HTML e CSS e JS

Curso em vídeo

**História da internet:**

Darpa criou a arpanet (rede) para interligar 4 PCs diferentes em estados diferentes dos EUA, por serem diferentes não falavam mesma língua, por isso criou-se protocolo NCP (possui o FTP para transmitir arquivos e o DNS para identificar maquinas na rede)

Criou-se sistema para enviar mensagens eletrônicas, o E-mail = usuário@servidor

Criou-se o protocolo TCP (substitui o NCP que não aguentava muitas maquinas interligadas), ele dividia os dados em pacotes, criou-se o termo internetting (que mais tarde substitui o arpanet por internet) representa redes que se comunicam entre si

Criou-se o protocolo IP, porque o TCP não era bom em identificar as maquinas na rede, juntou os 2 (chamando TCP/IP), que permitiu a ligação entre várias maquinas

Hipertexto = texto, imagem, som, conteúdos multimídias

Criou-se o protocolo HTTP, para transmitir conteúdos multimídia pela rede, criou uma linguagem de hipertexto (HTML) e criou o conceito de WWW (engloba todos servidores de multimídia que utilizam o HTTP)

**História do HTML:**

Estudantes da faculdade NCSA criaram o navegador Mosaic (mostravam todos conteúdos multimídia), depois fundaram sua própria empresa (Netscape)

Esses mesmos estudantes, depois da Netscape quebrar, criaram a fundação Mozilla que criou o Mozilla Firefox

Essa Netscape era rival da Windows (que tinha a internet explorer)

Google cria o navegador google chrome

HTML5 (2013): separa semântica, estilo (formatação) e interatividade (HTML, CSS, JS, respectivamente)

**Infraestrutura da internet:**

Tudo representado num computador é em binário

Se os PCs tiverem próximos dá para conectar por servidor local, se tiverem distantes precisa de um meio de comunicação para longas distancias. Linha discada ou serviço de banda larga (esse que usa antenas/telefones/TV a cabo etc como base). Esses utilizam ondas senoidais ou sinal analógico

Sinal digital = são aquelas ondas dos números binários (onda quadrada), o clock e o dado. Existe um aparelho que transforma o sinal digital e sinal analógico (modulação), o contrário é demodulação. O aparelho é o modem

Provedor de acesso = empresa que te dá acesso à internet, o modem da sua casa está diretamente ligado com o provedor, quando um aparelho se conecta com a internet ele recebe um IP (servidores também tem IP)

Um servidor armazena/hospeda um site, o servidor DNS ajuda localizar/identificar os sites nos servidores, sites terminados em “.br” vão pra um DNS que cuida só disso

Arquivos são transmitidos de maquinas para maquinas ou servidores para maquinas por pacotes, divide em partes e envia separadamente, aí no destinatário essas partes se juntam

Obs: MP3 = música e MP4 = vídeo

Domínio = nome que você dá a um site, você paga anualmente por ele

**HTML:**

Linguagem de Marcação (não é uma linguagem de programação), ou seja, marcar onde começa e termina um determinado conteúdo

Deve ser usado para descrever como será a estrutura de uma página web, quais conteúdos terá

index.html, style.css

O conteúdo (HTML), a apresentação (formatação fantástica, CSS) e o comportamento (interatividade, JavaScript)

Tudo que aparece entre <head> e </head> diz sobre como o programa vai se comportar

Tudo que aparece entre <body> e </body> é o conteúdo que aparece no site

h1, h2, h3 é hierarquia de título

Toda vez que um h1, h2 e/ou h3, tenha relação, seja sobre o mesmo assunto, colocar um <hgroup> pra manter a ordem no programa

Interface gráfica do usuário

Descrição gerada automaticamente esse ícone de tela no inspecionar elemento serve pra ver como vai ficar seu código em outras telas

<title>: é um texto que será exibido assim que você posicionar o ponteiro do mouse sobre alguma imagem, muito comum possuir o mesmo conteúdo do atributo alt.

<ol> faz lista ordenada, <ul> faz lista não ordenada, <li> faz itens da lista. Listas não ordenadas serão de grande uso na criação dos menus de navegação com <a>

Google fonts:

Texto

Descrição gerada automaticamente

Text-transform:

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

## Text-indent

Esta propriedade é usada para definir o recuo da primeira linha de um texto.

## Letter-spacing

Define o espaçamento entre as letras.

## \*\*Word\*\*-spacing

Define o espaçamento entre as palavras.

## Line-height

Define a altura da linha, permitindo assim alterar a distância entre as linhas do texto.

Links para ligações:

Uma imagem contendo Texto

Descrição gerada automaticamente

Links para baixar arquivos:

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Tags block: ocupam todo o espaço da linha onde estão posicionadas, dessa forma não permitem outras tags sejam do lado (<p>, <h1>, <li>), pode definir width e height

Tags inline: ocupam exatamente o tamanho do seu conteúdo (então não pode definir um width e height pra div), permitindo outras tags do lado caso exista espaço (<a>, <img>, <span>)

Tag display: alternando o seu valor entre inline e block

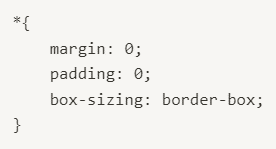
Tag float (float: right/left): serve para jogar o conteúdo para um canto da div, mas tome cuidado conteúdos que vierem logo após o elemento que usa essa propriedade, também flutuarão caso exista espaço para isso. Para isso não acontecer usa-se clear: both no elemento seguinte ao do float

ATENÇÃO: usou padding, usa box-sizing: border-box, pois alterará o comportamento dos elementos com padding, permitindo que o espaçamento interno seja aplicado, sem a altura e largura do container serem alteradas

Overflow: define o que irá acontecer quando o conteúdo inserido em uma div for maior que elaUma imagem contendo Interface gráfica do usuário

Descrição gerada automaticamente

uso do sinal de \* (asterisco) podemos aplicar formatações a todos os elementos de um página. Bem utilizado quando queremos resetar alguns padrões que os navegadores sempre aplicam assim que renderizam alguma página.



Pseudo classe sempre aparece com “:” antes

## Pseudo classe hover

Aplica uma formatação ao elemento quando o mouse passar sobre ele

## Pseudo classe nth-child(nº)

Aplica uma formatação a um elemento conforme a sua posição dentro de um grupo. Ou “odd” (ímpar) e “even” (par)

Flexbox/flex: conjunto de propriedades que possibilitarão o alinhamento dos containers, layouts flexíveis de uma maneira mais amigável e bem tranquila. Aplicarmos ao container principal/pai (flex container) com display: flex. Por padrão é row (linha), mas pode ser coluna com flex-direction: column

## Flex-wrap

Por padrão o Flexbox deixará todos os flex itens de um container em uma mesma linha, mesmo se você tiver um número absurdo de elementos, ele irá ajustar cada um deles gerando uma barra de rolagem horizontal.

Flex-wrap: wrap = permite a quebra de linhas quando os flex itens não couberem na mesma linha do container.

Justify-content e align-items só fazem sentido usar quando tem o display: flex

## Order

Também aplicado apenas aos flex itens, ela define a ordem como os elementos devem aparecer na tela.

Devemos usar tabelas apenas e tão somente, para a exibição de dados.

<table>

<caption> = título na tabela

<tr> = “table row” = linhas, dentro de cada uma terão suas colunas com seus conteúdos

<td> = informações devem ser inseridas, “table data”, dentro da tag <tr>, podem ter vários <td> dentro do mesmo <tr>

primeira linha de dados da tabela é considerada uma das mais importantes porque é ali que definiremos os nomes das colunas, podemos substituir a tag <td> pela tag <th> em todos <td> da 1º <tr>

dividir o conteúdo da sua tabela em cabeçalho, corpo e rodapé, uma forma mais organizada, <thead>, <tbody> e <tfoot>

Texto

Descrição gerada automaticamente

Uma imagem contendo Texto

Descrição gerada automaticamente

Na maioria das vezes usamos o <tfoot> como uma linha para alguma observação sobre os dados: um comentário. A tag <tfoot> deverá ser inserida em nosso código logo após a tag <thead>, abaixo dela é que deverá vir a tag <tbody>

Texto

Descrição gerada automaticamente cria borda para tabela, aparece antes de qualquer <tr>

Estilizando a tabela:

Texto

Descrição gerada automaticamente Texto

Descrição gerada automaticamente

Tabela

Descrição gerada automaticamente

Semântica:

Uma imagem contendo Diagrama

Descrição gerada automaticamente Na mesma section pode ter diversos articles

Header: Define o cabeçalho para a página ou para outros containers, muito usado para a inserção de logotipos, áreas de navegação e campos para busca.

Main: Define o container que receberá o conteúdo principal da sua página. não deveremos ter mais de um elemento <main> no mesmo documento HTML.

Section: Define uma ou mais seções em nossa página. deve abordar algum determinado conteúdo ou assunto, e sempre possuir um título. não usar esse elemento como um container genérico

Article: Define um container de conteúdo mais específico, podemos ter vários

Aside: Define um container de conteúdo separado, elemento é posicionado ao lado do conteúdo que ele foi associado, como uma barra lateral.

Nav: Define um container que deve agrupar os links criados pela <a> do sistema de navegação, sempre acompanhada de <ul> e <li>

<figure><figcaption>

<footer> = rodapé

<address> é utilizada para definir as informações de contato. Pode ser usada para: endereço, telefone, email, url

Position: define a posição que um elemento ficará em relação ao body ou ao seu container pai, top, bottom, left, right e z-index.

Position: absolute = qualquer lugar da div pai (se o pai tiver Position: relative) ou do body se não existir div pai

Position: relative =

Position: fixed

z-index: sobreposição, maior valor ficará acima dos demais.

<form> = formulário

Atributo action: Define o que irá acontecer com os dados digitados quando o usuário pressionar o botão de envio

Atributo method: define qual o tipo de método que será usado para enviar os dados digitados, possuímos o método GET (utiliza a URL, para enviar as informações digitadas ao servidor) e o método POST (envia as informações incluídas (encapsuladas) no corpo da mensagem, evitando assim a sua visualização, usado para formulários de contato, de login, de arquivos)

<label> = identificador para um respectivo campo do formulário, etiqueta, “usuario/login e senha”, possui atributo “for” que permitirá a criação de um vínculo entre o identificador e o campo desejado. For = id

<input> = respostas/inserção da maioria dos componentes do formulário. Atributo “type”

placeholder = consegue colocar dica de preenchimento para o usuário

required = obrigatorio campo de digitação, is simply taken by the browser to mean True if present.

radio, é aquele botão de selecionar coisas (filtros) dá várias opções e o usuário só pode escolher um, para isso “name” iguais

checkbox botão quadradinho, You can default a Checkbox to be ticked simply by adding the checked attribute. Now the box is ticked when the Form is loaded

disable faz com que não possa clicar

Type: range = insere uma barra deslizante para receber um valor de um intervalo previamente determinado. O intervalo padrão é de 0 a 100

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Transform: permite que você manipule visualmente um elemento possibilitando que fique inclinado, rotacionado, seja redimensionado. Scale(nº)=redimensionar (valor padrão é 1). ScaleX=redimensionar algum elemento horizontalmente. ScaleY=redimensionar algum elemento verticalmente. Skew(nº deg)=inclina um elemento para a esquerda ou direita. Rotate(nº deg)=rotaciona um elemento para a esquerda ou direita.

Design responsivo:

Uma imagem contendo Diagrama

Descrição gerada automaticamente

Bootstrap:

p-1a5 = padding (pt=padding-top, pb=padding-bottom, ps=padding-start(left), pe-padding=end(right); py=plano cartesiano(2D). Trocado p por m = margin

tudo que tiver “-outline” = hover, tipo btn-outline

btn-lg(large), btn-sm(small)

img-thumbnail faz uma moldura na imagem; img-fluid deixa responsivo

cria padroes de cores:

:root {

--blue-30: hsl(240, 100%, 30%);

--blue-50: hsl(240, 100%, 50%);

--blue-70: hsl(240, 100%, 70%);

--blue-90: hsl(240, 100%, 90%);

}

ATENÇÃO: para usar grid precisa de uma linha (row)

class: lead = deixa a fonte maior e mais fina

class: d-flex = display: flex

class: col-12 col-sm-12 col-md-6 = disposição dos conteúdos no row, sm md lg são os tamanhos de telas, e os nº são as quantidades de conteúdos 12 é o máximo

<https://app.netlify.com/drop> = link de criar site

qr code generator = site de criar qr pro nosso site

textarea = área para escrever algo no formulário

<select> = é aquele negocio de abrir opções, clica e abre várias opções, sempre acompanhado de <option value=""></option>

Obs: se colocar tag <optgroup label=”categorias etc”> vc separa as opções que o usuário pode clicar por categorias, fica mais fácil de navegar

selected = deixa opção selecionada, antes do usuário clicar em algo

datalist e option = usa pra colocar um campo de digitação e o próprio negócio complementa, por exemplo: Cidade: \_\_\_\_\_ e quando digita São aparece várias cidade começadas com São

<br> = line break

Texto, Carta

Descrição gerada automaticamente

Diagrama

Descrição gerada automaticamente com confiança média

Texto, Carta

Descrição gerada automaticamente

Tabela

Descrição gerada automaticamente com confiança média

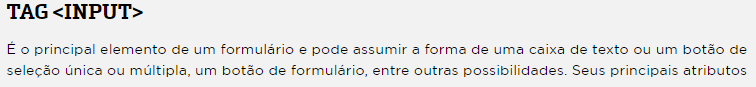
Colocando div consegue espaçar, mudar de lugar etc

Nano Courses

Forms:

Texto

Descrição gerada automaticamente



Texto

Descrição gerada automaticamente com confiança média

Texto

Descrição gerada automaticamente com confiança baixa

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Texto, Tabela

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Select faz aquele de clicar e aparecer opções de respostas no formulário

Texto

Descrição gerada automaticamente

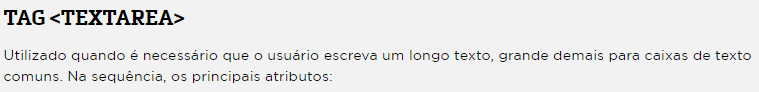
Atribuições na tag <select>





Tag <option> são as opções do select





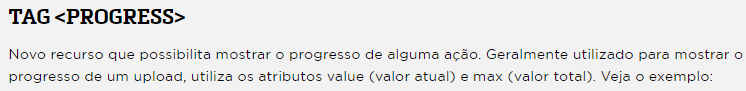
Obs: usar heigh e width no css para definir tamanho da textarea

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente



CSS:

Interface gráfica do usuário, Texto, Email

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Tela preta com letras brancas

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

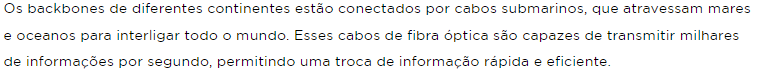
Internet:

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente



Texto

Descrição gerada automaticamente

Camadas do protocolo de rede:

Aplicação (enviar e receber infos, protocolos como SMTP=email, FTP=arquivos, HTTP=internet) > Transporte (recebe dados, verifica e divide eles) > Rede (anexados em IP do remetente e destinatário) > Interface (enviar e receber pacotes pela rede)

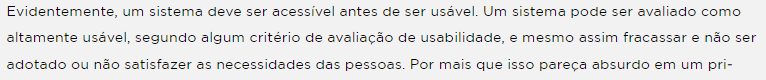
Dominio: tradução do IP para um nome mais fácil de lembrar (URL)

Associação entre o domínio e IP é feito pelo DNS (Sistema de domínio de nomes)

Diagrama

Descrição gerada automaticamente



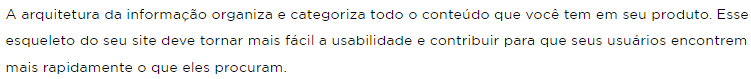


Texto

Descrição gerada automaticamente







Wireframe = layout do site/app, como os elementos (HTML) vão aparecer, forma, tamanho, letra, imagem, fonte

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente



Texto

Descrição gerada automaticamente com confiança média

Uma imagem contendo Linha do tempo

Descrição gerada automaticamenteTela de celular

Descrição gerada automaticamenteCelular com tela ligada

Descrição gerada automaticamente

Canvas:





Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente com confiança média







Texto

Descrição gerada automaticamente

Funções:

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente



Círculos:

Texto

Descrição gerada automaticamente com confiança média

Linhas:

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Fonte:

Texto

Descrição gerada automaticamente

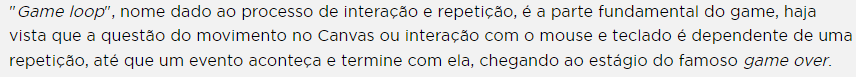
Texto, Carta

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

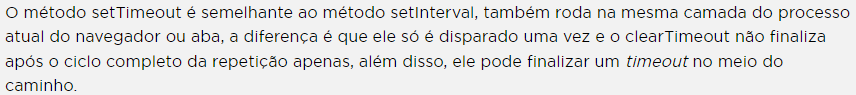
GameLoop:

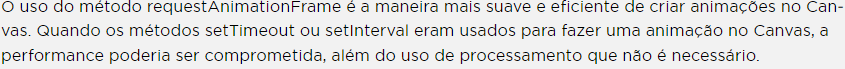


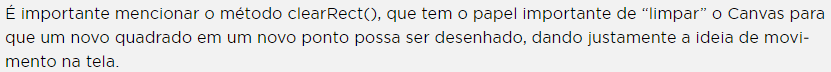


Texto

Descrição gerada automaticamente







Texto

Descrição gerada automaticamente

Set interval onde a função gameloop acontece em 50 milisegundos

Função gameloop onde se a coordenada eixo x for menor que 700, executa função desenharQuadrado, se não parar set interval

Função desenharQuadrado onde existe um clearRect para deixar o canvas limpo para ser desenhado, vem antes do desenho, depois fillStyle e depois fillRect

Texto

Descrição gerada automaticamente

Com o requestAnimationFrame é diferente, chama ele com a função gameloop

Função gameloop igual a outra mas o se não tem um clearTimeout e o requestAnimation denovo

Interação canvas e mouse/teclado:



Teclado de computador

Descrição gerada automaticamente

Números que identificam as teclas

Texto

Descrição gerada automaticamente

Texto, Carta

Descrição gerada automaticamente

80 = letra P

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Texto, Carta

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Colisão: precisa pensar na sobreposição de 2 objetos/players etc

Texto

Descrição gerada automaticamente

JavaScript: programação orientada a eventos, eventos disparados pelo usuário, interação

Não tem relação com java, só o nome parecido (marketing), mas código parecido pois ambos descendem da linguagem C

Texto

Descrição gerada automaticamente

Pode aparecer no HTML como JS obstrusivo, colocando um evento (onClick por exemplo), em um btn



Mas o JS junto do HTML é ruim

Tag <script> no HTML também é ruim, é obsoleto

JS não obstrusivo = JS separado do HTML, por exemplo:



Texto

Descrição gerada automaticamente

Para melhorar só se separar os arquivos JS e HTML, para os motores de busca se otimizarem



Tipos de variáveis:

Object – objetos instacionados a partir de uma classe ou retornados por função

**(Um tipo de object muito utilizado é o document)**

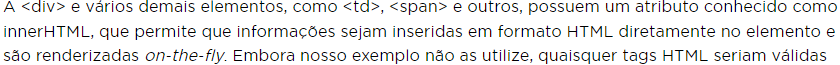
Undefined – variável declarada, mas sem valor, sem tipo

Null – sem valor, mas com tipo









Funções matemáticas em JS:

Uma imagem contendo Interface gráfica do usuário

Descrição gerada automaticamente

== é diferente de ===, == só compara os valores, === compara os valores e o tipo de variável

Eventos do JS:

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Expressões regulares:

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente



Texto

Descrição gerada automaticamente

Orientação a objeto no JS:



Texto

Descrição gerada automaticamente

Texto, Carta

Descrição gerada automaticamente

 = 

Jogos:

Texto

Descrição gerada automaticamente

Document.write(“”) = comando para escrever na tela, aparecer infos na tela

Date() =

() = parâmetros & {} = códigos

Existem diversas chamadas no JS começadas em “on”, olhar tema “eventos no JS” acima

Tela de computador com texto preto sobre fundo branco

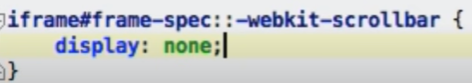
Descrição gerada automaticamente

Mudando imagens quando passa o mouse, quando clica, quando coloca e tira o mouse etc

Se você quiser carregar um conteúdo dentro de outro conteúdo, por exemplo, uma seção da página do demas dentro da página de web (no sprint 3/4 do robô), precisa usar a tag <iframe>, sempre acompanhado de name e id.

Tela de celular com publicação numa rede social

Descrição gerada automaticamente com confiança baixa Fica assim, uma abinha

 Esconde o scrollbar

Mapas de Imagem:

Tag <map>, ai precisa dar um name pra ele e linkar ele com a imagem que você quer. Para linkar coloca na tag <img> usemap=”#name” (name do map)

Map vem em seguida do img no código

Tag <area> dentro dela shapes=””, pode usar rect=retângulo, circle=círculo, poly=poligonal. Dentro da tag aparece coords=”x,y,x,y” (coordenadas, para saber elas, precisa de um editor de imagens, no editor clica F8 ou janelas + informações, nessa aba onde você passar/selecionar o cursor vai mostrar as coordenadas). E por fim na tag coloca href=”” pra saber o que abrir quando clicar (se quiser abrir so uma coisa especifica, colocar #id junto do href). Target=”name do iframe” é onde vai ser aberto, nesse contexto queremos que seja na tag <iframe>

Uma imagem contendo Forma

Descrição gerada automaticamente   

Gráfico

Descrição gerada automaticamente  

Para as coordenadas dos formatos circulares é um pouco diferente, diz qual a coordenada do centro e o raio. No editor de imagens usa ferramenta de círculos, circula a área e descobre a largura (diâmetro)



Diâmetro=23, aproxima para par (24), raio é metade, 12

Para as coordenadas dos formatos poligonais é um pouco diferente, pega (x,y) dos 4 cantos no sentido horário, começando com o esquerdo superior

Pode colocar imagem pelo css, não precisa ser só pela tag <img>

Texto

Descrição gerada automaticamente com confiança média

Background-position: 50% 50% significa que a imagem vai estar posicionada no centro dela

Tag <audio>, coloca id e controls=”controls” dentro, coloca tag <source> entre, dentro do source entra src=”.mp3 ou .ogg ou .m4a” e type=”áudio/mpeg ou áudio/ogg ou áudio/x-aac”

Tag <video>, coloca id dentro, coloca tag <source> entra src=”.mp4 ou .ogv ou .webm” e type=”vídeo/mp4 ou vídeo/ogg ou vídeo/webm”

Poster para o vídeo: poster=”imagem.jpg/png”

Colocar vídeo do YT: pega o link embedado, o iframe, só colar no seu HTML

**Só anotei por anotar, não tem porque usar isso quando tem bootstrap**

Formulário: evitar colocar mesmo nome no name e id nos inputs

Get é mais veloz, coloca os dados do form na url, mas não é tão seguro

Post é mais lento mas mais seguro, enviado por pacotes dentro da solicitação que fez no site

Obs: ai coloca dentro da tag <form> method=”post” e action=”mailto:contato@gmail.com” (sendo esse aiction pra onde as infos vao ser enviadas, nesse caso um email (mail to)). Mas é inútil pois:

Obs: linguagem de servidor, como PHP serve pra mexer com os dados que foram enviados pelo form e o tratamento dos dados

Responsividade

Podemos definir a internet como um conjunto de redes de computadores interligadas pelo mundo, que utilizam um conjunto de protocolos e serviços em comum para permitir uma comunicação.

1960/70/80 – EUA cria ARPANET usado por militares e faculdades

Diagrama

Descrição gerada automaticamente

1990 - Internet foi disponibilizada para a população Internet foi disponibilizada para a população em geral. Timothy John Berners-Lee desenvolveu a World Wide Web (www) que possibilita a utilização de interface gráfica e a criação de sites com a linguagem HTML e a sua transferência com o protocolo HTTP, permitindo navegar de um site a outro.

Servidores são computadores de alta capacidade de processamento e armazenamento que rodam softwares específicos e estão conectados à internet.

Backbone é um conjunto de poderosos computadores conectados por linhas de grande largura de banda, como canais de fibras ópticas, elos de satélites e elos de transmissão por rádio. Para constituir a internet, vários backbones estão interligados e encontram-se hierarquicamente divididos: os de ligações intercontinentais, que derivam de backbones internacionais, que por sua vez derivam de backbones nacionais.

Mapa

Descrição gerada automaticamente

TCP é o Protocolo de Controle de Transmissão e o IP é o Protocolo de Internet.

Diagrama

Descrição gerada automaticamente

HTML:

Texto

Descrição gerada automaticamente





Texto

Descrição gerada automaticamente Interface gráfica do usuário, Aplicativo, Teams

Descrição gerada automaticamente

CSS: Efeito cascata = ao aplicar mais de um estilo em um elemento, poderá ocorrer uma sobreposição de estilos. Do estilo mais geral ao mais específico: navegador -> externo -> <style> no head -> inline

JS: só se popularizou depois que o framework para back-end NodeJS foi criado. Dev full-stack usa para programar todo sistema. Existem outros frameworks também: Electrom (apps desktop), Johnny-Five ( IoT e robótica), MongoDB (banco de dados)

Função “alert” é igual “console.log” mas ela apresenta uma caixa de confirmação para o usuário com a mensagem digitada (o console.log só aparece no console)

DOM = Document Object Model = possui todas infos do HTML

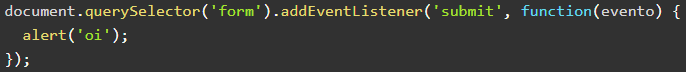
Função “querySelector” serve para pesquisar alguma info no document/DOM. Pesquisar tag, class, id

Texto

Descrição gerada automaticamente

Função “addEventListener” = escutador de eventos, usado para quando usuário faz algo

Função anônimo = não tem identificador/nome, exemplo:



Função chamada “forEach“= percorre cada elemento da lista e executa uma determinada ação.

Acessado o valor digitado no input pela propriedade “value”, esse valor deve ser atribuído à propriedade textContent da “td” criada.

Texto

Descrição gerada automaticamente (dentro do function)

“td” deve ser inserido dentro da “tr”, afinal é filha. Esse vínculo deve ser realizado pela função “appendChild”.

Função “toFixed” determina o número máximo de casas decimais em sua exibição para o usuário

Texto

Descrição gerada automaticamente (abaixo e fora do function)

“tbody querySelector” aparece fora de tudo, fora do function grande

“preventDefault();” =

Função “reset” (“this.reset()” aparece o this pois refere-se ao próprio formulário), deixa o formulário em branco, aparece no final do código. Na sequência, acessamos, por meio da lista campos, o objeto que representa o input do nome do paciente (pois é o primeiro index do array input/campos), e chamamos a função “focus” para retornar o foco das entradas (mouse e teclado) para esse input.

Bootstrap: utiliza-se frameworks de JS como o jQuery e Popper.js que são dependências do Bootstrap

Glyphicons -> no body coloca <span class=”glyphicon glyphicon-btc”>

No css: span.glyphicon {font-size como quiser} para mexer no tamanho

Container-fluid é uma variação do container, não precisa aparecer no html “class=”container container-fluid””

Col = column no grid: col-xs = celular, col-sm = tablet, col-md = pc, col-lg = smartviews

Tem 12 colunas no row, então se tiver 2 conteúdos/colunas no row, a soma deles precisa dar 12. Então col-xs-6 e col-xs-8 não pode

Modulo 2 capítulo 3 aula 3 - fala sobre harmonização de cores

Recolor/Coolors – sites de paleta de cores

Sites legais:

<https://www.le28-lille.com/>

<https://calem.pt/>

<https://takearecess.com/>

Letras com serifa ajudam na leitura

Font-family normalmente usa ”safe combination”, quando não aceitar uma fonte, usa outra, se não outra. Ex: Arial, Helvetica, sans-serif;

Shorthand/Simplificação do font: font-style, font-weight, font-size, font-family;

Whatfontis.com – site para descobrir qual font é a partir de uma imagem

Texto

Descrição gerada automaticamente font baixada

Pseudo-classe é o estado que a tag está, ex: h1:hover, “:hover” é a pseudo-classe

Texto

Descrição gerada automaticamente quando passa o mouse na div/conteúdo, o parágrafo filho aparece (display: block;)

Pseudo-elemento tem 2x “:”, ex: a::after,

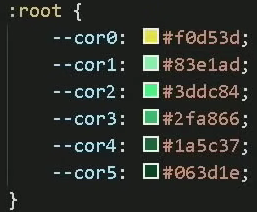
Texto

Descrição gerada automaticamente com confiança baixa valores padrões das tags, se quebra ou não linha

Atenção: se fizer uma imagem de borda, não tira borda normal, coloca ela no código também

MockFlow – site de fazer wireframe para sites

Tag abbr quando coloca o mouse e deixa em cima aparece uma mensagem do sistema, uma explicação ou tradução etc.

 variáveis

Body min e max width no css padrão, margin auto, img width 100% = responsividade básica

Atenção: uma img não é um bloco, então margin auto não é suficiente pra centralizar, precisa de display block também

List-style-type no css –

Centralizar vídeos do youtube no site – coloca dentro de uma div e no css da div coloca position relative e Absolute, com % no width e height

GIT e GITHUB

Repositório (lugar onde você guarda arquivos) local e remoto, local é na máquina e remoto é na nuvem

GitHub tem nome parecido, mas não tem relação com Git, o GitHub é um lugar/servidor para hospedar os projetos, igual o netlify. GitHub faz um backup dos projetos na nuvem, da para compartilhar, outras pessoas verem etc

Gerenciador de versões – criar versões do site, porque mais tarde pode desistir de alguma alteração e querer voltar atras e não saber como. Git é um gerenciador de versões. Mas vai manter dentro da própria máquina

Push é expressão para colocar o arquivo do local no remoto, git para o github