

Week 5 Glossary


Printer-friendly version



Online users



(last 1 minutes: 1)



Jurgen Mane

Search

☒ Search full text

Add a new entry

Browse the glossary using this index

Special |
 A |
 B |
 C |
 D |
 E |
 F |
 G |
 H |
 I |
 J |
 K |
 L |
 M |
 N |
 O |
 P |
 Q |
 R |
 S |
 T |
 U |
 V |
 W |
 X |
 Y |
 Z |
 ALL

A

Arrow functions

Arrow functions were introduced in ES6 JavaScript and have a shorter syntax than function expressions or function declarations.

Aside from syntax differences, they do not have their own *this* binding.

- MDN - Arrow functions

C

Const

Const is a new variable declaration keyword introduced in ES6. It is block scoped. Variables declared with const must be initialized at declaration, and cannot be reassigned.

- MDN - Const

E

Event handlers

Event handlers are typically functions that are registered to a DOM node intended to respond to a particular event firing. Two ways to register event handlers are by using the *onevent* handlers, or event listeners. Some documentation may specifically refer to *onevent* handlers when speaking of event handlers. Generally speaking, event handlers are the code that responds an event, and the functions that `addEventListener` runs when the event it's listening for is fired off may be referred to as event handlers as well.

- MDN - Introduction to events

Event listeners

Event listeners are added to and removed from DOM nodes using the `addEventListener` and `removeEventListener` methods. They specify a particular event to listen for, and a function to run when the event is fired. The function that's run when the event is fired may be referred to as the event handler. Thus, event listeners may be considered as a way to register event handlers, similar to the *onevent* handlers.

- MDN - Event listeners

Events

Events are occurrences that can happen within a system that can be observed and used to trigger responses. In the DOM/browser, that includes events such as a document finishing loading, a button being clicked, a window being scrolled or changed in size, et cetera. There are dozens of different DOM/browser events.

- MDN - Introduction to events

F

Function declarations

Function declarations are the original way to create a function in JavaScript. A function declaration begins with the function keyword, followed by a name for the function, a parameter list in parentheses, then the function body. Function declarations are hoisted.

- MDN - Function declarations

Function expressions

Function expressions came after function declarations in JavaScript, but before arrow functions. With function expressions, you use the function keyword to define a function within an expression. Function expressions can be named or anonymous, and they are not hoisted.

- MDN - Function expressions

L

Let

Let is a variable declaration keyword in JavaScript that was introduced in ES6. It is block scoped.

- MDN - Let

S

Scope

Scope refers to the section of code for which a variable or function is accessible.

- [MDN - Scope](#)

V



Var

Var is the original variable declaration keyword in JavaScript. It is function scoped. It is largely deprecated in favor of the use of the new variable declaration keywords, let and const.

- [MDN - Var](#)

[Course page](#)[Previous](#)[Next](#)