

1 - O que é um servidor web? Faça uma pesquisa e descreva qual a sua utilidade, se é software ou hardware e qual serviço ele presta.

Um servidor web é um sistema que armazena, processa e entrega páginas da web aos usuários.

Hardware: É o computador físico (servidor) onde os dados e aplicações da web estão armazenados.

Software: É o programa instalado nesse hardware que gerencia as requisições e responde aos usuários.

A principal função de um servidor web é:

Receber e responder a solicitações de clientes (normalmente navegadores), entregando páginas HTML, arquivos CSS, imagens, vídeos e outros conteúdos.

Executar aplicações web (como sites dinâmicos que usam linguagens como PHP, Python ou JavaScript).

Gerenciar conexões seguras, cookies, sessões de usuário e outros aspectos técnicos da navegação na web.

2 - O que é um navegador, ou browser? Para que serve e quais são os principais navegadores do mercado. descreva também as vantagens e desvantagens de cada um.

Um navegador, ou browser, é um programa de software usado para acessar e visualizar páginas na internet. Ele interpreta o código das páginas (HTML, CSS, JavaScript, etc.) e as apresenta de forma visual e interativa para o usuário.

O navegador serve para: acessar sites, navegar pela web, executar aplicações web e garantir segurança e privacidade

Navegador	Vantagem	Desvantagem
Google Chrome	- Rápido e estável	- Alto consumo de memória RAM
	- Suporte amplo a extensões	- Pode comprometer a privacidade do usuário
	- Atualizações frequentes	

Mozilla Firefox

- Foco na privacidade
- Consome mais RAM que alguns concorrentes
- Boa performance
- Algumas páginas têm compatibilidade menor
- Código aberto

Microsoft Edge

- Integrado ao Windows
- Ainda tem resistência por parte de usuários por causa do antigo Internet Explorer
- Baseado no mesmo motor do Chrome (Chromium)
- Menor uso de RAM que o Chrome

Safari (Apple)

- Ótima integração com o sistema macOS/iOS
- Disponível apenas para dispositivos Apple
- Eficiência energética
- Menos extensões disponíveis

Opera

- Leve e rápido
- Menor compatibilidade com alguns sites
- Recursos embutidos como VPN gratuita e bloqueador de anúncios
- Comunidade e suporte menores

3 - O que é HTTP? Para que serve?

HTTP significa HyperText Transfer Protocol. É o protocolo de comunicação usado na internet para transferir dados entre navegadores (como Chrome, Firefox) e servidores web.

4 - O que é HTML? Explique a relação entre HTML, servidores Web e Navegadores.

O HTML é a linguagem usada para criar páginas da web. Ele organiza os conteúdos como títulos, textos, imagens e links.

Você digita um site no navegador (ex: www.exemplo.com).

O navegador usa HTTP para pedir a página ao servidor web.
O servidor envia o código HTML da página.

O navegador lê o HTML e exibe o site para você

5 - O que é uma tag html? Explique como o seu comportamento pode ser definido a partir dos seus atributos.

Uma tag HTML é um comando usado para marcar e organizar elementos dentro de uma página web. Ela define o que cada parte do conteúdo representa, como títulos, parágrafos, imagens, links, botões, entre outros.

Atributos são uma configuração dentro da tag que altera seu comportamento ou aparência. Os atributos controlam como as tags se comportam: o que mostram, como se apresentam e como o usuário interage com elas.

Sem atributos, as tags têm um comportamento padrão.

Com atributos, é possível personalizar totalmente esse comportamento.

6 - O que é uma URL? Quais são as partes que compõem uma URL? Explique cada uma delas.

URL significa Uniform Resource Locator. É o endereço completo de um recurso na internet, como uma página da web, imagem, vídeo ou arquivo.

ex: <https://www.exemplo.com/pagina1/index.html?busca=livros#secao2>

Parte	Exemplo	O que faz
1. Protocolo	https://	Define como o navegador vai se comunicar com o servidor. Ex: http ou https (mais seguro)
2. Domínio	www.exemplo.com	Nome do site ou servidor onde está o conteúdo. Pode ter subdomínios (www, blog, etc.)
3. Caminho	/pagina1/index.html	Mostra em qual pasta ou arquivo do servidor está o recurso
4. Parâmetros	?busca=livros	(opcional) Envia informações para o site, como em formulários de busca
5. Fragmento	#secao2	(opcional) Leva o usuário a uma parte específica da página (âncora)

7 - Qual a diferença entre URLs relativas e absolutas? Como esses conceitos podem ser usados na construção de páginas web?

Uma URL absoluta é o endereço completo de um recurso, com todas as informações desde o protocolo até o caminho final.

Uma URL relativa é parcial, ou seja, mostra apenas o caminho em relação à página atual. Depende da posição do arquivo atual, não precisa do domínio nem do protocolo e é usada para ligar arquivos dentro do mesmo site (páginas, imagens, scripts, etc.).

8 - Cite pelo menos 3 exemplos de como as URLs podem ser usadas no desenvolvimento de páginas web.

Você pode usar uma URL para fazer um botão ou texto levar o usuário a outra página.

As URLs são usadas para indicar o local da imagem que será exibida no site.

É comum usar URLs para linkar arquivos externos que ajudam na aparência e funcionalidades do site.

9 - Qual a diferença entre os elementos `<p>` e `<pre>` em HTML?

`<p>` Serve para criar um parágrafo de texto, o navegador ignora espaços extras, quebras de linha e tabulações dentro do texto. O texto é formatado automaticamente, com espaçamento padrão entre parágrafos.

`<pre>` Serve para exibir texto exatamente como foi escrito, mantendo espaços, quebras de linha e tabulações. Usado para mostrar códigos, poesia, ASCII art, ou qualquer texto que precise de formatação fixa. O texto dentro fica em fonte monoespaçada (como a de código).

10 - Qual a diferença entre o conteúdo das tags `<head>` e `<body>` em html?

`<head>` Contém informações invisíveis para o usuário, mas importantes para o navegador, buscadores e funcionamento do site. Serve para configurar a página, carregar arquivos externos, definir título e metadados.

`<body>` Contém todo o conteúdo visível ao usuário: textos, imagens, vídeos, botões, formulários, etc. É o que realmente aparece na tela quando o site é carregado.

11 - O que acontece, ao abrir uma página web, quando a mesma contém uma tag html é usada de forma errada?

Quando uma tag HTML é usada de forma incorreta, o navegador tenta interpretar o código mesmo assim, mas isso pode causar problemas de exibição, funcionamento ou até de acessibilidade.

12 - O que é um site? Onde um site pode ficar armazenado?

Site é um conjunto de páginas web organizadas e interligadas, que podem conter textos, imagens, vídeos, links, formulários, entre outros conteúdos. Essas páginas são acessadas por meio de um navegador. O Site fica armazenado em servidores e para que todos possam ver o site, ele precisa estar hospedado.

13 - Como se pode alterar a aparência (cor, tamanho borda...) de elementos HTML? Dê exemplos.

pode se alterar a aparência dos elementos HTML usando CSS.

Você pode aplicar CSS de 3 formas principais:

CSS **Inline**; Aplica o estilo diretamente no elemento HTML usando o atributo `style`.

```
<p style="color: blue; font-size: 18px;">Texto azul e maior</p>
```

CSS **Interno**; Os estilos são definidos no cabeçalho da página e aplicados a vários elementos.

```
<head>

  <style>

    h1 {

      color: red;

      border: 2px solid black;

      font-size: 30px;

    }

  </style>

</head>

<body>

  <h1>Título com borda e cor vermelha</h1>

</body>
```

CSS **Externo**; Mais organizado, ideal para sites maiores. O arquivo CSS é separado e incluído no HTML com a tag `<link>`.

```
<head>

  <link rel="stylesheet" href="estilo.css">

</head>

<body>

  <div class="caixa">Conteúdo</div>
```

```
</body>
```

```
.caixa {  
  
  background-color: lightgray;  
  
  border: 1px solid #333;  
  
  padding: 20px;  
  
  font-size: 16px;  
  
}
```

14 - O que é CSS? Descreva a sintaxe das regras de folhas de estilo.

CSS significa Cascading Style Sheets. É a linguagem usada para estilizar páginas web.

Com o CSS, é possível definir cores, fontes, tamanhos, margens, posicionamento e muito mais para os elementos HTML. Ele separa o conteúdo (HTML) da apresentação visual (CSS).

A **sintaxe das regras CSS** é composta de seletor, propriedade e valor.

ex:

```
seletor {  
  
  propriedade: valor;  
  
}
```

Seletor

- Identifica qual elemento HTML será estilizado.
- Exemplos: h1, .classe, #id.

Declaração

- Contém uma **propriedade** e um **valor** separados por : e terminados com ;.
- Exemplo: color: red;

Propriedade

- Indica o **aspecto a ser estilizado** (cor, tamanho, borda, etc.).
- Exemplo: color, font-size.

Valor

- Define **como a propriedade será aplicada**.
- Exemplo: red, 16px.

15 - Qual a diferença entre CSS interno, externo e embutido? Dê um exemplo de cada um.

embutido: Aplicado diretamente dentro da tag HTML, usando o atributo style.

interno: Escrito dentro do próprio arquivo HTML, entre a tag <style>, geralmente no <head>

externo: Fica em um arquivo separado com extensão .css. É o mais organizado, ideal para sites maiores e reutilização de estilos em várias páginas. O arquivo é vinculado com a tag <link> no <head> do HTML.

16 - Quais são os tipos de listas disponíveis em HTML?

Lista Ordenada(): Os itens são numerados automaticamente (1, 2, 3...). Usada quando a ordem importa, como em receitas, etapas, rankings etc.

Lista Não Ordenada (): Os itens aparecem com marcadores (bolinhas). Usada quando a ordem não importa, como em listas de compras, tópicos, etc.

Lista de Definição (<dl>): Usada para apresentar termos e suas definições, como em glossários ou descrições.

17 - Qual a diferença entre as tags TR, TH e TD no uso de tabelas em HTML?

<tr> — Table Row (Linha da Tabela)

- Define uma linha na tabela.
- Dentro dela vão as células da linha (<th> ou <td>).

`<th>` — Table Header (Cabeçalho da Tabela)

- Cria uma célula de cabeçalho.
- O conteúdo geralmente aparece em negrito e centralizado.
- Usada dentro da `<tr>` para representar títulos das colunas.

`<td>` — Table Data (Dado da Tabela)

- Cria uma célula de dados (conteúdo comum da tabela).
- Usada dentro da `<tr>` para representar os dados das colunas.

18 - O que é GIT? Qual a sua utilidade no desenvolvimento de páginas web?

Git é um sistema de controle de versão. Ele permite que desenvolvedores acompanhem todas as alterações feitas no código de um projeto ao longo do tempo.

No desenvolvimento de páginas web, ele torna o processo mais seguro, organizado e colaborativo

19 - Explique o que é um repositório.

Um repositório é um local onde ficam armazenados os arquivos de um projeto, junto com todo o seu histórico de alterações.

20 - Cite três comandos básicos para o uso de ferramentas de controle de versão (GIT) e explique cada um deles.

`git init`

- O que faz:
Cria um novo repositório Git vazio na pasta atual.
- Quando usar:
Quando você quer começar a usar o Git em um projeto novo ou existente que ainda não tem controle de versão.

`git add`

- O que faz:
Adiciona arquivos ou mudanças específicas à área de stage (preparação) para serem confirmadas (committed).

- Quando usar:
Depois de editar arquivos, você usa git add para preparar essas alterações para o próximo commit.

git commit

- O que faz:
Salva (confirma) as mudanças adicionadas com o git add no histórico do repositório. Cada commit é uma “foto” do estado do projeto naquele momento.
- Quando usar:
Sempre que quiser registrar um ponto do desenvolvimento com uma mensagem descritiva.

21 - Explique o que é necessário, e quais são os passos para tornar uma página web acessível para "a internet".

Escolher onde seu site vai ficar armazenado, transferir os arquivos do site para o servidor e verificar se a página está acessível.

22 - Explique as diferenças entre os tipos de "inputs" para formulários HTML.

Tipo	Descrição	Uso comum
text	Campo para texto simples	Nome, endereço, etc.
password	Campo para senha; caracteres ficam ocultos	Senhas e dados confidenciais
email	Campo para e-mail; valida formato de email	Cadastro e login
number	Campo para números; pode ter limites	Idade, quantidade
checkbox	Caixa de seleção que permite marcar várias opções	Aceite de termos, preferências
radio	Botões para escolha exclusiva entre opções	Sexo, seleção única
submit	Botão para enviar o formulário	Enviar dados para o servidor

file	Permite seleccionar arquivos para upload	Envio de documentos ou imagens
date	Campo para seleção de data com calendário	Data de nascimento, compromissos
color	Seleção de cor via paleta	Escolha de cores
range	Controle deslizante para escolher valor numérico	Volume, intensidade, avaliações

23 - Quais são os principais atributos do elemento input?

Atributo	Função
type	Define o tipo do campo (text, password, email, etc.)
name	Nome do campo, usado para identificar os dados enviados no formulário
value	Valor inicial ou padrão do campo
placeholder	Texto exibido dentro do campo quando vazio (dica para o usuário)
required	Indica que o campo é obrigatório para envio do formulário
readonly	Campo só leitura, o usuário não pode editar
disabled	Campo desativado, não pode ser usado nem enviado
maxlength	Limita o número máximo de caracteres que podem ser digitados
min	Valor mínimo permitido (para tipos como number, date)
max	Valor máximo permitido (para tipos como number, date)

step	Define o intervalo para números ou datas (ex: 1, 0.1)
autocomplete	Controla se o navegador sugere preenchimento automático
checked	Marca o input como selecionado (usado em checkbox e radio)

24 - Para que servem os atributos "action" e "target" de um form?

action

ex: <form action="processa.php" method="post">

- **Para que serve:**
Define a **URL** (endereço) para onde os dados do formulário serão enviados quando o usuário clicar em “Enviar”.

target

ex: <form action="processa.php" target="_blank">

- **Para que serve:**
Define **onde** a resposta do servidor será exibida após o envio do formulário.

25 - Qual a utilidade do atributo "method" de um form? Quais são os valores possíveis para ele e qual a utilidade de cada um?

Utilidade: Define o **método HTTP** que o navegador vai usar para enviar os dados do formulário ao servidor.

Valor	Descrição	Uso comum
GET	Envia os dados pela URL (aparece na barra de endereço)	Ideal para buscas, filtros, quando os dados não são sensíveis

POST	Envia os dados no corpo da requisição , invisível na URL	Usado para enviar dados confidenciais, cadastros, formulários com muitas informações
-------------	---	--