**Package Scripts**

The **"scripts"** field in **package.json** can be used to define aliases for shell commands that are relevant to a Node.js project.

To demonstrate the concept, let's add a lint script. Currently the **package.json** **"scripts"** field looks like so:

**"scripts": {  
    "test": "echo \"Error: no test specified\" && exit 1"  
 },**

Let's update it to the following:

**"scripts": {  
    "test": "echo \"Error: no test specified\" && exit 1",  
    "lint": "standard"  
  },**

Recall that we have a development dependency installed called **standard**. This is a code linter, see [*"JavaScript Standard Style"*](https://standardjs.com/) article for more details.

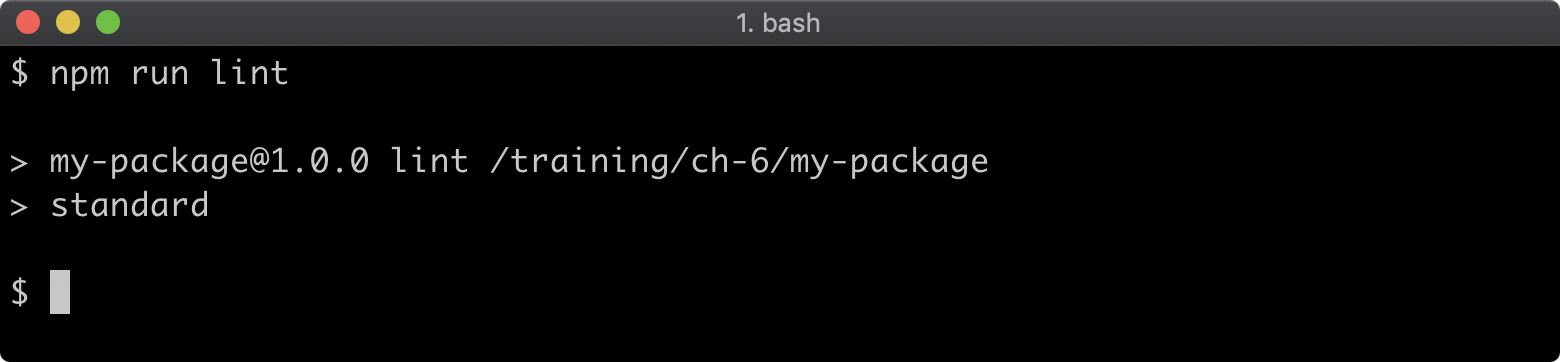
Packages can assign a **"bin"** field in their **package.json**, which will associate a namespace with a Node program script within that package. In the case of **standard**, it associates a command named **standard** with a Node program script that performs linting. The associated commands of all installed packages are available within any defined **package.json** scripts.

Let's make sure all dependencies are installed before we try out the *"lint"* script by running.

**npm install**

Next, to execute the script use **npm run**:

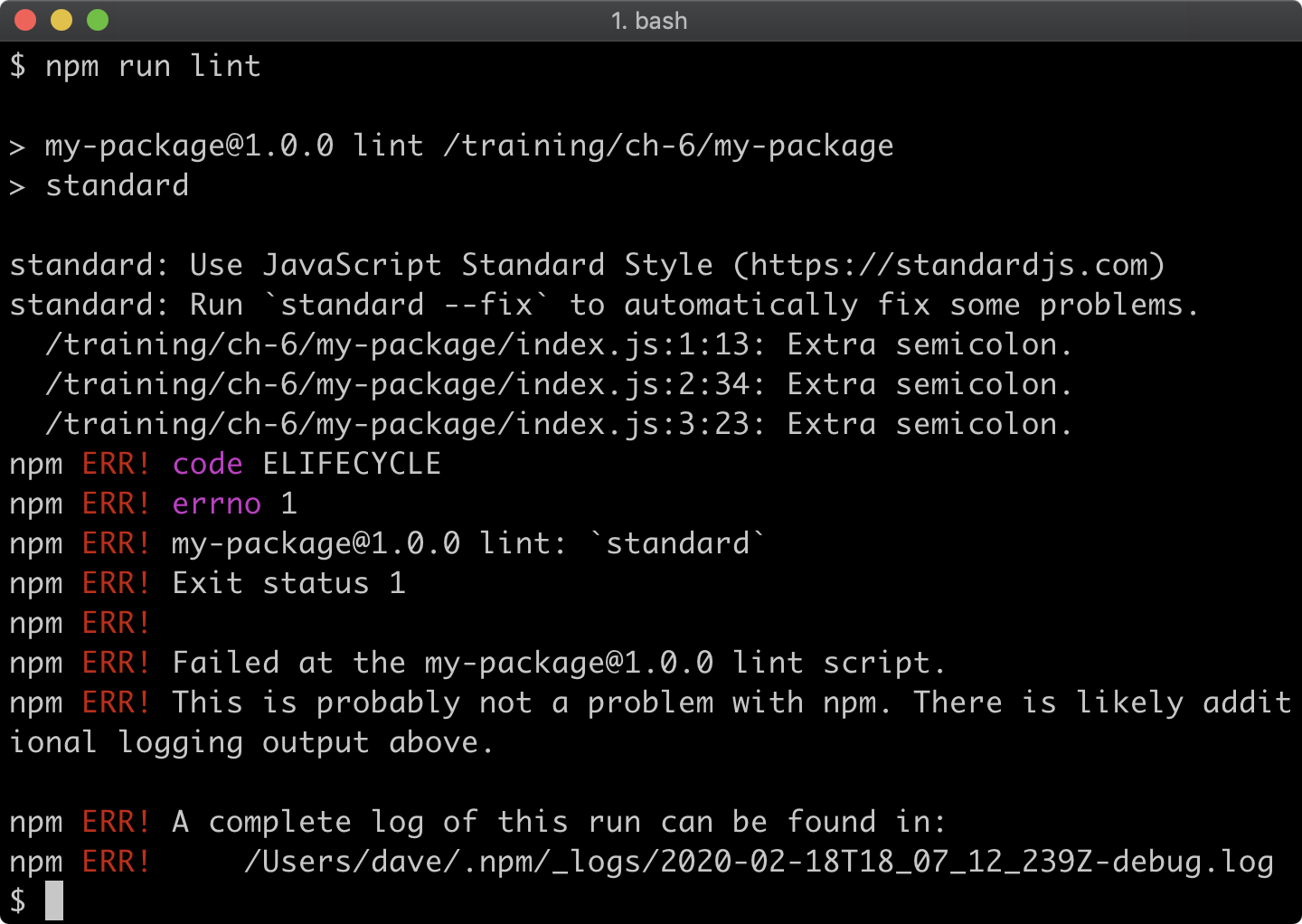
**npm run lint**



There will be no output because there are no files to lint, let's add a file to **my-package** called **index.js** with the following contents:

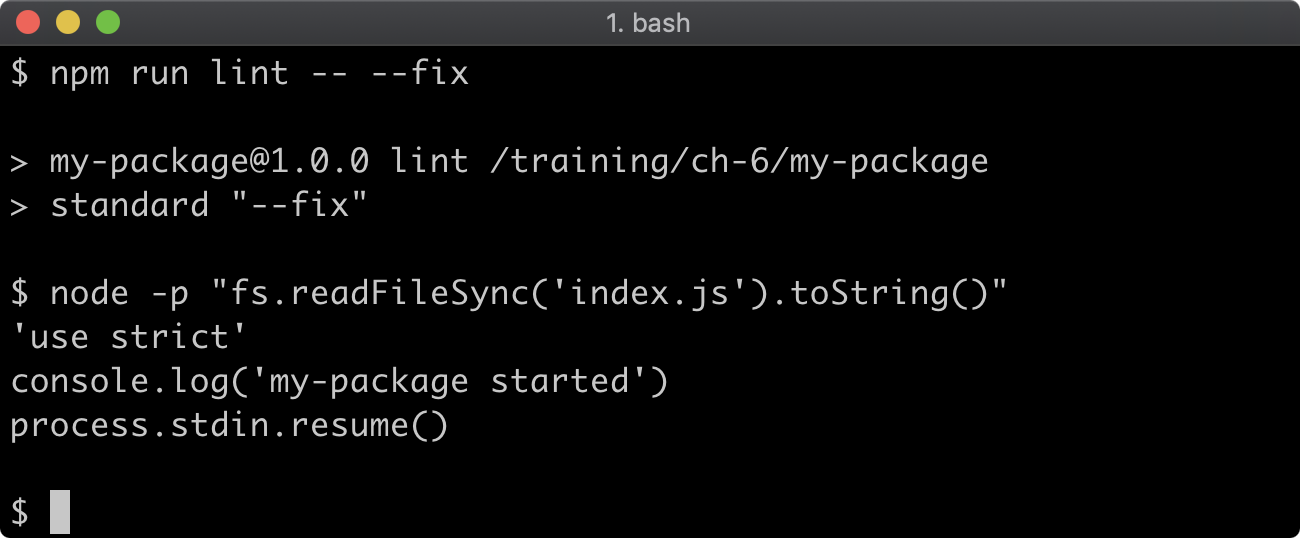
**'use strict';  
console.log('my-package started');  
process.stdin.resume();**

Now let's execute **npm run lint** again:



We have some lint errors. The **standard** linter has a **--fix** flag that we can use to autocorrect the lint errors. We can use a double dash (**--**) to pass flags via **npm run** to the aliased command:

**npm run lint -- --fix**



*Cont'd on the next page.*