

Theory: Window Object

🕒 9 minutes

0 / 5 problems solved

Skip this topic

Start practicing

602 users solved this topic. Latest completion was about 4 hours ago.

Warm-up game: spot the nearest window! Where are you looking? It's right here: you are reading this topic in one of them! `Window Object` is an object that represents a browser window. It provides the JavaScript browser interaction interface. If an HTML document contains an `<iframe>` tag, the browser creates one `Window Object` for the HTML document and one `Window Object` for each such tag in the document.

Let's take a closer look at the properties and methods of this object.

§1. Properties

One of the features of the `Window Object` is that it is **global**. This means that access to its properties and methods can be carried out from anywhere in the program. Look at an example of accessing a closed property which returns `true` or `false` depending on whether the window is closed or not:

```
1 | console.log(window.closed);
```

It can also be written as follows:

```
1 | console.log(closed);
```

As you see, this can be done without specifying the `Window Object`.

Let's look at another property:

```
1 | console.log(innerHeight);
```

It returns the height of the current window. And here is the `innerWidth` property:

```
1 | console.log(innerWidth);
```

It contains the value of the width of the window.

We couldn't possibly fit all properties here, so we're inviting you to check out the [full list of properties](#) yourself.

§2. Methods

In order to create a new empty browser window, use the following method:

```
1 | open();
```

You can specify the URL as a parameter, for example, open a new window with the Hyperskill website:

```
1 | open("https://hyperskill.org/");
```

To close the current window, use the `close()` method:

```
1 | close();
```

Keep it in mind: JavaScript can only close the window that was opened with it!

Another important method is calling an alert window that contains a message and an `OK` button. For example, try entering the following code in the console:

Current topic:

[Window Object](#) ...

Topic depends on:

✗ [Introduction to Objects](#) ...

Table of contents:

[↑ Window Object](#)

[§1. Properties](#)

[§2. Methods](#)

[§3. Conclusion](#)

[Feedback & Comments](#)

```
1 | alert("Hello Hyperskill!");
```

A similar method is `confirm()`:

```
1 | confirm("Hyperskill is excellent");
```


However, while `alert()` does not return anything, `confirm()` returns `true` if the `OK` button was pressed and `false` if the user clicked `Cancel`.

Again, this is not a complete list of methods, so we invite you to check out [other methods](#).

§3. Conclusion

Today we considered an important type of object, `Window Object`. In the context of browsers, it is important to know that the window object is not part of JS itself (not part of the spec), that is why other environments don't have any `Window Object`. Remember that this object is global, so access to its properties and methods is possible from anywhere in the program and without specifying an object. It has an extensive list of properties and methods, some of which we considered here.

 Report a typo

54 users liked this theory.  didn't like it. What about you?



Start practicing

[Comments \(0\)](#)

[Hints \(0\)](#)

[Useful links \(0\)](#)

[Show discussion](#)