1. **What are the differences between var, let, and const?**

   While var, let and const are similar in that they are all used for declaring variables, there are a few major differences. Var specifically has a global scope when it is declared outside of a function, meaning it can be used anywhere in the code. Although when it is declared in the function, it can only be used inside that function. They can also be declared again and re-assigned within the same scope. Let on the other hand is blocked scoped, which means that if it is defined within a block, or a set of curly braces, whether it is a function, if/else statement, loop, etc., once that block is closed off that variable will no longer be defined. Let can be re-assigned, but not redeclared, unlike var, thus giving us less chance of accidentally re-using the same variable name, unless it is within a different scope. Const is similar to let in that it is block scoped. The difference between const and both let and var is that const cannot be re-assigned or redeclared. This is useful when you don't want the user to change the value of a specific variable.

   https://www.freecodecamp.org/news/var-let-and-const-whats-the-difference/

2. **What are the differences between callbacks and promises?**

   Callbacks and promises work very similarly, in the sense that they both use a function as a parameter for another function. A big difference is in syntax, where callback looks similar to a function, using parameters, returns, and braces, promises have a simpler syntax. Promises just take in one parameter, which is the function being called in, and the return value gets evaluated by .then or .catch. This makes a more legible code for promises where we can use multiple .then methods where callback syntax would create multiple blocks with indented braces, called callback hell, making it harder to read and debug.

   https://betterprogramming.pub/callbacks-vs-promises-in-javascript-1f074e93a3b5

3. **What are some features that are new with ES6?**

   ES6 is the second big update for JavaScript made in 2015. There were a lot of new features added in this update. This includes the following: classes, symbol, array.from(), array keys, new number properties, and one I found very surprising map objects, mainly because we've been told about these in class and they seem to be very useful in coding that it's surprising this was barely added to JavaScript in 2015. Specifically in this week's lesson we learned about let and const keywords and how they vary from var. Arrow functions, which allow us to write a function in less lines, but still legible and understandable code. Template literals is another update that was brought up with ES6. This makes logging out any phrase very simple. It is also a lot cleaner because it is similar to writing on a text editor. You do have to remember to use ${} to output any variable you need, but other than that just using backticks is all you need compared to concatenating various strings and variables. I find using template literals as well as arrow functions more convenient to use because it really condenses the lines of code into a simple view.

   https://www.w3schools.com/js/js_es6.asp