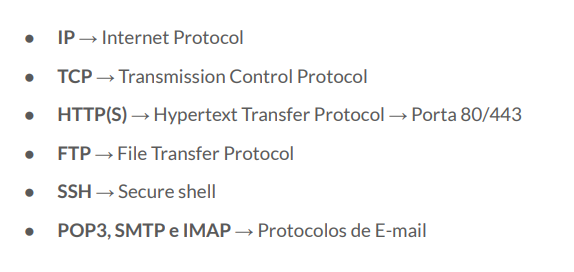
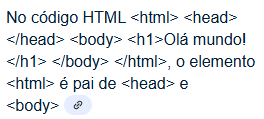
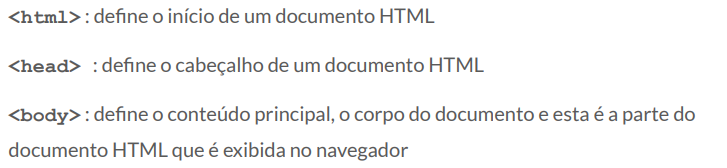
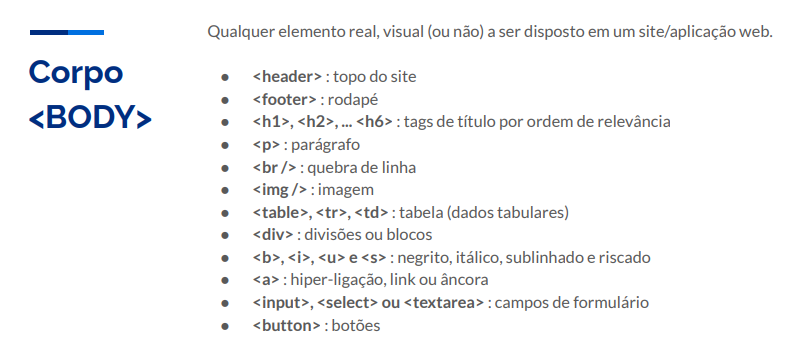
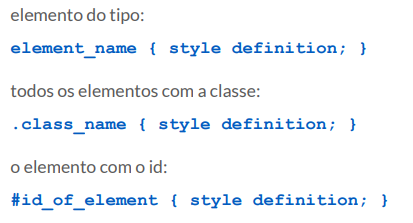
**FUNDAMENTOS DA WEB & INTRODUÇÃO**

* Cada ponto da rede têm um endereço IP
* **BACKBOONE**: espalhados pelo mundo para distribuir Internet – cabos gigantes
* Empresas que contratam e se ligam ao sinal de backboones para distribuir dados aos seus usuários - oi, vivo, claro
* **DNS**: domain name systwm – computador que relaciona o endereço nominal com o endereço real (IP)
* E cada domínio também indica um servidor DNS responsável por gerenciar suas rotas
* **PROTOCOLO** :controla e possibilita uma conexão, comunicação e transferência entre dados de dois sistemas computacionais
* Os protocolos podem ser implementados pelo hardware, software ou por uma combinação dos dois - aparecem em Inicializar, formatar, transmitir e finalizar uma mensagem
* 
* **HTTP**: Hipermídia → Hipertexto (texto com links)
* Www: roda na porta 80
* Primeiro método HTTP, protocolo simples de texto, método get
* **HTTPS**: é o HTTP com segurança - roda em outra porta 443
* **HTML**: HyperText Markup Language, ou linguagem de marcação - Tim Berners-Lee junto da WWW
* HTML possui marcadores
* Um elemento é formado por um nome de marcador (tag), atributos, valores e filhos
* Os elementos filhos são os elementos que estão contidos dentro de outro elemento, chamado de pai
* 
* Umelemento com filho é aquele que abre e fecha - <a> </a>
* <meta charset = ”utf-8” />
* 
* 
* **HTML5** : acessibilidade
* **CSS**: Cascading Style Sheets ou folha de estilos em cascata
* Documentos inline ou externos
* 
* **DISPLAY** - define como um elemento aparece na página — se ocupa uma linha inteira (block), fica na mesma linha (inline), é flexível (flex), invisível (none), etc. É o jeito que ele se comporta no layout
* **POSITION** - define como um elemento é posicionado na página:

static (padrão, segue o fluxo normal – ex: um parágrafo comum),

relative (pode ser movido em relação a onde estaria – ex: top: 10px sobe 10px),

absolute (posicionado em relação ao pai com position – ex: menu flutuante),

fixed (fica preso na tela ao rolar – ex: botão “voltar ao topo”),

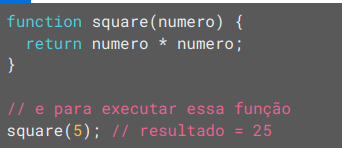
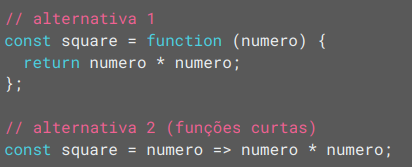
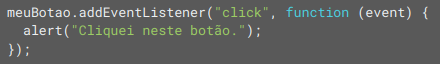
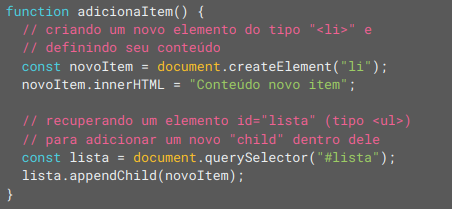
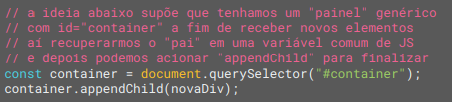
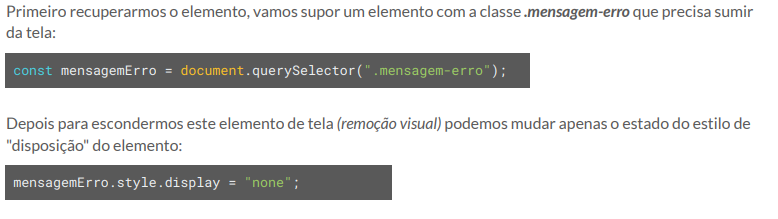
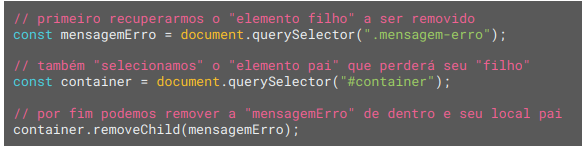
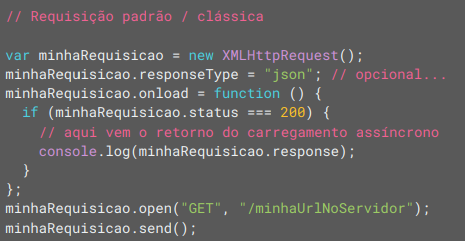
sticky (gruda ao rolar até certo ponto – ex: cabeçalho que fixa no topo)

* **Pseudo-classes (ex: :hover, :focus)**

Definem um estado especial de um elemento.  
 Ex: button:hover muda o botão quando o mouse passa por cima,  
 input:focus muda o campo quando está selecionado para digitar.

* **transition** cria animações suaves quando uma propriedade muda, como em button:hover { background-color: red; } combinado com transition: background-color 0.3s ease;, fazendo o botão mudar de cor suavemente ao passar o mouse, ou em div:hover { transform: scale(1.1); } com transition: transform 0.2s;, que faz a caixa aumentar de tamanho de forma
* Acessibilidade web – strong: negrito, em: itálico, span: sublinhado

**JS**

* Uma linguagem de **programação interpretada**
* Baseada em **ECMAScript \***
* Versão estável **ES2015 (ES5)**
* Funções que são embarcadas ou incluídas em páginas HTML e que interagem com o **Modelo de Objeto de Documentos (DOM)**
* Protótipos - mecanismo de **herança** semelhante a classes
* **Funções e métodos** - não há distinção entre a definição de uma função e a definição de um método
* Interação com **elementos de uma página HTML** (DOM)
* **Requisitar dados** e informações do servidor **sem recarregar a página (AJAX)**
* **var -** declara uma variável
* **let -** declara uma variável local de escopo do bloco
* **const -** declara uma constante de escopo de bloco, apenas de leitura
* 
* 
* 
* 
* 
* 
* 
* 
* 
* 
* 
* 
* 
* 
* 