Matt Strayer

Week 5 Node.js Practice Questions

Web Applications

Coded Examples can be found on my GitHub Account using this link:

**Explain what is an error first callback?  Please provide a coded example.**

A callback is an asynchronous equivalent for a function. A callback function is called after the completion of a task. This allows other code to run in the meantime without waiting on the function to load data, thus the need for a callback. In an **error first callback**, the first argument for the callback function would be an error object, followed by an argument for the data. If an error occurs during operations, that call back function is called with an Error object passed as the first argument, if no errors, then the first argument will be passed as null.

**Explain what Promises are.  Please provide a coded example.**

A promise is essentially a placeholder or proxy for a value that isn’t known when the promise is created. This lets asynchronous functions return values like a synchronous function would, but instead of returning the final value, it returns a promise to provide that value in the future. It’s in one of three states; pending, fulfilled or rejected. It starts out in the pending phase and can then either be rejected with an error or fulfilled with a value.

**What is a test pyramid?  Please give an example.**

A test pyramid is essentially a set of guidelines that describes some of the tests you should put in place for your applications. In the test pyramid, there are unit tests, integrations or service tests, and end-to-end or UI tests. At the base are unit tests, above them are integration tests, and at the top are end-to-end tests. You should have more integration tests than end-to-end tests, and more unit tests than integration tests. Unit tests involve the smallest testable part of code, like a function. These are faster than the UI tests at the top of the pyramid.