**Lesson 1**

Write a program that prints the text "HELLO WORLD" to the console (stdout)

**Solution 1**

console.log("HELLO WORLD");

**Lesson 2**

Write a program that accepts one or more numbers as command-line arguments and prints the sum of those numbers to the console (stdout).

**Solution 2**

var data = process.argv

var info = data.slice(2);

function sum(array){

var result = array.reduce(function(a,b){

a = parseInt(a);

b = parseInt(b);

return a+b;

},0);

console.log(result);

};

sum(info);

**Lesson 3 -** MY FIRST I/O

Write a program that uses a single synchronous filesystem operation to

read a file and print the number of newlines (\n) it contains to the

console (stdout), similar to running cat file | wc -l.

The full path to the file to read will be provided as the first

command-line argument (i.e., process.argv[2]). You do not need to make

your own test file.

**Notes:** All synchronous (or blocking) filesystem methods in the fs module end with 'Sync'.

**Solution 3**

var linkto = process.argv[2]; // get path for the file

var fs = require("fs");

var data = fs.readFileSync(linkto); // read file

var info = data.toString(); // turn file into string

var count = info.split("\n").length-1; // split the string by \n, and the count the number of iten in the array.

console.log(count);

**Lesson 4 -** MY FIRST ASYNC I/O! (Exercise 4 of 13)

Write a program that uses a single asynchronous filesystem operation to read a file and print the number of newlines it contains to the console (stdout), similar to running cat file | wc -l.

The full path to the file to read will be provided as the first command-line argument.

**Solution 4**

var linkto = process.argv[2]; // get path for the file

var fs = require("fs");

fs.readFile(linkto, function(err,data){

if(err){

console.log(err);

};

var info = data.toString(); // turn file into string

var count = info.split("\n").length-1; // split the string by \n, and the count the number of iten in the array.

console.log(count);

});

**Lesson 5 –**  ## FILTERED LS

Create a program that prints a list of files in a given directory, filtered by the extension of the files. You will be provided a directory name as the first argument to your program (e.g. '/path/to/dir/') and a file extension to filter by as the second argument.

For example, if you get 'txt' as the second argument then you will need to filter the list to only files that end with .txt. Note that the second argument will not come prefixed with a '.'.

Keep in mind that the first arguments of your program are not the first values of the process.argv array, as the first two values are reserved for system info by Node.

The list of files should be printed to the console, one file per line. You must use asynchronous I/O.

**Solution 5**

var fs = require("fs");

var linkto = process.argv[2]; // get path for the file

var ending = "."+process.argv[3] // get the ending extension to match with

var endinglength=ending.length; //determine the length of ending

fs.readdir(linkto, function(err,data){

if (err){

console.log(err);

};

var filtered = data.filter(function(word){

if(word.substr(-endinglength)==ending){ // test if ending of the word matches with the ending we want

console.log(word)

return

};

});

});