

Exercício sobre modal

- 1) Crie uma pasta chamada modal em httdocs do XAMPP.
- 2) Nessa pasta crie o arquivo index.html conforme imagem abaixo.

```
1  <!DOCTYPE html>
2  <html lang="pt-br">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Meu modal</title>
7  </head>
8  <body>
9      <h1>Modal Simples</h1>
10     <button id="btnAbrirModal">Abrir modal</button>
11     <!-- O dialog é nosso modal -->
12     <dialog id="modalDialog">
13         <div id="divFormModal">
14             <h2>Meu Formulário</h2>
15             <hr>
16             <div id="">
17                 <form id="modalForm" action="">
18                     <label for="nome">nome</label>
19                     <input type="text" id="nome"/>
20                     <label for="email">E-mail</label>
21                     <input type="text" id="email"/>
22                     <input type="reset" value="Limpar">
23                     <input type="submit" value="Enviar">
24                 </form>
25             </div>
26             <hr>
27         </div>
28         <input type="button" id="btnFecharModal" value="Fechar">
29     </dialog>
30 </body>
31 <script></script>
32 </html>
```

- 3) Agora vamos manipular nosso modal e os botões de abrir e fechar via JavaScript. Entre as tags script escreva o código para recuperar e manipular os elementos.

```
31 <script>
32 //Recuperando os botões
33 const btnModalAbrir = document.querySelector("#btnAbrirModal");
34 const btnModalFechar = document.querySelector("#btnFecharModal");
35 const modalForm = document.querySelector("#modalForm");
36 //Recuperando o modal
37 const modal = document.querySelector("#modalDialog");
38
39 //fazendo o modal abrir
40 btnModalAbrir.onclick=function(){
41     modal.showModal();
42 }
43 //Fazendo o modal fechar
44 btnModalFechar.addEventListener('click',()=>{
45     modal.close();
46 });
47 //Ao submeter o formulário
48 modalForm.addEventListener('submit',(e)=>{
49     e.preventDefault();
50     modal.close();
51 })
52 </script>
```

- 4) Uma coisa interessante a se fazer é tornar o fundo opaco. Para isso, vamos criar um pequeno código css para nosso modal. Crie as tags style por enquanto e coloque o seguinte código css entre elas. Ao final, faça o teste.

```
6      <style>
7          #modalDialog{
8              margin: 2px solid black;
9          }
10         #modalDialog::backdrop{
11             background-color: grey;
12         }
13     </style>
14     <title>Meu modal</title>
```

No primeiro trecho colocamos a borda do modal sólida, preta e com 2 pixels.

No segundo trecho colocamos o fundo cinza.

- 5) Algumas pessoas acham interessante fazer com que o modal se feche ao clicar fora dele. Para isso basta acrescentar o seguinte código em nosso Javascript. Ao final, faça o teste.

```
60 //Fechar ao clicar fora da área do modal
61 modal.addEventListener('click',(e)=>{
62     //pegando as posições do modal na tela
63     const areaModal = modal.getBoundingClientRect();
64     //Verificando se o clique ocorreu fora do modal
65     //Seu professor pode explicar melhor esse código no quadro
66     if(
67         e.clientX < areaModal.left ||
68         e.clientX > areaModal.right ||
69         e.clientY < areaModal.top ||
70         e.clientY > areaModal.bottom
71     ){
72         dialog.close();
73     }
74 });
```

- 6) Crie um arquivo modal.js, recorte todo esse código e coloque nesse arquivo.
7) Em seguida, referencie esse arquivo utilizando a diretiva defer. Pensando no futuro, aproveite para definir o type=module.

```
13     </style>
14     <script src="modal.js" type="module" defer></script>
```

- 8) Recorte todo o código referente ao formulário e coloque dentro de um arquivo formulário.html. Posteriormente carregaremos esse conteúdo no formulário de id modalForm. Veja como o arquivo deve ficar.

```
2     <label for="nome">nome</label>
3     <input type="text" id="nome"/>
4     <label for="email">E-mail</label>
5     <input type="text" id="email"/>
6     <input type="reset" value="Limpar">
7     <input type="submit" value="Enviar">
```

- 9) Agora vamos criar um arquivo útil.js e dentro dele criar uma função assíncrona para carregar um conteúdo via fetch com async await. Não se esqueça de exportar essa função.

```

JS util.js > ...
1  async function preencherHtml(recurso, elemento){
2      const meuFetch = await fetch(recurso);
3      const resposta = await meuFetch.text();
4      elemento.innerHTML = resposta;
5  }
6  export {preencherHtml}

```

10) Importe essa função em modal.js e chame a função ao abrir o modal. Tenha especial atenção às linhas 1 e 11.

```

1  import { preencherHtml } from "../util.js";
2  //Recuperando os botões
3  const btnModalAbrir = document.querySelector("#btnAbrirModal");
4  const btnModalFechar = document.querySelector("#btnFecharModal");
5  const modalForm = document.querySelector("#modalForm");
6  //Recuperando o modal
7  const modal = document.querySelector("#modalDialog");
8
9  //fazendo o modal abrir
10 btnModalAbrir.onclick=function(){
11     preencherHtml("formulario.html", document.querySelector("#modalForm"));
12     modal.showModal();
13 }

```

11) Agora verifique se tudo funciona corretamente.