

Sweet Alert 2

DigitalHouse >
Coding School



**Certified Tech
Developer**
The Ultimate Degree

Índice

1. Para que serve?
2. Como se instala?
3. Exemplos de uso

1 | Para que serve?

Uma das maneiras que temos de dar feedback a uma pessoa em nosso aplicativo é usando o `alert()`.



O método alert()

```
alert("Olá Mundo!")
```



Este método mostra um alerta de navegador nativo, no qual podemos personalizar apenas o corpo da mensagem.

A principal desvantagem é que **NÃO** nos permite personalizar a interface visual adicionando estilos ou modificando os elementos existentes no alerta.

Sweet Alert 2

Essa biblioteca permite mostrar alertas personalizados aos usuários através de elementos que apresentam uma interface amigável.

Além disso, oferece várias interfaces visuais, o que nos permite personalizar o tipo de alerta conforme a utilização que queremos dar em cada parte da nossa aplicação.

sweetalert2

2 | Como se instala?

Instalação mediante CDN

```
<!DOCTYPE html>
<html lang="pt-br" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Olá Mundo!</title>
  </head>
  <body>
    <script src="//cdn.jsdelivr.net/npm/sweetalert2@11"></script>
  </body>
</html>
```

Inserimos o **script** com o url para inicializar a biblioteca.



Pronto!



Uma vez inicializada a biblioteca, podemos utilizá-la!

3 | Ejemplos de uso

Sintaxe básica

Objeto Swal

É o **objeto global** da biblioteca, que nos dá acesso aos seus métodos.

Mensagem

Aqui colocamos a mensagem que queremos.



```
js Swal.fire('Uma mensagem básica')
```

The diagram illustrates the syntax of the `Swal.fire()` function. It features a dark grey code block containing the text `js Swal.fire('Uma mensagem básica')`. Above the code block, there are three curly braces: an orange one under `Swal`, a grey one under `fire`, and a teal one under the opening parenthesis of the string argument. Lines connect these braces to their respective descriptions: 'Objeto Swal' (orange), 'Mensagem' (grey), and 'Método Fire' (teal).

Método Fire

Esse método nos permite lançar a notificação para o navegador.

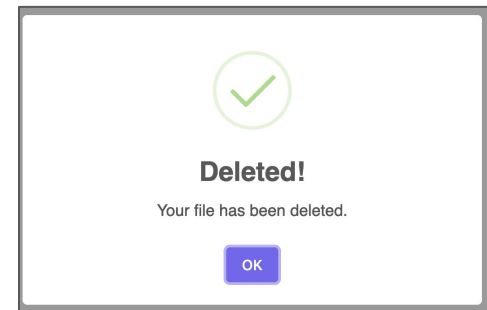
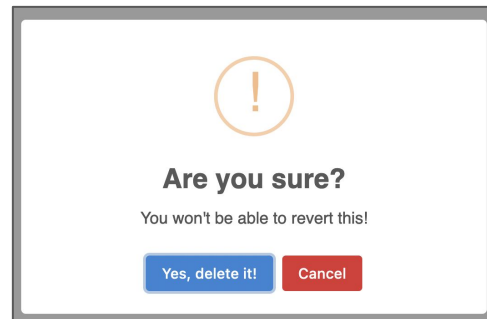
Alerta de êxito

```
// Mostra um alerta de êxito
Swal.fire(
  'Good job!', // Título
  'You clicked the button!', // Mensagem
  'success' // Tipo de ícone
)
```



Modal de confirmação

```
Swal.fire({
  title: 'Are you sure?',
  text: "You won't be able to revert this!",
  icon: 'warning',
  showCancelButton: true,
  confirmButtonColor: '#3085d6',
  cancelButtonColor: '#d33',
  confirmButtonText: 'Yes, delete it!'
}).then((result) => {
  // Validamos se o usuário confirma a ação
  if (result.isConfirmed) {
    Swal.fire(
      'Deleted!',
      'Your file has been deleted.',
      'success'
    )
  }
})
```



API Request

Com SweetAlert, podemos até incluir solicitações para uma API dentro do alerta. Podemos ver um exemplo [aqui](#).

sweetalert2

Documentação



Documentação do Sweet Alert:
<https://sweetalert2.github.io/>

DigitalHouse>
Coding School