Redo yesterday's lab - Lab 5 using Async & Await

You may still use Promise, add new methods to finish the lab.

Exercise 01 changes:

After your making the change, your method call will look like below:

console.log('start');

isPrimeAsync(5);

console.log('end');

The output in console:

start

end

{prime:true}

const promise = new Promise(function(resolve, reject) {

    setTimeout(() => {

        for (let i = 2, s = Math.sqrt(this.num); i <= s; i++) {

            if (this.num % i === 0) {

                reject({ prime: false });

            }

        }

        resolve({ prime: this.num > 1 });

    }, 500);

});

function isPrime(num) {

    return promise;

}

async function isPrimeAsync(num) {

    let result = await isPrime(num);

    console.log(result);

}

console.log('start');

isPrimeAsync(7);

console.log('end');

async function removeDuplicates(arr) {

    let temp = [];

    let j = 0;

    for (let i = 0; i < arr.length - 1; i++) {

        if(uniqe(arr[i],i, arr)){

            temp.push(arr[i]);

        }

    }

     return temp;

}

function uniqe(num, loc, arr){

    for (let i = loc+1; i < arr.length; i++) {

        if (arr[i] === num) {

            return false

        }

    }

    return true;

}

Array.prototype.removeDuplicatesAsync = async function() {

    let arr = this;

    let result = await removeDuplicates(arr);

    console.log(result);

}

console.log(`start`);

[4, 1, 5, 7, 2, 3, 1, 4, 6, 5, 2].removeDuplicatesAsync();

console.log(`end`);