

Mark McCreery

Front-End Course 7.27

August 23, 2022

Week 4 Research

Var can be either be globally scoped or scoped within a function depending on where the variable is declared. If declared within a function, it will only be referenced within the scope of the function. Also, var is the only one of the three that can be redeclared without causing any sort of error. Let variables are locally scoped and won't be able to be accessed outside the block of code that they're contained in. They can be updated within the same code as well, however you cannot redeclare the same let variable within the same block of code. Const variables are the strictest in their actions. Like let variables, const variables are block scoped. Also, they cannot be updated or redeclared; const variables must also be initialized at declaration as well.

A callback is essentially a function that acts upon a function to complete a routine or action. Usually these are used in asynchronous steps that have to pull information from an external source or code. Promises act in the same way. The main difference between the two is that promises are objects that are set up to take either a completion or a failure in an asynchronous function. You can use callbacks in the same way, but promises are more streamlined and make the code easier to read.

Works Cited

Atuonwu, Sarah Chima. "Var, Let, and Const – What's the Difference?" *FreeCodeCamp.org*, FreeCodeCamp.org, 28 Apr. 2021, <https://www.freecodecamp.org/news/var-let-and-const-whats-the-difference/>.

"Callback Function - MDN Web Docs Glossary: Definitions of Web-Related Terms: MDN." *MDN Web Docs Glossary: Definitions of Web-Related Terms / MDN*, https://developer.mozilla.org/en-US/docs/Glossary/Callback_function.

"Using Promises - Javascript: MDN." *JavaScript / MDN*, https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Using_promises.