Your question to answer:

1. What is a closure, and why would you use one?

Answer to the question you are asking:

1. What is the callback queue in JavaScript, and how does it work?

The callback queue is one piece in the larger structure of how functions are run in JavaScript. The overall structure includes the stack, web API, callback queue, and render queue. Specifically, the callback queue is where functions are pushed from the web API when they are ready to run. As code executes, it gets pushed onto the stack. If the functions in the stack do something relevant to the web API, such as adding a click listener or using setTimeout, that code will be popped off the stack and pushed into the web API. When the web API recognizes that it needs to run one of the associated callbacks (e.g., when a click happens or when the setTimeout expires), it will push that callback into the callback queue. Callbacks in the callback queue will get pushed onto the stack one by one **when the stack is empty.** Essentially, the callback queue is a way of managing asynchronous functions so that they are executed when the synchronous thread is available.

