

# Module 5 – JavaScript Asynchronous

How do we handle asynchronous  
stuff in JavaScript



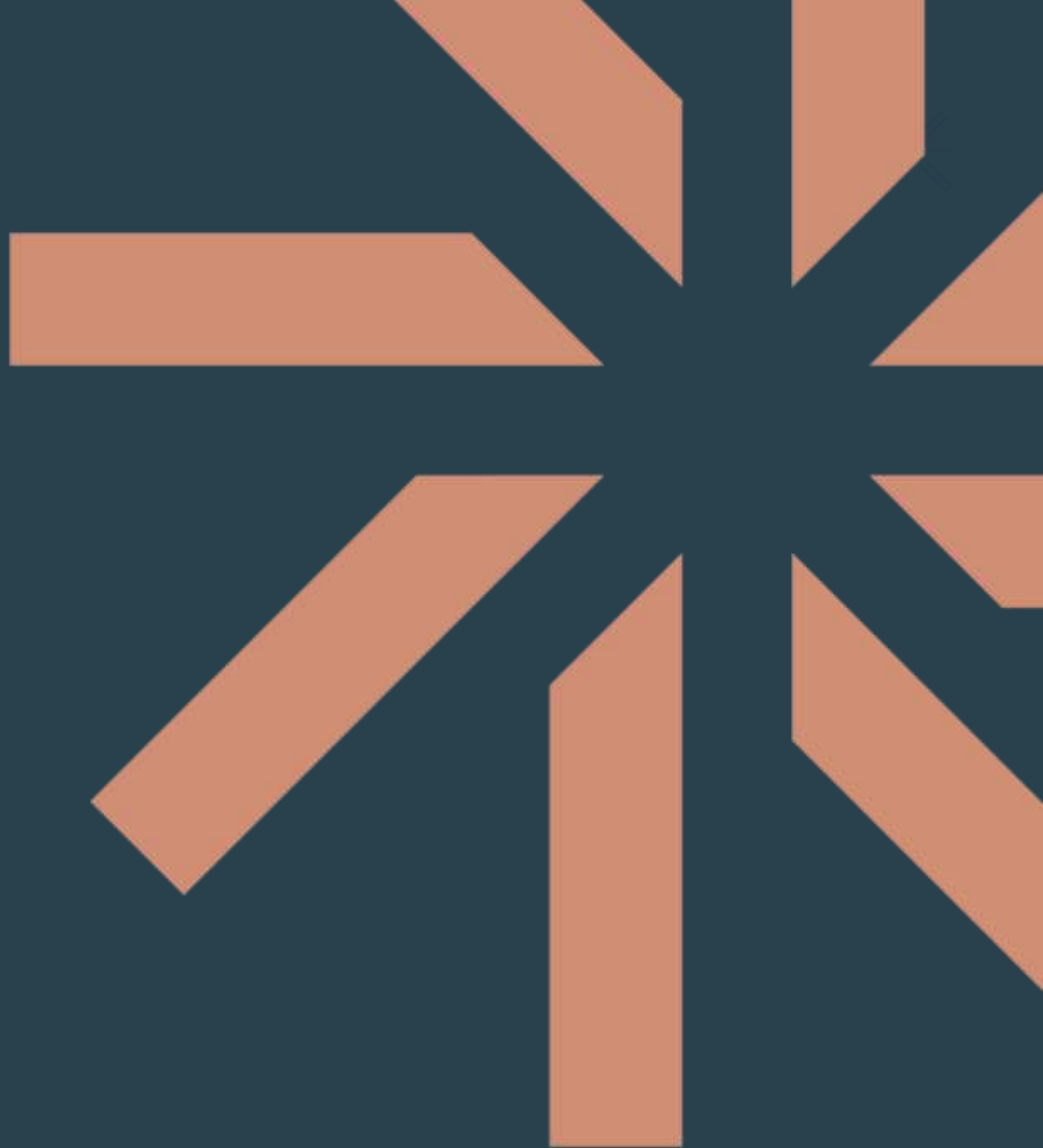
# From last class

- When is our JavaScript script executed?
- Where is JavaScript code executed?
- Why is JavaScript not stuck when we call `SetTimeout()`?
- How does the browser help the JavaScript execution?
- What is a callback function?



# Class plan

- What is callback hell
- Avoiding callback hell with promises
- Promises resolving
- Async/Await in JavaScript



# Callback hell



- Asynchronous execution is when multiple stuff are executed at the same time
- Callbacks are great for making sure asynchronous code executes in the right order
- Callbacks can sometimes be hard to organize, write and read
- A huge callback tree is called callback hell

# How to avoid callback hell



- Promises are a great way to handle asynchronous tasks
- Promises are easy to read and organize
- A promise is resolved when we get an answer from the task we are doing asynchronously
- Async/Await – A nicer and shorter way of handling asynchronous tasks
- Async/Await still works with promises in the background



Questions?