In JavaScript, Promises are objects that represent completion or failure of asynchronous operations and their resulting values. A Promise has 3 states: pending (initial, neither fulfilled nor rejected), fulfilled (the operation completed!!), and rejected (the operation failed!!). Promises eliminate the need for multiple callback functions and help simplify code. They are also very good at handling errors. The following snippet of code checks isMomHappy and either resolves the promise (she is) or rejects the promise (she isn’t).

Snippet:

var isMomHappy = false;

var willIGetNewPhone = new Promise(

function (resolve, reject) {

if (isMomHappy) {

var phone = {

brand: 'Samsung',

color: 'black'

};

resolve(phone); // fulfilled

} else {

var reason = new Error('mom is not happy');

reject(reason); // reject

}

}

);

Async functions are asynchronous operations that return Promises. They operate in a separate order than the rest of the code, letting the code pass them by until their Promise is resolved. Await functions pause the execution of the async functions until the Promise is resolved. Await is only valid inside async. While async is paused due to await, the code that calls the async function continues running. In the code snippet below, showOff creates a new promise that can only be resolved. Then, the askMom function (an async function) awaits willIGetNewPhone and showOff. If they are successful, it outputs the message from showOff, and if not it outputs the error message from the previous willIGetNewPhone function.

async function showOff(phone) {

return new Promise(

(resolve, reject) => {

var message = 'Hey friend, I have a new ' +

phone.color + ' ' + phone.brand + ' phone';

resolve(message);

}

);

};

// call our promise

async function askMom() {

try {

console.log('before asking Mom');

let phone = await willIGetNewPhone;

let message = await showOff(phone);

console.log(message);

console.log('after asking mom');

}

catch (error) {

console.log(error.message);

}

}

(async () => {

await askMom();

})();