

A8. Comando ls versión modular

Creamos el módulo

```
const path = require('path');
const fs = require('fs');
const r = process.argv[2];

//COMO ARGUMENTO r es la ruta , e es la extension, y callback

module.exports = (r,e,callback)=> {

  fs.readdir(r,function(err,lista){

    if(err)

    return callback(err,null);

    for(var archivo in lista){
      if(path.extname(lista[archivo])=="."+e){
        console.log(lista[archivo]);
        // console.log(archivo);
      }
      callback(null,lista);
    }
  })
}
```

Creamos nuestro archivo principal

```
1  var modulo = require('./modulo.js');
2  const r = process.argv[2];
3  function ls(r,e){
4
5    console.log("Buscando archivos en " + r + " Con la extension "+ e);
6    setTimeout(() =>
7      modulo(r,e, (err,modulo) =>{
8
9        if(err){
10          console.log("error: " , "Ha ocurrido un problema");
11        }
12      })),2000);
13
14  };
15
16  ls('/home/ivan/node/A8','txt')
```

Probamos el programa, y vemos como nos devuelve los archivos txt

```
root@debian-ivan:/home/ivan/node/A8# node ls.js
Buscando archivos en /home/ivan/node/A8 Con la extension txt
hola.txt
saliendo.txt
root@debian-ivan:/home/ivan/node/A8# _
```

Ahora vamos a poner un error en el código para ver si nos salta el mensaje de error

```
GNU nano 2.7.4 fichero: ls.js
var modulo = require('./modulo.js');
const r = process.argv[2];
function ls(r,e){
    console.log("Buscando archivos en " + r + " Con la extension "+ e);
    setTimeout(() =>
    modulo(e,e, (err,module) =>{
        if(err){
            console.log("error: " , "Ha ocurrido un problema");
        }
    },2000);
};
ls('/home/ivan/node/A8','txt')
```

Vemos como salta un error

```
root@debian-ivan:/home/ivan/node/A8# node ls.js
Buscando archivos en /home/ivan/node/A8 Con la extension txt
error:  Ha ocurrido un problema
root@debian-ivan:/home/ivan/node/A8#
```