Call back JavaScript

Callback Function

A callback function is a function (It can be any function Anonymous Function, Arrow Function) passed into another function as an argument, which is then invoked inside the outer function to complete some kind of routine or action.

```
function show(){
    console.log("I am show Function");
}
function geeky(callback){
    callback();
}
geeky(show);
```

```
function show(a){
    console.log("I am show Function" + a);
}
function geeky(callback){
    var a = 101;
    callback(a);
}
geeky(show);
```

```
function show(a){
    console.log("I am show Function" + a);
}
function geeky(callback){
    var a = 101;
    callback(a);
}
geeky(show);
```

```
function geeky(callback) {
      var a = 101;
      callback(a);
    }
geeky(function(a) {
      console.log("I am show Function " + a);
    });
```

```
function geeky(callback) {
      var a = 101;
      callback(a);
    }
geeky(function show(a) {
      console.log("I am show Function " + a);
    });
```

```
function geeky(callback) {
      var a = 101;
      callback(a);
    }
geeky(a => console.log("I am show Function " + a));
```

```
function show(){
    console.log("I am show Function");
}
function geeky(callback){
    callback();
}

geeky(show);

console.log("End");
```

Synchronous - It waits for each operation to complete, after that it executes the next operation.

```
setTimeout(function show(){
    console.log("I am show Function");
}, 5000);

console.log("End");
```

Asynchronous - It never waits for each operation to complete, rather it executes all operations in the first GO only.

What problem callback trying to solve?

```
let calc = function(num1, num2, calcType){

if (calcType === "add"){
  return num1+ num2;
} else if (calcType === "multiply"){
  return num1* num2;
}

};

console.log(calc(2,3, 'add'));
```

```
let add = function(a,b){
  return a+b;
};

let multiply = function(a,b){
  return a*b;
};

let calc = function(num1, num2, callback){
  callback(num1, num2);
};

console.log(calc(2,3, add));
```

```
let multiply = function(a, b) {
  return a * b;
};
let doWhatever = function(a, b) {
  console.log(`here are your two numbers bakck ${a} , ${b}`);
};

let calc = function(num1, num2, callback) {
  if (typeof callback === "function"){
  return callback(num1, num2);
  }

};
```

```
var myArr = [{
  num: 5,
  str: 'apple'
}, {
  num: 7,
  str: 'cabbage'
}, {
  num: 1,
  str: 'ban'
}];

myArr.sort(function(val1, val2) {
  if (val1.str > val2.str) {
    return -1;
  } else {
    return 1;
  }
});

console.log(myArr);
```

```
var floppy = require('floppy');

floppy.load('disk', function(data) {
    floppy.load('disk', function(data) {
        floppy.load('disk', function(data) {
            floppy.load('disk', function(data) {
                floppy.load('disk', function(data) {
                    floppy.load('disk', function(data) {
                        floppy.load('disk', function(data) {
                            floppy.load('disk', function(data) {
                            });
                        });
                    });
                });
            });
        });
    });
});
```


PROMISE

```
let promiseToCleanTheRoom = new Promise(function(resolve, reject) {

  //cleaning the room

  let isClean = false;

  if (isClean) {
    resolve('Clean');
  } else {
    reject('not Clean');
  }

});

promiseToCleanTheRoom.then(function(fromResolve) {
  console.log('the room is' + fromResolve);
}).catch(function(fromReject){
  console.log('the room is' + fromReject);
})
```

```
let cleanRoom = function() {
  return new Promise(function(resolve, reject) {
    resolve('Cleaned The Room');
  });
};

let removeGarbage = function(p) {
  return new Promise(function(resolve, reject) {
    resolve('remove Garbage');
  });
};

let winIcecream = function(p) {
  return new Promise(function(resolve, reject) {
    resolve('won Icecream');
  });
};
```

JAVASCRIPT

```javascript
let cleanRoom = function() {
  return new Promise(function(resolve, reject) {
    resolve('Cleaned The Room');
  });
};

let removeGarbage = function(p) {
  return new Promise(function(resolve, reject) {
    resolve('remove Garbage');
  });
};

let winIcecream = function(p) {
  return new Promise(function(resolve, reject) {
    resolve('won Icecream');
  });
};

cleanRoom().then(function(result){
  return removeGarbage(result);
}).then(function(result){
  return winIcecream(result);
}).then(function(result){
  console.log('finished ');
})
```