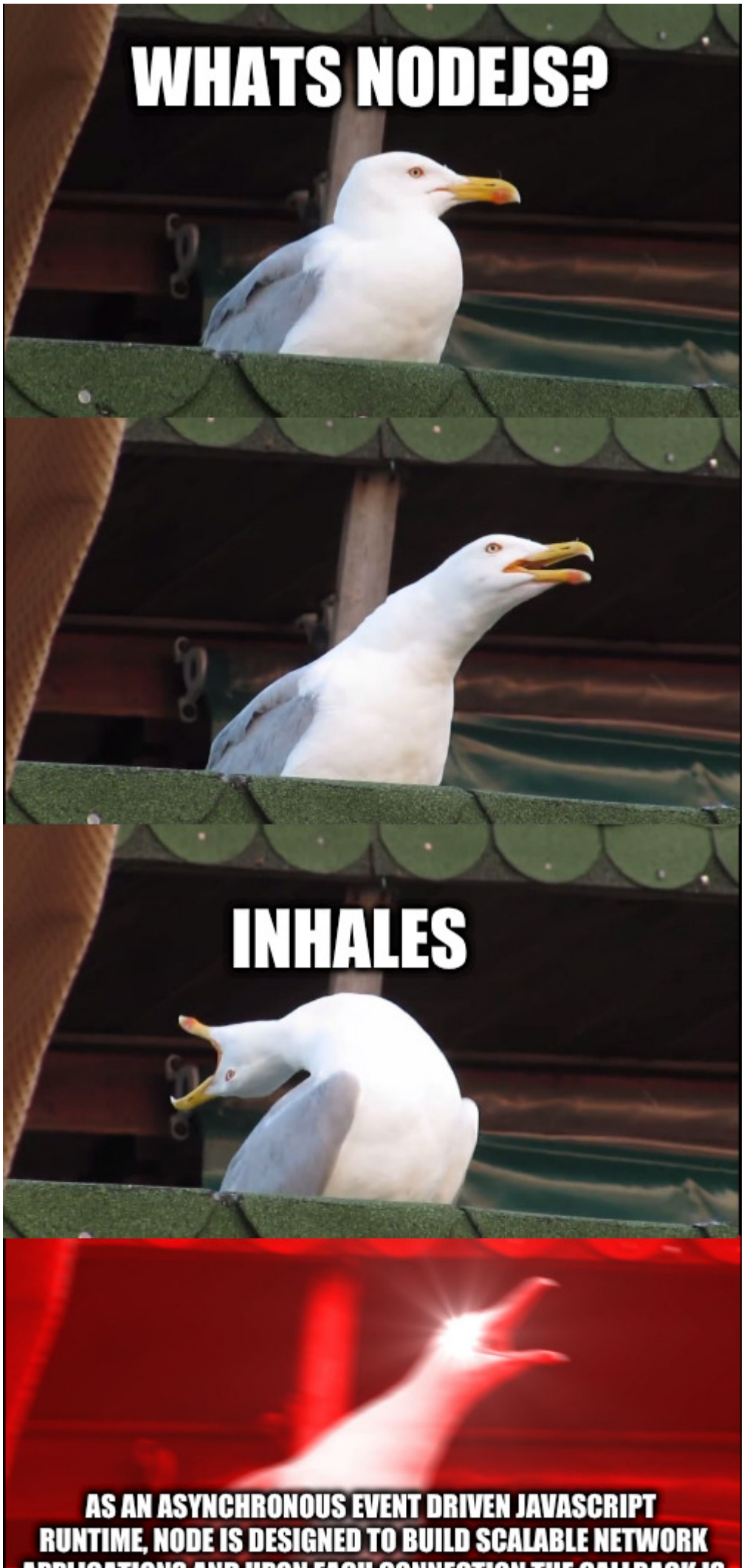
NodeJS

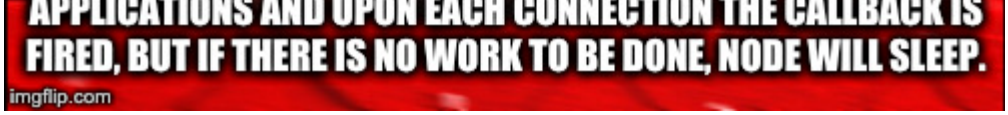
In this lesson, we're going to get an introduction to server-side programming in JavaScript.

WE'LL COVER THE FOLLOWING

What is Node.js?

The official Node.js website says, “As an asynchronous event-driven JavaScript runtime, Node is designed to build scalable network applications. In the following “hello world” example, many connections can be handled concurrently. Upon each connection, the callback is fired, but if there is no work to be done, Node will sleep.”


WHATS NODEJS?
INHALES
AS AN ASYNCHRONOUS EVENT DRIVEN JAVASCRIPT
RUNTIME, NODE IS DESIGNED TO BUILD SCALABLE NETWORK



You're probably thinking, "...???"

What they actually mean to say is that Node is a runtime environment that can run JavaScript outside of browsers. Also that servers written in Node can handle multiple connections at once, which makes them more efficient.

Why Node?

Node is great for software prototyping i.e., building incomplete versions of something that demonstrate what it is meant to do eventually. Node is also incredibly fast and highly scalable, which means that it is great to use in the industry. Furthermore, it uses JavaScript, which is a language that most people are already familiar with, so writing in Node is easy. Also, the code written for Node is usually cleaner and more consistent. Lastly, there is a large ecosystem of open-source libraries available for Node.

A sample Node JS server

Here is an example of a Node server that can also be found on the website.

```
const http = require('http');

const hostname = '0.0.0.0';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World\n');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

In the next chapter, we will talk about web development frameworks.