**Aula 06: Formulários e Github Pages**

Texto

O conteúdo gerado por IA pode estar incorreto.Adicionaremos alguns elementos de formulário que vocês ainda não viram. A estrutura básica do nosso formulário de cadastro, antes de começarmos, seria algo assim:

**O elemento <select>**

Texto

O conteúdo gerado por IA pode estar incorreto.Usado para criar uma lista suspensa de opções. Cada opção é definida pela tag <option>. O atributo **value** é o que será enviado quando o formulário for submetido.

**O elemento <input type="radio">**

Texto

O conteúdo gerado por IA pode estar incorreto.Os botões configurados dessa forma permitem que o usuário escolha apenas uma opção de um conjunto. Para que eles funcionem em grupo, todos devem ter o mesmo atributo **name**.

**O elemento <input type="checkbox">**

Texto

O conteúdo gerado por IA pode estar incorreto.Diferente do anterior, os checkbox permitem selecionar uma ou várias opções de um conjunto.

**O elemento <input type="color">**

Este é um seletor de cores nativo do HTML, muito útil para personalização de ambientes.

**O elemento <input type="file">**

Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.Permite que o usuário envie um arquivo do seu computador.

**O elemento <textarea>**

Para caixas de texto de múltiplas linhas, como comentários ou mensagens longas.

**Atividade Prática:**

Voltando para a Situação de Aprendizagem, vocês terão de finalizar a página de cadastro/contato do site da empresa JWC. Usem o que aprenderam hoje para construir um formulário completo e funcional, explorando todos os elementos que vimos na demonstração.

**Diferença entre os métodos GET e POST no GitHub**

Ao enviar um trabalho para o GitHub, podemos aproveitar para entender a diferença prática entre os métodos **GET** e **POST**.

Uma forma de fazer isso é com a criação de um formulário de exemplo:  
  
Crie um formulário simples que submeta dados para um serviço web de teste, como o *https://httpbin.org/get* e o [*https://httpbin.org/post*](https://httpbin.org/post).

Texto

O conteúdo gerado por IA pode estar incorreto.**Formulário com método GET:**

Ao clicar em "Enviar (GET)", a URL na barra de endereço mudará para:

*https://httpbin.org/get?nome=Jo%C3%A3o&email=joao%40example.com*

Observem como os dados (nome e email) são visíveis diretamente na URL.

Texto

O conteúdo gerado por IA pode estar incorreto.**Formulário com método POST:**

Ao clicar em "Enviar (POST)", a URL não muda. Os dados (nome e email) são enviados no corpo da requisição HTTP, de forma invisível para o usuário na barra de endereço. Isso é o que o torna mais seguro para informações confidenciais.

**GitHub Pages**

Para finalizar o encontro de hoje, temos um guia de como usar o **GitHub Pages** para hospedar seu site (*landing page*) de forma gratuita, direto do seu repositório.

**Passo 1: Preparar seu Repositório**

Primeiro, você precisa ter o código do seu site em um repositório no GitHub.

1. **Crie um novo repositório:** Acesse o GitHub e clique no botão **"New"** para criar um novo repositório.
2. **Nomeie o repositório:** Dê um nome para o seu repositório. Para um site pessoal (por exemplo, meunome.github.io), o nome do repositório deve ser seu-username.github.io. Se for um projeto (por exemplo, projeto-site), o nome pode ser o que você quiser. Para um perfil profissional, pode ser ‘meuportfolio’ (no final, teremos ‘seu-username.github.io/meuportfolio’).
3. **Adicione seu código:** Faça o upload dos arquivos do seu projeto (HTML, CSS, imagens etc.) para o repositório. O arquivo principal, que será a sua página inicial, precisa se chamar **index.html**. **ATENÇÃO para modificar os caminhos de diretório dos arquivos estáticos nas suas páginas em html!**

**Passo 2: Ativar o GitHub Pages**

1. **Acesse as configurações:** No seu repositório, clique na aba **"Settings"** (Configurações - fica no menu superior, último ícone).
2. **Vá para "Pages":** No menu lateral esquerdo, clique em **"Pages"**.
3. **Escolha a branch:** No campo **"Branch"**, selecione a branch que contém o seu código. Na maioria dos casos, a branch padrão é a **main**.
4. **Salve as configurações:** Clique no botão **"Save"**.

**Passo 3: Acesse seu Site**

Após salvar as configurações, o GitHub leva alguns minutos para processar os arquivos e publicar o seu site.

1. **Obtenha a URL:** Na mesma página **"Pages"**, após alguns minutos, a URL do seu site publicado aparecerá. A URL seguirá este formato:
   * **Para sites pessoais:** https://seu-username.github.io
   * **Para sites de projetos, profissionais:** https://seu-username.github.io/nome-do-repositorio