003 npm - multiple tasks & vs

NPM Kata List – Blog Page

[NPM Kata List on GitHub](https://github.com/robertdunaway/katas-gulp)

# Duration

10 minutes

# Brief

Execute multiple tasks with NPM scripts.

### For More Information

BING/GOOGLE: “NPM multiple tasks visual studio”

# Instructions

Get tutorial folder or the entire katas-npm repo.

Open the [before/\*.sln] file and execute the kata.

Feel free to execute this kata multiple times because repetition creates motor memory.

# Github

* Before (start kata with this)
  + https://github.com/robertdunaway/katas-npm/tree/master/003%20npm%20-%20multiple%20tasks%20%26%20vs/before
* After
  + https://github.com/robertdunaway/katas-npm/tree/master/003%20npm%20-%20multiple%20tasks%20%26%20vs/after

# Kata

In this kata create and run multiple tasks.

Create three new tasks: task1.js, task2.js, and task3.js

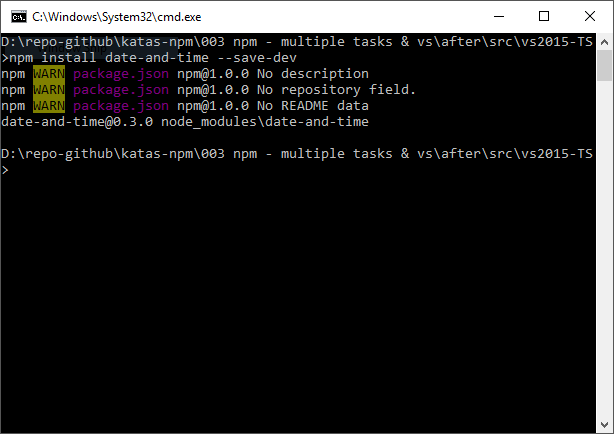
Execute these three tasks.

# Review

Here is one way you could have accomplished this task.

To show when a task is executed we will use an NPM package called date-and-time.

npm install date-and-time --save-dev



Create a task named timer.js to implement the time stamp portion of this example.

var date = require('date-and-time');

// function works for 2 seconds at the start and at the end it shows times when it stops

module.exports = function (name) {

var time = function () {

return date.format(new Date(), 'H:mm:ss');

}

console.log(name + ' started at ' + time());

};

Create three more files with the following code.

task1.js

var timer = require('./timer.js');

timer("task1");

task2.js

var timer = require('./timer.js');

timer("task2");

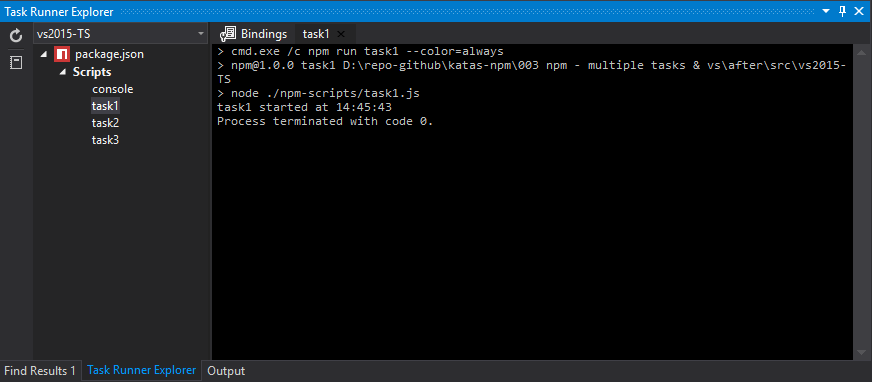
task3.js

var timer = require('./timer.js');

timer("task3");

Add task1, task2, and task3 to the package.json file.

Right click one of the tasks in the Task Runner Explorer and select Run.



At this point you can execute each task individually but our goal is to execute multiple tasks.

Add a new task to the package.json file named allTasks and execute all three tasks.

{

"name": "npm",

"version": "1.0.0",

"description": "",

"main": "index.js",

"scripts": {

"console": "node ./npm-scripts/console.js",

"task1": "node ./npm-scripts/task1.js",

"task2": "node ./npm-scripts/task2.js",

"task3": "node ./npm-scripts/task3.js",

"allTasks": "node ./npm-scripts/task1.js && node ./npm-scripts/task2.js && node ./npm-scripts/task3.js"

},

"author": "",

"license": "ISC",

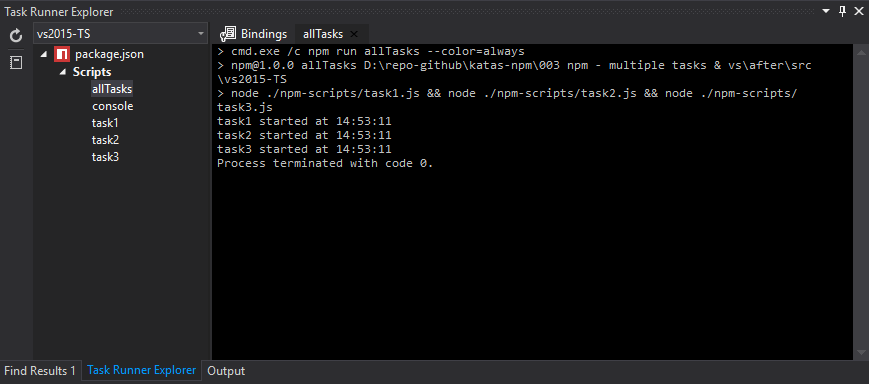
"devDependencies": {

"date-and-time": "^0.3.0"

}

}

Adding the “&&” symbol between each task causes the task to run in sequence.



# Next

Take a few minutes and imagine more examples.