



@CODE.CLASH

JavaScript

BOM

NEXT →

Hey Everyones 🖐️

In this Post, you will learn about JavaScript Browser Object Model (BOM) with the help of examples.

Do Like, save and Share This Post If You Found This Helpful.

BOM

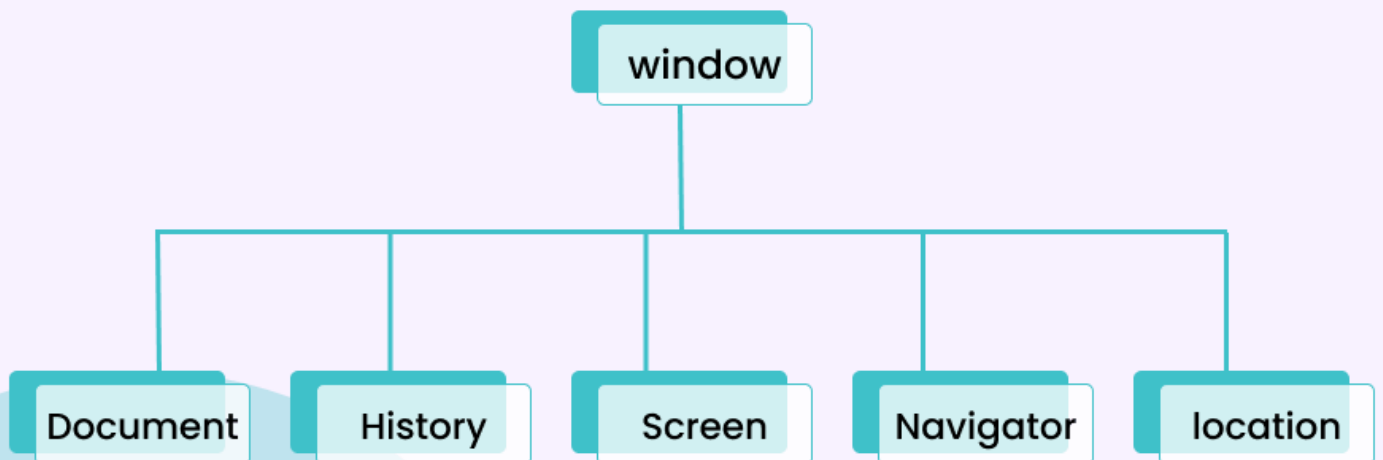
- The Browser Object Model (BOM) is used to interact with the browser.
- The default object of browser is **window** means you can call all the functions of window by specifying window or directly.

```
window.alert("hello javatpoint");
```

- Same as

```
alert("hello javatpoint");
```

- You can use a lot of properties defined underneath the window object like document, history, screen, navigator, location, innerHeight, innerWidth.



- **Note:** The document object represents an html document. It forms DOM, we'll see it in the next post.

Window Object

- The window object represents a window in browser.
- An object of window is created automatically by the browser.
- The important methods of window object are as follows:

Method	Description
alert()	displays the alert box containing message with ok button.
confirm()	displays the confirm dialog box containing message with ok and cancel button.
prompt()	displays a dialog box to get input from the user.
open()	opens the new window.
close()	closes the current window.
setTimeout()	performs action after specified time like calling function, evaluating expressions etc.

Responsive Table

- Example of alert()

```
<input type="button" value="click" onclick="msg()" />
<script type="text/javascript">
function msg(){
    alert("Hello Alert Box");
}
</script>
```

- Example of open()

```
<input type="button" value="javatpoint" onclick="msg()" />
<script type="text/javascript">
function msg(){
    open("http://www.instagram.com");
}
</script>
```

- Example of prompt()

```
<input type="button" value="click" onclick="msg()" />

<script type="text/javascript">
function msg(){
    var v= prompt("Who are you?");
    alert("I am "+v);
}
</script>
```

History Object

- The JavaScript **history object** represents an array of URLs visited by the user.
- By using this object, you can load previous, forward or any particular page.
- There are only **1 property** of history object.
 - `length` - returns the length of the history URLs.
- There are only **3 methods** of history object.
 - a. `forward()` - loads the next page.
 - b. `back()` - loads the previous page.
 - c. `go()` - loads the given page number.

NEXT →

```
history.back();//for previous page
history.forward();//for next page
history.go(2);//for next 2nd page
history.go(-2);//for previous 2nd page
```


Navigator Object

- The JavaScript **navigator object** is used for browser detection.
- It can be used to get browser information such as appName, appCodeName, userAgent etc.
- There are many **properties** of navigator object that returns information of the browser.
 - a. appName - returns the name.
 - b. appVersion - returns the version.
 - c. appCodeName - returns the code name.etc..

```
document.writeln("<br>navigator.appCodeName: "+navigator.appCodeName);  
document.writeln("<br>navigator.appName: "+navigator.appName);  
document.writeln("<br>navigator.appVersion: "+navigator.appVersion);  
document.writeln("<br>navigator.language: "+navigator.language);
```


Screen Object

- The JavaScript **screen object** holds information of browser screen.
- It can be used to display screen width, height, colorDepth, pixelDepth etc.
- There are many **properties** of screen object.
 - a. width - returns the width of the screen
 - b. height - returns the height of the screen
 - c. availWidth - returns the available width etc..

```
document.writeln("<br>screen.width: "+screen.width);  
document.writeln("<br>screen.height: "+screen.height);  
document.writeln("<br>screen.availWidth: "+screen.availWidth);  
document.writeln("<br>screen.availHeight: "+screen.availHeight);  
document.writeln("<br>screen.colorDepth: "+screen.colorDepth);  
document.writeln("<br>screen.pixelDepth: "+screen.pixelDepth);
```

@CODE.CLASH

THANKS FOR YOUR ATTENTION



CODE.CLASH



IMTIYAZ NANDASANIYA

**LIKE AND SAVE IT
FOR LATER**