Web Technology
(KCS-602)
Unit 3
Functions, Objects

Prepared By

Abhishek Kesharwani

Assistant Professor,UCER Naini,Allahabad

JavaScript Functions

- A function is a reusable code-block that will be executed by an event, or when the function is called.
- To keep the browser from executing a script when the page loads, you can put your script into a function.
- A function contains code that will be executed by an event or by a call to that function.
- You may call a function from anywhere within the page (or even from other pages if the function is embedded in an external .js file).

JavaScript Function Syntax

function *functionname*()
{
some code to be executed
}

JavaScript Functions

A function is a block of code that will be executed when "someone" calls it.

```
<!DOCTYPE html>
<html>
<head>
<script>
function myFunction()
{
alert("Hello World!");
}
</script>
</head>
<body>
<button onclick="myFunction()">Try it</button>
</body>
</html>
```

Functions With a Return Value

- Sometimes you want your function to return a value back to where the call was made.
- This is possible by using the *return* statement.
- When using the *return* statement, the function will stop executing, and return the specified value.

Calling a Function with Arguments

- When you call a function, you can pass along some values to it, these values are called *arguments* or *parameters*.
- These arguments can be used inside the function.

```
<button onclick="myFunction(‘Abhishek ',’Professor')">click here</button>

<script>
function myFunction(name,job)
{
alert("Welcome " + name + ", the " + job);
}
</script>
```

Functions With a Return Value

- Sometimes you want your function to return a value back to where the call was made.
- This is possible by using the *return* statement.
- When using the *return* statement, the function will stop executing, and return the specified value.

Syntax

```
function myFunction()
{
var x=5;
return x;
}
```

Examples

```
<html>
<head>
<script type="text/javascript">
function product(a,b)
{
return a*b
}
</script>
</head>
<body>
<script type="text/javascript">
document.write(product(4,3))
</script>
<p>The script in the body section calls a function with two parameters
    (4 and 3).</p>
<p>The function will return the product of these two parameters.</p>
</body>
</html>
```

Browser Object Model

The **Browser Object Model** (BOM) is used to interact with the browser.

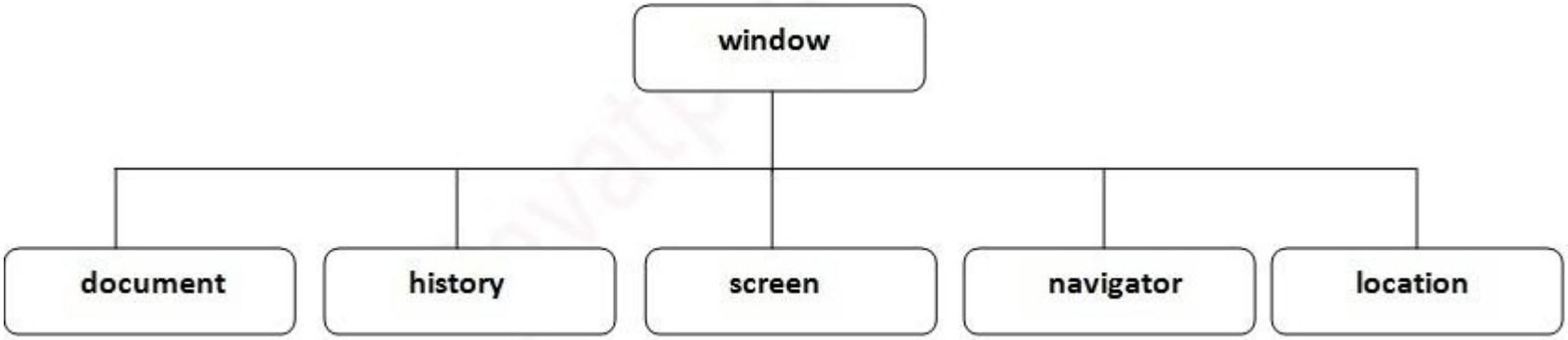
The default object of browser is window means you can call all the functions of window by specifying window or directly. For example:

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

is same as:

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

Browser Object Model



Window Object

- The **window object** represents a window in browser. An object of window is created automatically by the browser.
- Window is the object of browser, **it is not the object of javascript**. The javascript objects are string, array, date etc.

Window Object

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.

JavaScript Popup Boxes

- In JavaScript we can create three kinds of popup boxes: Alert box, Confirm box, and Prompt box.

Alert Box

An alert box is often used if you want to make sure information comes through to the user.

When an alert box pops up, the user will have to click "OK" to proceed.

Syntax:

```
alert("sometext")
```

Confirm Box

A confirm box is often used if you want the user to verify or accept something.

When a confirm box pops up, the user will have to click either "OK" or "Cancel" to proceed.

If the user clicks "OK", the box returns true. If the user clicks "Cancel", the box returns false.

Syntax:

confirm("sometext")

Prompt Box

A prompt box is often used if you want the user to input a value before entering a page.

When a prompt box pops up, the user will have to click either "OK" or "Cancel" to proceed after entering an input value.

If the user clicks "OK" the box returns the input value. If the user clicks "Cancel" the box returns null.

Syntax:

prompt("sometext","defaultvalue“)

JavaScript Objects

- In real life, a car is an object.
- A car has properties like weight and color, and methods like start and stop:

Properties	Methods	
	car.name = Fiat car.model = 500 car.weight = 850kg car.color = white	car.start() car.drive() car.brake() car.stop()

```
<p id="demo"></p>
<script>
// Create an object:
const person = {firstName:"John", lastName:"Doe",
age:50, eyeColor:"blue"};
// Display some data from the object:
document.getElementById("demo").innerHTML =
person.firstName + " is " + person.age + " years
old.";
</script>
```

JavaScript Date Object

The **JavaScript date** object can be used to get year, month and day.

Method	Description
getFullYear()	Get **year** as a four digit number (yyyy)
getMonth()	Get **month** as a number (0-11)
getDate()	Get **day** as a number (1-31)
getDay()	Get **weekday** as a number (0-6)
getHours()	Get **hour** (0-23)
getMinutes()	Get **minute** (0-59)
getSeconds()	Get **second** (0-59)
getMilliseconds()	Get **millisecond** (0-999)

```
<html>
<body>
Current Date and Time: <span id="txt"></span>
<script>
var today=new Date();
document.getElementById('txt').innerHTML=today;
</script>
</body>
</html>
```

JavaScript Math Object

- The **JavaScript math** object provides several constants and methods to perform mathematical operation. Unlike date object, it doesn't have constructors.

```
<!DOCTYPE html>
<html>
<body>
Round of 4.3 is: <span id="p6"></span><br>
Round of 4.7 is: <span id="p7"></span>
<script>
document.getElementById('p6').innerHTML=Math.round(4.3);
document.getElementById('p7').innerHTML=Math.round(4.7);
</script>
</body>
</html>
```

```
<!DOCTYPE html>
<html>
<body>

Absolute value of -4 is: <span id="p8"></span>
<script>
document.getElementById('p8').innerHTML=Math.abs(-4);
</script>

</body>
</html>
```