

Interaction: alert, prompt, confirm

As we'll be using the browser as our demo environment, let's see a couple of functions to interact with the user: `alert`, `prompt` and `confirm`.

alert

This one we've seen already. It shows a message and waits for the user to press "OK".

For example:

```
alert("Hello");
```

The mini-window with the message is called a *modal window*. The word "modal" means that the visitor can't interact with the rest of the page, press other buttons, etc, until they have dealt with the window. In this case – until they press "OK".

prompt

The function `prompt` accepts two arguments:

```
result = prompt(title, [default]);
```

It shows a modal window with a text message, an input field for the visitor, and the buttons OK/Cancel.

title

The text to show the visitor.

default

An optional second parameter, the initial value for the input field.

The square brackets in syntax [. . .]

The square brackets around `default` in the syntax above denote that the parameter is optional, not required.

The visitor can type something in the prompt input field and press OK. Then we get that text in the `result`. Or they can cancel the input by pressing Cancel or hitting the `Esc` key, then we get `null` as the `result`.

The call to `prompt` returns the text from the input field or `null` if the input was canceled.

For instance:

```
let age = prompt('How old are you?', 100);

alert(`You are ${age} years old!`); // You are 100 years old!
```

In IE: always supply a default

The second parameter is optional, but if we don't supply it, Internet Explorer will insert the text "undefined" into the prompt.

Run this code in Internet Explorer to see:

```
let test = prompt("Test");
```

So, for prompts to look good in IE, we recommend always providing the second argument:

```
let test = prompt("Test", ''); // <-- for IE
```

confirm

The syntax:

```
result = confirm(question);
```

The function `confirm` shows a modal window with a `question` and two buttons: OK and Cancel.

The result is `true` if OK is pressed and `false` otherwise.

For example:

```
let isBoss = confirm("Are you the boss?");  
  
alert( isBoss ); // true if OK is pressed
```

Summary

We covered 3 browser-specific functions to interact with visitors:

alert

shows a message.

prompt

shows a message asking the user to input text. It returns the text or, if Cancel button or `Esc` is clicked, `null`.

confirm

shows a message and waits for the user to press "OK" or "Cancel". It returns `true` for OK and `false` for Cancel/`Esc`.

All these methods are modal: they pause script execution and don't allow the visitor to interact with the rest of the page until the window has been dismissed.

There are two limitations shared by all the methods above:

1. The exact location of the modal window is determined by the browser. Usually, it's in the center.

2. The exact look of the window also depends on the browser. We can't modify it.

That is the price for simplicity. There are other ways to show nicer windows and richer interaction with the visitor, but if “bells and whistles” do not matter much, these methods work just fine.