CallBacks

So the callback is a function inside another function thanks for this we can manipulate result of function what is already done.
Examples:

```
//First example

function d(a) {
   console.log(a * 2);
}

function a(b, c, d) {
   setTimeout(function () {
       var e = b + c;
       d(e);
       console.log("se ha ejecutado la functiona a");

   }, 1000)

}

a(4, 5, d);
```

In here I create function “a” with parameters b,c,d where parameter ‘d’ is our callback.
I create new variable ‘e’ and I sum to parameters “b” and “c”. Result of variable ‘e’ I give to our callback ‘d’.
Then I create function “d” with parameter ‘a’ what is 9 and i multiply 2 what is result 18.
At the end I just call function “a” with arguments (4,5,d).

Example with objects:

```
var person = [{
        name: "Bugs",
        age: 24,
        weight: 90,
    },
    {
        name: "Tweety",
        age: 22,
        weight: 120,
    },
    {
        name: "Lola",
        age: 21,
        weight: 50,
    },
    {
        name: "Daffy",
        age: 20,
        weight: 60,
    },
    {
        name: "Porky",
        age: 34,
        weight: 200,
    },
]


function iter(person, callback) {
    for (var i = 0; i < person.length; i++) {
        callback(person[i]);

    }
}
// I use callback for display persons who have less or equal 24 years
iter(person, function (cal) {
    if (cal.age <= 24) {
        console.log(cal.name);
    }})
```

In this example I create class person with array objects. Then I create function parameters(person and callback) for iteration our objects and I give
person[i] to our callback.
I call function iter with argument of class and anonymous function with parameter (cal) what is our person[i].
Inside of anonymous function, easy I get persons who have less or equal 24 years.

```
//example for count weight and get number of persons of persons in this
case we take only 5
var ntest = function (callback) {
   var totalAge = 0,
       totalCount = 0;
   iter(person, function (cal) {
       totalAge += cal.age;
       totalCount++;
   })
   callback(totalCount, totalAge);

}
//example for counting medium of the age of 5 people
ntest(function (a, b) {
   console.log(a, "counter");
   console.log(b, "total age");
   console.log(b / a, "average of age");

});
```

In this example I create new function ‘ntest’ with callback parameter inside I create two variables (totalAge,totalCount).Then I call function ‘iter’ and I use

callback for sum age from all persons. I call callback and give arguments(totalCount,totalAge).
Next I call ‘ntest’ with anonymous function what is our callback with parameters ‘a’,’b’ what is (totalCount,totalAge).I display ‘a’,’b’ and ‘b/a’.

```
// array with callback
function putArray(cal) {

   var arr = [];
   iter(person, function (call) {
       if (call.age < 24 && call.weight < 100) {
           arr.push(call.name);

       }
   })
   cal(arr);
}

putArray(function (person) {
   console.log(person);
});
```

In this example I make function ‘putArray’ with callback (cal).
I create variable array, then I call function ‘iter’ with argument ‘person’ and function anonymous what is our callback of iter (person[i]).
I use condition if for get persons with less than 24 years and less than 100kg of weight.
I push persons names to array.
I call cal with argument ‘arr’ what is our array.
I call function ‘putArray’ with our callback for display persons from array.

My opinion :
It’s really good pill, the callbacks is a knowledge what I was missing.Still I can have problems with read more complicated code like callbacks hell, but for sure its gonna be much easier write my own code. Thank you.