS para a máquina cliente e eles são disponibilizados para o navegador. A função do navegador é analisar as linhas e disponibilizar um resultado visual para a máquina cliente.

Aula 2:21:26 27/01/2022

* Site estático é a entrega do mesmo site para todos os clientes, ou seja, os resultados e experiências serão os mesmos;
* HTML, CSS e JS são tecnologias client-side funcionam do lado do cliente. Essas tecnologias estão operando somente na máquina cliente, pois a servidora só entrega o arquivo para esses programas. Os desenvolvedores dessa área são denominados front-end e se dedicam a interação do usuário com os sites;
* PHP, JS Node, C#, Python, Rubi e Java são tecnologias server-side e funcionam do lado do servidor. Os desenvolvedores dessa área são denominados back-end e se dedicam a uma boa comunicação entre o código e o servidor;
* No caso de sites que não são estáticos, como por exemplo sites de compra online, não é possível programar só com tecnologias client-side, porque é necessário verificar diversas informações que estão armazenadas em um banco de dados. Exemplo de como funciona um site com as 2 linguagens: o cliente faz uma solicitação de camisas para máquina servidora, por exemplo. No servidor, há a linguagem PHP (pode ser qualquer outra linguagem de server-side) que vai enviar essa solicitação para o banco de dados. O banco vai devolver essa informação, a linguagem vai fazer um arquivo em HTML e devolver para a máquina cliente que através do navegador, irá trazer as informações desejadas;

Diagrama

Descrição gerada automaticamente

* O desenvolvedor que se especializa nas duas áreas (front-end e back-end) é chamado de full stack.

**Capítulo 4**

Aula 1: 19:58 31/01/2022

* Instalação e configuração das ferramentas. Nessa aula eu instalei a extensão de linguagem para o VSC ficar em PT-BRr, instalei as extensões para suporte Java Script e PHP, salvamento automático, quebra de linha e aumento da fonte;

Aula 2: 21:13 31/01/2022

* Atalho para criação da base do código html “!”;
* Quando quiser colocar uma tag, não precisa dos colchetes, podemos colocar apenas o nome da tag, por exemplo “h1” que o VSC já traz o colchete e o fechamento da tag;
* Os arquivos em html vem com a nomenclatura index.html;
* “hr” significa linha horizontal.

**Capítulo 5**

Aula 1: 20:47 02/02/2022

* Para quebra de texto, utilizar a tag <br>, que significa “break row”;

Aula 2: 20:47 02/02/2022

* Na página 4 do PDF do capitulo 5 há uma tabela com tags para trazerem símbolos para o site. Esses códigos são chamados de “HTML Entities”
* Para ver sobre os codepoints para inserção de emojis, basta acessar o site emojipedia.org;
* Para adicionar comentários no html, basta adicionar “!--";
* No momento de acrescentar os emojis, é necessário colocar “&#x” no lugar de “U+”.

**Capítulo 6**

Aula 1: 20:23 03/02/2022

* Para não ter problemas com direitos de imagens, baixe de sites que não cobre direitos autorais. Para isso há sites com imagens de domínio público: UnSplash, Pexels, FreePik, Rawpixel, Pixabay, Libreshot, Wikimedia Commons;
* Outra forma de ter acesso a imagens livre de direitos autorais, basta pesquisar o que deseja no google, clicar em google imagens, clicar em ferramentas, clicar em direitos de uso e escolher a opção “licença Creative Commons”.

Aula 2: 22:32 03/02/2022

* JPEG (Joint Photographics Experts Group) é um formato com tamanho de arquivo extremamente reduzido que é muito útil para que as imagens carreguem rapidamente nos sites, porém caso seja muito compactado ao colocar no site, a imagem fica borrada, então é necessário ficar atento;
* PNG (Portable Network Graphics) é um formato utilizado para quando precisamos adicionar transparência a imagem e é menos compactado, para o caso de imagens grandes. Ex: fotos sem fundo branco, apenas “recortadas” sem fundo;
* No google imagens, em ferramentas, podemos escolher o tamanho das imagens e cor. Isso ajuda a ver o escolher o formato do arquivo;
* Quando for escolher um formato para inserir nos sites, dá prioridade para JPEG com compactação de 30% a 50%, e usar PNG apenas quando precisar de transparência na foto.

Aula 3: 20:46 04/02/2022

* Apesar de uma imagem com maior resolução ter mais qualidade, não significa que sempre precisaremos utilizar imagens em um tamanho maior. Por isso é importante sempre usar a imagem do tamanho certo e não ficar regulando-a no CSS, pois isso pode ocasionar lentidão no site;
* O Google limita o alcance de sites lentos, então é importante criar um site bem otimizado;
* Para redimensionar imagens, podemos usar o Gimp. O máximo de resolução para uma imagem em um site é 1500 de largura. No Gimp, ao colocar o valor de largura na aba imagem, redimensionar imagem, após dá um enter, já calcula a altura proporcionalmente. Deixamos a resolução em 70% geralmente;
* Uma dica importante é salvar as fotos originais como full, pois se precisar ajustar depois novamente, tem a imagem de tamanho original.

Aula 4: 21:18 04/02/2022

* A tag <img> carrega imagens que estão na pasta do projeto atual ou em links externos. Ela tem como parâmetros básicos o src (origem) e alt (texto alternativo);
* Ao clicar dentro das aspas do parâmetro sr, clique em Ctrl+Espaço, dessa forma o editor vai abrir uma listagem com todos os arquivos disponíveis na pasta do seu projeto, assim basta escolher o desejado;
* Para adicionar imagens de links externos, ou seja, que não estão baixadas na pasta do projeto, basta escolher a imagem na internet, clicar com o botão direito sobre a imagem e clicar em “copiar link da imagem”. Após, é só colar dentro das aspas do parâmetro src;
* Os textos alternativos são importantes para mecanismos de buscas (exemplo, pelo texto inserido, sua imagem pode aparecer no resultado de busca do google fotos) e para acessibilidade, pois o navegador é capaz de descrever a imagem por esse texto para deficientes visuais, por exemplo.

Aula 5: 22:20 04/02/2022

* Favicons são ícones que ficam ao lado do título da página. O formato desses arquivos está em ICO e podem ser baixados no site IconArchive, desenhados do zero no site favicon.cc ou criado a partir de imagens já obtidas pelo site favicon.io;
* Para adicionar favicons, precisamos editar o HTML na parte <head> incluindo a tag <link> e escolhendo a opção link:favicon. Após, é só editar o nome no parâmetro “href” como na dica anterior para o caminho ser inserido;
* Pode ter outros formatos de arquivos para favicons, como por exemplo PNG e SVG. Este último é até a melhor opção por ser vetorizado, mas não é a mais indicada, pois nem todos os navegadores são compatíveis com esse formato e pode gerar inconsistências. O mais indicado é o formato ICO.

**Capítulo 7**

Aula 1: 00:04 05/02/2022

* A hierarquia dos títulos serve para organizar o conteúdo e há seis níveis que são descritos por <h1>, <h2>, <h3>, <h4>, <h5> e <h6>;
* A hierarquia funciona da seguinte forma: o <h1>é o assunto principal, o <h2> é um sub-assunto do <h1>, o <h3> é um sub-assunto do <h2>, que está acima dele. E vai seguindo dessa forma até o <h6>;
* O <h1> pode ser usado mais de uma vez nos seus sites se houverem vários assuntos principais diferentes;
* Para escrever textos genéricos no HTML, basta digitar dentro de um parágrafo a palavra “lorem” e dar um enter. Os famosos Lorem Ipsum são nada mais que textos genéricos;
* Para criar uma configuração diferente no CSS na parte de folhas de estilo, precisamos adicionar no parágrafo posterior ao <h1> uma classe de destaque, que fica da seguinte forma: <p class=”destaque”>;
* Caso o estilo de cada título não seja configurado, o título h1 terá uma letra maior, e os demais cada vez menores.

**Capítulo 8**

Aula 1: 20:24 07/02/2022

* Semântica é o significado dos vocábulos e o HTML5 é focado nessa parte;
* Para ajuste de formas é utilizado as CSS;
* Na página 4 do capítulo 8 tem links com as tags obsoletas direto do site oficial de HTML;
* Não utilizar tags obsoletas para programar sites;
* Para mudar as cores das páginas, podemos utilizar a tag <body> e o parâmetro {background-color: };

Aula 2: 19:53 08/02/2022

* A tag <Strong> significa “força” e deixa as palavras dentro dela em negrito, focando na semântica;
* A tag <em> significa “ênfase” e deixa as palavras dentro dela em itálico, focando na semântica;
* Para marcar uma frase ou palavra com sentido da caneta “marca texto”, utilizamos a tag <mark>...</mark>. Para mudar a cor da caneta, é necessário configurar com as CSS;
* Para colocar uma frase ou palavra já digitada dentro de uma tag, podemos apertar ctrl + shift + P, depois digitar abb, clicar na opção “Wrap with abbreviation”, após digitar a tag desejada sem os colchetes angulares e dar enter. Dessa forma, ele fará o que chamamos de “envelopamento com abreviatura”.

Aula 3: 20:55 09/02/2022

* Quando queremos marcar textos, utilizamos a tag <mark>. Para mudar a cor, dentro do primeiro colchete da tag, damos um espaço, escrevemos a palavra style. Depois, dentro das aspas, escrevemos “background-color” e escolhemos a cor. Se você digitar o nome da cor, te encaminha para ela também ou parando o mouse sobre a palavra, pode-se escolher a cor sem saber o nome. Ex.: <mark style="background-color: lime;">um texto marcado</mark>;
* Caso tenham várias marcações no texto e precisem ser da mesma cor, precisamos abrir a tag <style> no head do código, colocar mark, depois background-color e escolher a cor (exemplo no exercício 8);
* A tag <big> não é mais utilizada, letras grandes tem que ser formatadas nas CSS, porém a <small> ainda é super utilizada, e serve para deixar as letras menores;
* Para “textos deletados”, utilizamos a tag <del> ao invés de <strike> que se tornou obsoleta. No caso essa tag não exclui exatamente o texto, mas passa uma linha em cima dele mostrando que essa parte pode ser ignorada pelo leitor;
* No caso de “textos inseridos”, utilizamos a tag <ins> ao invés da tag <u> que se tornou obsoleta. Nessa tag, o texto será sublinhado, informando que é uma parte importante de ser lida pelo leitor;
* Para colocar textos sobrescritos ou subscritos (números ou símbolos pequenos sobre/sob a letra ou palavra), devemos usar as tags <sup> para números em cima (m²), e <sub> para números abaixo (h2o).

Aula 4: 23:11 11/02/2022

* A tag <code> serve para delimitar o código, dando valor semântico ao indicar para o navegador que a informação digitada é referente a um código de computador. Essa tag facilita bastante a leitura, pois deixam as letras mono-espaçadas, porém ele não arruma o código;
* Para manter a formatação igual a digitada, podemos usar a tag <pre> e dentro dela colocar a <code>. Aliás, essa junção de tags pode ser bastante utilizada em outros casos, como por exemplo a junção <Strong> e <em>, que deixa o texto negrito e itálico ao mesmo tempo;
* Para tirar o espaçamento dentro da tag <code>, basta selecionar o código e para recuar, clicar em shift + tab, e para adicionar espaçamento, clicar apenas em tab;
* Também é possível inserir citações. Temos algumas tags para isso: <q> (do inglês *quote*, que significa citar), ela só coloca aspas. Outra tag é a <blockquote>, neste caso o texto dentro da tag ganha um parágrafo para si com recuo. Além disso, podemos colocar a palavra cite dentro da tag para adicionar links;
* Para abreviações com significado ao passar o mouse por cima da sigla, usamos a tag <abbr> e dentro do primeiro colchete angular, damos um espaço e escrevemos title, e dentro das aspas o significado da sigla, como o exemplo abaixo:



* Para inverter textos, podemos usar a tag <dbo> que significa bi-directional override. Nessa tag, vem junto a tag “dir” que pede para indicar a direção do texto, que são rtl = da direita para a esquerda (rigth-to-left) e ltr = da esquerda para a direita (left-to-right).

**Capítulo 9**

Aula 1: 01:18 12/02/2022

* O HTML dá suporte para 3 tipos de listas: listas ordenadas, listas não ordenadas e listas de definição;
* Ordered lists são listas onde a ordem dos itens é muito importante. Para criação das listas ordernadas, usamos a tag <ol> para delimitar a lista e <li> (list item) para identificar cada item da lista;
* A tag <li> não precisa de fechamento;
* Dentro da tag <ol>, temos o parâmetro “type” para configurar o marcador da lista. Podem ser utilizados: 1 (listas numeradas), A (listas alfabéticas maiúsculas), a (listas alfabéticas minúsculas), I (romanos maiúsculos) e i (romanos minúsculas);
* Também é possível indicar o início da contagem usando o parâmetro “start”, colocando dentro dos parênteses o número/letra;
* Unordered lists são listas que a ordenação dos itens não faz diferença, como uma lista de compras por exemplo. Para listas não ordenadas, utilizamos a tag <ul> para delimitar a lista e a tag <li> para adição dos itens;
* Por padrão, o marcador é uma bolinha preta totalmente preenchida (disc), mas também é possível mudar o formato utilizando o parâmetro “type”. Podem ser utilizados: disc (padrão, bolinha totalmente preenchida), clircle (bola preta sem preenchimento) e square (quadrado preto totalmente pintado).

Aula 2: 01:34 12/02/2022

* Para mudar uma letra de várias tags, por exemplo alterar a letra “o” da tag <ol> por “u”, podemos segurar alt e clicar no meio dessa tag, depois apagar vários “os” e inserir o “u”. Assim, fará a troca de vários ao mesmo tempo;
* Também é possível misturar as listas, conforme o código abaixo:

Texto

Descrição gerada automaticamente

* Lista de definições é uma lista sem demarcadores, como se fosse um dicionário;
* Toda lista de definições está dentro da tag <dl> (definition list);
* Cada termo é um <dt> (definition term) e possue fechamento opcional no HTML5;
* Cada descrição é um <dd> (definition description) e possuem fechamento opcional no HTML5.

**Capítulo 10**

Aula 1: 20:20 14/02/2022

* Os hiperlinks permitem a ligação de um ponto a outro. Quando baixamos um arquivo, clicamos em um link que nos encaminha para outro site ou página, estamos interagindo com hiperlinks;
* Os mecanismos de busca do google utilizam os links para fazer as pesquisas;
* Para criar um hiperlink, devemos criar âncoras através da tag <a>. No atributo href, criamos uma referência hipertexto;
* Outro atributo utilizado é o “hreflang”, que indica qual o idioma principal do site para o qual o link está desviando o fluxo de navegação;
* Para controlar o site de destino, podemos usar o atributo “target”. Ele suporta os valores \_blank (vai abrir uma nova página), \_self (vai abrir o link na janela ou frame atual, como funciona o padrão), \_top (vai desfazer todos os frames e abrir o destino no navegadorcompleto), \_parent (similar ao uso do \_top em uma referência a janela mãe e nome-do-frame (caso esteja usando frames, indicar o nome da janela a abrir). Porém os mais utilizados dessa lista são \_self e \_blank;
* Para indicar a natureza do link, podemos utilizar o atributo “rel”, e ele tem vários valores: next (próxima parte do doc atual), prev (parte anterior do doc atual), author (indica para o site do autor do artigo atual), external (outro site que não é o atual) e nofollow (indica para um site que não deve ser seguido/aval pelo google, como um link pago por exemplo);
* É extremamente indicado utilizar o atributo target=”\_blank” rel=”external” para links externos, que encaminham para outro site. Dessa forma, o seu não será fechado.

Aula 2: 21:48 15/02/2022

* Links internos ou locais, são links que encaminham o visitante para uma página dentro do nosso próprio site, não sendo necessário colocar a URL completa no código;
* Para poder encaminhar para segunda página do site, ela precisa estar dentro do mesmo arquivo onde está a primeira página;
* É importante usar os atributos de próxima página e anterior para que o google não decida por si só a próxima página, e sim leia o código e respeite a ordem desejada;
* Para voltar uma pasta, colocamos ../ e./ é para apontar para a própria pasta (esse é um comando Linux);
* Links externos nos levam para outros sites, e nesse caso é necessário indicar a URL completa, incluindo http:// ou https:// e o caminho que leve a uma página específica, se for necessário.

Aula 3: 21:20 16/02/2022

* Quando não tivermos um link para adicionar no “href”, podemos utilizar “#”;
* Para fazer links de downloads, deveremos fazer o link diretamente para o arquivo que se deseja baixar e adicionar o atributo “download”, com parâmetro “type” para indicar o tipo do arquivo que será baixado. O código ficará: <a href="Livro/meulivro.zip" download="meulivro.zip" type="application/zip">Livro compactado em ZIP</a>;
* No caso de baixar PDF, o link encaminha para outra página, mas não abre o arquivo mesmo com esses atributos. Porém, caso queira baixar, podemos clicar com o botão direito no link e apertar em “abrir link como” e o arquivo abrirá o local onde será baixado;
* Os “media types” mais utilizados no dia a dia, são: application/zip, text/html, text/css, text/Javascript, vídeo/mp4, vídeo/H264, vídeo/JPEG, áudio/aac, audio/mpeg, font/ttf, image/jpeg e image/png. Para mais, acessar o link <https://www.iana.org/assignments/media-types/media-types.xhtml>;

Aula 4:

* Somente instruções sobre as próximas aulas e desafios.

**Capítulo 11**

Aula 1: 23:40 18/02/2022

* Para carregar imagens em celulares, deveremos utilizar o tamanho 300x300, resolução a 80. Para tablets, 700x700 com resolução a 80 e para computadores e TVs 1000x1000 a 80;

Aula 2: 20:03 21/02/2022

* Para adaptar as imagens ao tamanho da tela, utilizamos as tags <picture> e <source>;
* A tag <picture> vai encontrar outras fontes de imagem;
* Usamos a tag <source> para poder controlar os tamanhos das imagens;
* A tag <source> tem 3 atributos: type (indica a media type da imagem que utilizamos), srcset (vai configurar o nome da imagem que será carregada quando o tamanho indicado for atingido e nela não funciona o atalho crtl + espaço) e media (indica o tamanho máximo para carregar a imagem indicada no atributo srcset, e o ideal é que seja um pouco a mais do tamanho máximo, tipo 50px);
* Para inserir a <source> deve-se respeitar uma ordem crescente;
* O último item dentro de <picture> deverá ser a imagem padrão.

Aula 3: 20:29 21/02/2022

* Para conseguir áudios sem direitos autorais para colocar nos sites, podemos ir no youtube, clicar no nosso canal, YouTube Studio, biblioteca de áudio e em músicas gratuitas podemos escolher entre necessária atribuição (necessário colocar a frase no site) e sem atribuição. Após, é só baixar;
* Os formatos aceitos pelos navegadores para áudios são MP3, WAV e OGG, sendo que o WAV não é recomendado por ser muito pesado;
* Podemos colocar áudios nos sites das seguintes formas: somente com a tag <audio src="" controls autoplay></audio>, ou então utilizar o atributo “source”, com ele podemos fazer uma espécie de conversor colocando diversos formatos de áudios para o site ir tentando carregar, sendo que é necessário colocar em ordem crescente, sendo o primeiro o formato que mais deseja carregar. Fica da seguinte forma:

<audio preload="metadata" autoplay controls loop>

<source src="midia/guanacast-33.mp3" type="audio/mpeg">

<source src="midia/guanacast-33.ogg" type="audio/ogg">

<source src="midia/guanacast-33.wav" type="audio/wav">

<p>Infelizmente seu navegador não consegue reproduzir áudio. <a href="midia/guanacast-33.mp3"></a>Clique aqui para baixar o arquivo MP3.</p>

</audio>

* O atributo “preload” tem algumas opções: auto (vai carregar todo o áudio mesmo que o usuário nunca aperte o play, não é recomendado); metadata (carrega somente informações sobre o arquivo, o áudio em si somente no momento do play) e none (não carrega absolutamente nada);
* O atributo “controls” mostra o controle do play no site;
* O atributo “loop” reproduz o áudio novamente após acabar;
* O atributo “autoplay” toca o áudio automaticamente, mas ele não funciona em qualquer navegador.

Aula 4: 21:47 22/02/2022

* Para baixar conversor de vídeos: <https://handbrake.fr/downloads.php>. Ao acessar ele, escolhemos o arquivo, o tipo e tamanho de arquivo que iremos converter e depois iniciaremos a conversão;
* Os formatos mais utilizados são MP4 e MV4;
* Podemos baixar vídeos sem direitos autorais no site Pexels;
* No hand só é possível converter para os formatos MP4, MV4 e WEBM. Para converter em OGV, podemos apenas colocar no google e usar um site gratuito. Ex: [Convertendo seus arquivos...55.84% (media.io)](https://www.media.io/br/convert/mp4-to-ogv.html);

Aula 5: 21:50 22/02/2022

* Para inserir vídeos hospedados localmente (selfhost), podemos utilizar a tag <video>. Ela pode ser usada sem o parâmetro “source”, porém ele abrirá apenas um formato de arquivo e isso não é bom, porque os navegadores abrem formatos diferentes de arquivos;
* Dentro da tag <vídeo>, podemos usar os atributos “width” para ajustar a dimensão do vídeo em tela (isso não modifica o tamanho do arquivo, somente as dimensões), o atributo “poster” para colocar capa do vídeo enquanto ele não é reproduzido, o atributo “controls” para ter o controle de play no vídeo e o atributo “loop” para tocar o vídeo sempre após o play;
* No atributo “source:src” inserimos apenas o caminho no “src” e o formato do arquivo no “type”, que são: vídeo m4v o type é vídeo/mp4, mp4 o type é vídeo/mp4, ogv o type é vídeo/ogg e webm o type é vídeo/webm;
* Sempre colocar a tag de <p> com informativo para aparecer uma mensagem quando não rodar o vídeo.

Aula 6: 21:14 23/02/2022

* Como hospedar vídeos no próprio servidor fica extremamente caro, outra forma para fazer isso é colocando vídeos de links externos, como por exemplo do YouTube e Vimeo. Para isso, o cliente pode criar uma conta nessas duas plataformas;
* No YouTube, podemos acessar o vídeo, clicar em compartilhar e depois copiar a tag de HTML que irá ser disponibilizado. Esse link vem dentro da tag <iframe> e podemos colar essa tag e colar no código para que esse vídeo apareça no site, porém nesse formato o vídeo fica público e tem um excelente algoritmo de busca (codec), fazendo com que o vídeo rode melhor;
* Caso você queira hospedar os vídeos de forma exclusiva, podemos usar o Vimeo. A qualidade de imagem é muito boa, porém não tem um algoritmo de busca (codec) tão bom quanto o Youtube, fazendo com que o vídeo trave um pouco dependendo da internet e tem limite de vídeos para subir.

Aula 7:

* Explicação do desafio 9.

**Capítulo 12**

Aula 1: 20:38 02/03/2022

* Inline style: entre as tags na área <body>, para edições pontuais e deve ser evitada;
* Para fazer alterações visuais pontuais, colocamos as tags referentes as CSS dentro dos locais onde desejamos mudar. Exemplo: <h1 style="color: red;">Capítulo 1</h1>;
* Para alterar estilos, podemos usar a tag <style> e dentro dela atributos, como por exemplo color, font-family, text-align para alinhar texto e muitas outras.

Aula 2: 21:50 02/03/2022

* Internal style: tags de style na área <head> e deve ser usada apenas para configurações pequenas;
* A tag <style> deve estar sempre dentro na área <head> para estar em conformidade com a W3C;

Aula 3: 21:50 02/03/2022

* External style: link para arquivo externo de configuração somente das CSS na área <head> e deve ser usada sempre que puder;
* Também é possível manter as CSS fora do HTML, utilizando a tag <link rel="stylesheet" href="style.css">. Essa tag fica embaixo da tah <title>, mas não é obrigatória essa ordem, só tem que estar na área “head”;
* Clicando sobre o link com ctrl apertado, ele vai te encaminhar para o arquivo, que caso não tenha sido criado, ele perguntará se deseja criar;
* Os comentários nas CSS são colocados entre /\* e \*/;
* É possível misturar as 3 técnicas em um documento, criando um CSS externo para as configurações globais, CSS interno para as configurações locais de um documento e CSS inline para pequenas configurações pontuais;
* Também podemos utilizar mais de um link para CSS, no caso de estilos extras para algumas páginas. Ele irá somar os dois estilos dos dois links.