console.log('files reading ');

var fs = require('fs');

fs.readFile('package.json','utf8', function (err, contents) {

if (!err) console.log(contents);

});

console.log('files writting ');

var new\_input = 'this is writting by node programs';

fs.writeFile('writing.txt',new\_input,function (err){

if (err){

return console.log(err);

}

console.log('files has been altered');

});

console.log('buffers');

var mybuffer = Buffer.from('==ii1j2i3h1i23h', 'base64');

console.log(mybuffer);

require('fs').writeFile('logo.jpg', mybuffer,function(err){

console.log(err);

});

console.log('events');

var EventEmitter = require('events').EventEmitter;

var action = new EventEmitter;

action .on('event', function () {

console.log('Action event has been fired');

});

action.emit('event');

// simplifies Chapter 5 code files looking through

var fs = require('fs'),stdin = process.stdin,stdout = process.stdout;

fs.readdir(process.cwd(), function (err, files) {

console.log('');

if (!files.length) {

return console.log(' \033[31m No files to show!\033[39m\n');

}

console.log(' Select which file or directory you want to see\n');

//var stats = [];

function file(i) {

var filename = files[i];

fs.stat(\_\_dirname + '/' + filename, function (err, stat) {

if (stat.isDirectory()) {

console.log(' '+i+ ' \033[36m' + filename + '/\033[39m ');

} else {

console.log(' '+i+ '\033[90m' + filename + '\033[39m');

}

if(++i == files.length){

read();

}else{

file(i);

}

});

}

function read(){

console.log('');

stdout.write( 'Enter your choice');

stdin.resume();

stdin.setEncoding('utf8');

stdin.on('data',option);

}

function option(data){

var filename = files[Number(data)];

if (!filename) {

stdout.write(' enter your choice:');

}else{

stdin.pause();

fs.readFile(\_\_dirname + '/' + filename, 'utf8', function(err, data){

//stats[i] = stat;

console.log('');

console.log('\033[90m'+ data.replace(/ ( .\*)/g,' $1')+'\033[39m');

});

}

}

file(0);

});