**LAB1**

**EXERCISE 1.- Write the necessary Node script to make this code work for all arrays:**

**[1,2,3,4,5,6,7,8].even(); // [2,4,6,8] [1,2,3,4,5,6,7,8].odd(); // [1,3,5,7]**

**Test your code in Node.JS CLI**

function getEven() {

return this.filter(n => n % 2 == 0);

}

function getOdd() {

return this.filter(n => n % 2 !== 0);

}

Array.prototype.even = getEven;

Array.prototype.odd = getOdd;

var a = **[1,2,3,4,5,6,7,8]**;

console.log(a.even());

console.log(a.odd());

**EXERCISE 2.-**

function slow(callback) {

if (Math.random() > 0.5) {

return callback("Error", null)

}

return callback(null, { id: 12345 })

}

function exec(fn) {

let obj = {};

fn(function (err, data) {

obj.done = function (cb) {

if (err === null) {

cb(data);

}

return this;

}

obj.fail = function (cb) {

if (err !== null) {

cb(err);

}

return this;

}

});

return obj;

}

**EXERCISE 3**

**Explain why do we want sometimes to use setImmediate instead of using setTimeout?**

SetImmediate execute after I/O events, so we can be sure all the connections I/O are close. SetTimeOut execute in the timer phase, that means before I/O callbacks

**Explain the difference between process.nextTick and setImmediate?**

Basically process.nextTick execute before setImmediate

**Name 10 global modules/methods available in Node environment.**

|  |  |
| --- | --- |
| SetTimeOut | SetImmediate |
| Process.nextTick | Exports |
| require | Buffer |
| path | loaded |
| filename | id |