

JavaScript Callback Function

Kristina Darroch

What is a Callback Function?

A callback function, also known as a higher-order function, is a function that is passed to another function (let's call this other function “otherFunction”) as a parameter, and the callback function is called (or executed) inside the otherFunction. A callback function is essentially a pattern (an established solution to a common problem), and therefore, the use of a callback function is also known as a callback pattern.

How do Would I Use a Callback function in my Code?

However javascript is an event driven language. If `do_a` takes longer than `do_b`, the result of `do_b` comes out first than `do_a`;

So how do we make sure `do_b` comes out after `do_a` in that situation? This is where callbacks comes in handy.

How do Callback Functions work?

JavaScript statements are executed line by line. However, with effects, the next line of code can be run even though the effect is not finished. This can create errors.

To prevent this, you can create a callback function.

A callback function is executed after the current effect is finished.

Typical syntax: `$(selector).hide(speed,callback);`

Example

The example below has a callback parameter that is a function that will be executed after the hide effect is completed:

```
$( "button" ).click( function() {  
  
    $( "p" ).hide( "slow", function() {  
  
        alert( "The paragraph is now hidden" );  
  
    } );  
  
} );
```

Example without callback

The example below has no callback parameter, and the alert box will be displayed before the hide effect is completed:

```
$( "button" ).click( function() {  
  
    $( "p" ).hide( 1000 );  
  
    alert( "The paragraph is now hidden" );  
  
} );
```

Sources

- http://www.w3schools.com/jquery/jquery_callback.asp
- <http://javascriptissexy.com/understand-javascript-callback-functions-and-use-them/>
-

Callback function:

- When a function is given and before the function returns the call, it does something before returning.

(MILK EXAMPLE)