Discussion Board 6.1 JavaScript Promises

Shane Hingtgen

2/6/23

Bellevue University

What is a JavaScript Promise? A JavaScript Promise is an object that serves as the inevitable failure or completion of the asynchronous JavaScript syntax and the value it results to. (*Promise - JavaScript | MDN*, 2022) According to MDN docs the promise is a substitute for a value that may not be known. It is used in the asynchronous syntax to create an eventual future value returned. (*Promise - JavaScript | MDN*, 2022)

A promise is composed of three states pending, fulfilled or rejected. In the pending stat it is the first state that is not fulfilled or rejected. In the fulfilled stage it means that is was executed successfully. Then finally the third stage, if the operation did not execute successfully is the rejected stage, which means it was not successful and it failed. The promise will only be completed once it is either fulfilled or rejected and not in pending stage (*Promise - JavaScript | MDN*, 2022) Below is a flowchart from MDN Docs showing how it is sequenced.

Diagram

Description automatically generated

(*Promise - JavaScript | MDN*, 2022)

This is a basic syntax, Using Promise() you can create a new promise.

new Promise((resolveOuter) => {

resolveOuter(

new Promise((resolveInner) => {

setTimeout(resolveInner, 1000);

})

);

});

There are other variations you can add with Promise, such as Promise.all(), which will wait until each promise has be complete or if any of the promises are rejected.

Promise.allSettled() which will wait for all the prmoises to be settled, complete or rejected

Promise.reject() which will return the promise that has been rejected with the reason.

Promise.resolve() returns a promise that has been fulfilled with the value.

References:

*Promise - JavaScript | MDN*. (2022, December 13). MDN Web Docs. Retrieved February 6, 2023, from https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\_Objects/Promise