

Hot tip

Just as object functions are known as “methods” their variables are known as “properties”.

Passing function arguments

Functions and variables are the key components of JavaScript.

A function may be called once or numerous times to execute the statements it contains. Those functions that belong to an object, such as `document.write()`, are known as “methods” – just to differentiate them from user-defined functions. Both have trailing parentheses that may accept “argument” values to be passed to the function for manipulation. For example, the text string value passed in the parentheses of the `document.write()` method that gets written into the HTML document.

The number of arguments passed to a function must match those specified within the parentheses of the function block declaration. For example, a user-defined function requiring exactly one argument looks like this:

```
function function-name ( arg )
{
  // Statements to be executed go here.
}
```

Multiple arguments can be specified as a comma-separated list:

```
function function-name ( argA, argB, argC )
{
  // Statements to be executed go here.
}
```

Like variable names, function names and argument names may comprise letters, numbers, and underscore characters, but may not contain spaces or begin with a number. Additionally you must avoid the JavaScript keywords, reserved words, and object names listed in the tables on page 9.

Optionally a function can return a value to the caller using the `return` keyword at the end of the function block. After a return statement has been made the script flow continues at the caller – so no further statements in the called function get executed. It is typical to return the result of manipulating passed argument values back to the caller:

```
function function-name ( argA, argB, argC )
{
  // Statements to be executed go here.

  return result ;
}
```

Don't forget

Multiple arguments must also be passed as a comma-separated list.

...cont'd

1 Create a HTML document that embeds an external JavaScript file and has a "panel" element

```
<script type="text/javascript" src="argument.js"></script>
<div id="panel"><noscript>
  <div>! JavaScript is Not Enabled.</div></noscript>
</div>
```



argument.html

2 Open a plain text editor and add a function to execute after the document has loaded

```
function init()
{
}
window.onload=init;
```



argument.js

3 In the function block, insert a statement that calls another user-defined function and passes it four argument values

```
document.getElementById("panel").innerHTML=
  stringify( "JavaScript", "In", "Easy", "Steps" );
```

4 Next insert a second statement that also calls the user-defined function, passing it four different argument values

```
document.getElementById("panel").innerHTML+=
  stringify( "Written", "By", "Mike", "McGrath" );
```

5 Now, before the init function block, declare the function being called from the statements within the init function

```
function stringify( argA, argB, argC, argD )
{
  var str=argA+" "+argB+" "+argC+" "+argD+"<br>";
  return str;
}
```

6 Save the script alongside the HTML document then open the page in your browser to see the returned values

Beware



A function must have been declared before it can be called so function declarations should appear first in the script.

Function Arguments

JavaScript Console Panel

JavaScript In Easy Steps
Written By Mike McGrath