Dialogues in the web

I found an interesting thing that would make it eay to create dialogues for web applications.

The `dialog` element represents a dialog box or other interactive component, such as a dismissible alert, inspector, or subwindow.

This post will walk through creating a dialogue using the dialogue element.

To get started we need some HTML on the page.

We need a button we can click to open the dialogue.

```
<button class="button
 open-button">
 open modal
</button>
```

We also need the dialogue that we want to open. Even though the dialog is on the page it won't be displayed until we open it.

The close button inside the dialogue will close the dialogue and return the focust to the parent window.

```
<dialog class="modal" id="modal">
  <h2>An interesting title</h2>
  <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Rerum esse
  <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Atque, quo.
  <button class="button close-button">close modal</button>
</dialog>
```

The interactive part of the dialogue is done in Javascript.

1. We capture references to the dialogue and the open and close buttons.
2. We add a click event listener in the open button to show the dialogue using the showModal() of the dialog element to open the dialogue.
    1. We can control the type of dialogue that we create. If we use show()

instead of `showModal` we'll get a different type of dialogue displayed on the page

3. We also add a click event listener to the close button to close the dialogue using the [close](#) method of the dialog element

    1. The close method is the same regardless of the type of dialogue we open

```
// 1
const modal = document.querySelector("#mo"#modal"onst openModal = document
const closeModal = document.querySele".open-button"utton");

// 2
openModal.addEventListener("click", () => {
  modal.showModal();
});

"click"// 3
closeModal.addEventListener("click", () => {
  modal.close();
});
```

Support for the `dialog` element is pretty good. All major desktop browsers support it as do most major mobile browsers other than Opera Mini.

Dialog element

Method of easily creating custom dialog boxes to display to the user with modal or non-modal option
includes a `::backdrop` pseudo-element for behind the element.

IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser
9	103	103	104	15.5	15.5			4.4
10	104	104	105	15.6	15.6			4.4.4
11	105	105	106	16.0	16.0	all	105	105
		106	107	16.1	16.1			
		107	108	TP				
			109					

✓ ✗ Partial Support

Global: 92.72% +

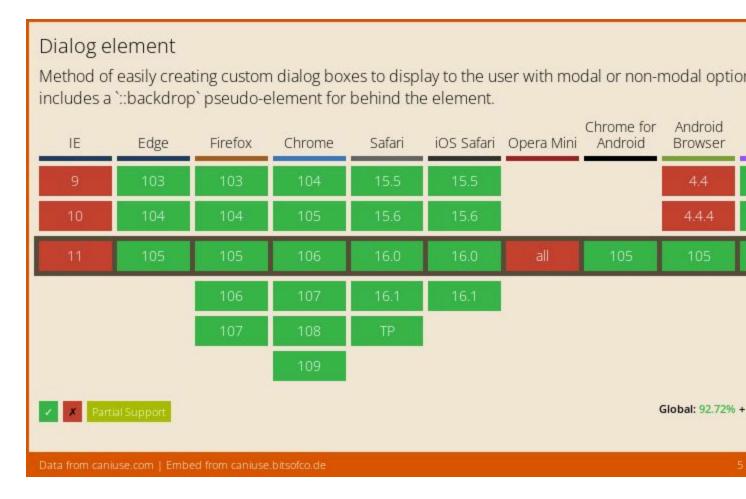Data from caniuse.com | Embed from caniuse.bitsofco.de

5

Figure 1: Data on support for the dialog feature across the major browsers from caniuse.com

We shouldn't need a polyfill for the `dialog` element but, if you need it, one is
[available](#) from the Chrome team.