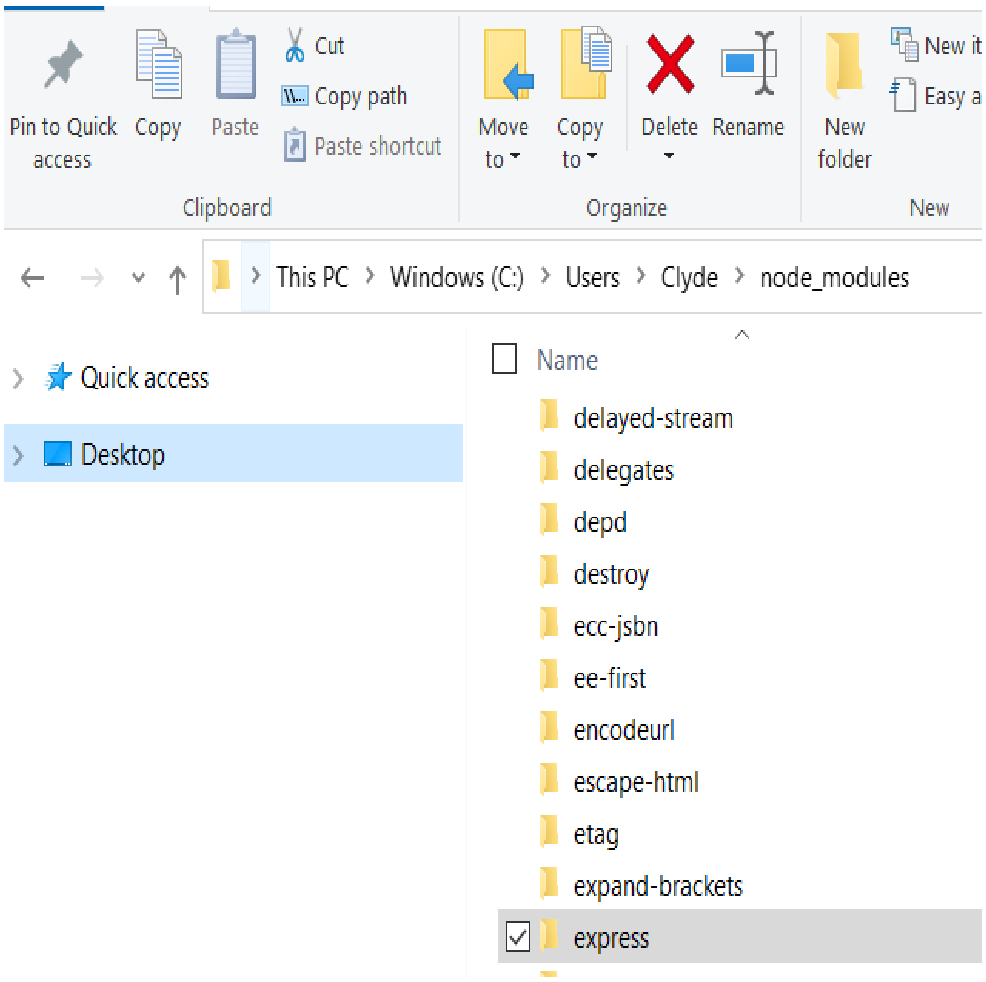
This is a little advanced, I will admit, but if you are curious then read on...

Let's look at the example of this line of code that we have written in our Node file:

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

Where does Node know where to find this module? As I mentioned earlier, I have not installed the Express framework locally in my project folder. I have Express installed globally on my machine.

Here is where it is located on my machine:



This PC › Windows (C:) › Users › Clyde › node_modules

delayed-stream
delegates
depd
destroy
ecc-jsbn
ee-first
encodeurl
escape-html
etag
expand-brackets
express

Where does Node look to find the required module?

Let me say, off the bat, that Node.js uses a complex algorithm to locate the requested module.

It first determines whether the named module is what's known as a **core module**. A core module is installed directly when you install Node. It comes shipped with Node. Examples of core modules are the HTTP module, FS module, and Events module.

Examples of core modules are the HTTP module, FS module, and Events module. There are a lot more.

These core modules will load first and will therefore load very fast in Node’s loading algorithm.

“But Clyde, what if the named module is not a core module? Where does Node then look?”

Great question.

Our Express module is not a core module.

In this case, Node will then look for a directory named **"node_modules"**. It will start in the current directory. Remember a few lectures back, in the article, I mentioned that you have the option of installing Express locally in your folder. If it’s not there, the Node will work its way up the folder hierarchy, checking each level for a **node_modules** folder.

In the preceding screenshot, Node will continue to search until it finds Express here:

`C:\Users\Clyde\node_modules`

This will cover most instances of modules. However, if Node still doesn’t find it, it does a few other checks that I don’t want to discuss here.

Bottom line: Node performs a complex searching algorithm when trying to find a module that you require.