1.1 TypeScript Configuration with Grunt

This section will guide you to:

* Configure TypeScript with Grunt

This guide has three subsections, namely:

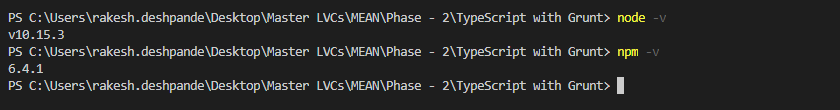
1.1.1 Setting up Grunt environment with TypeScript

1.1.2 Generating a .js file with server.ts and Gruntfile.js

1.1.3 Pushing the code to GitHub repositories

**Step 1.1.1:** Setting up Grunt environment with TypeScript

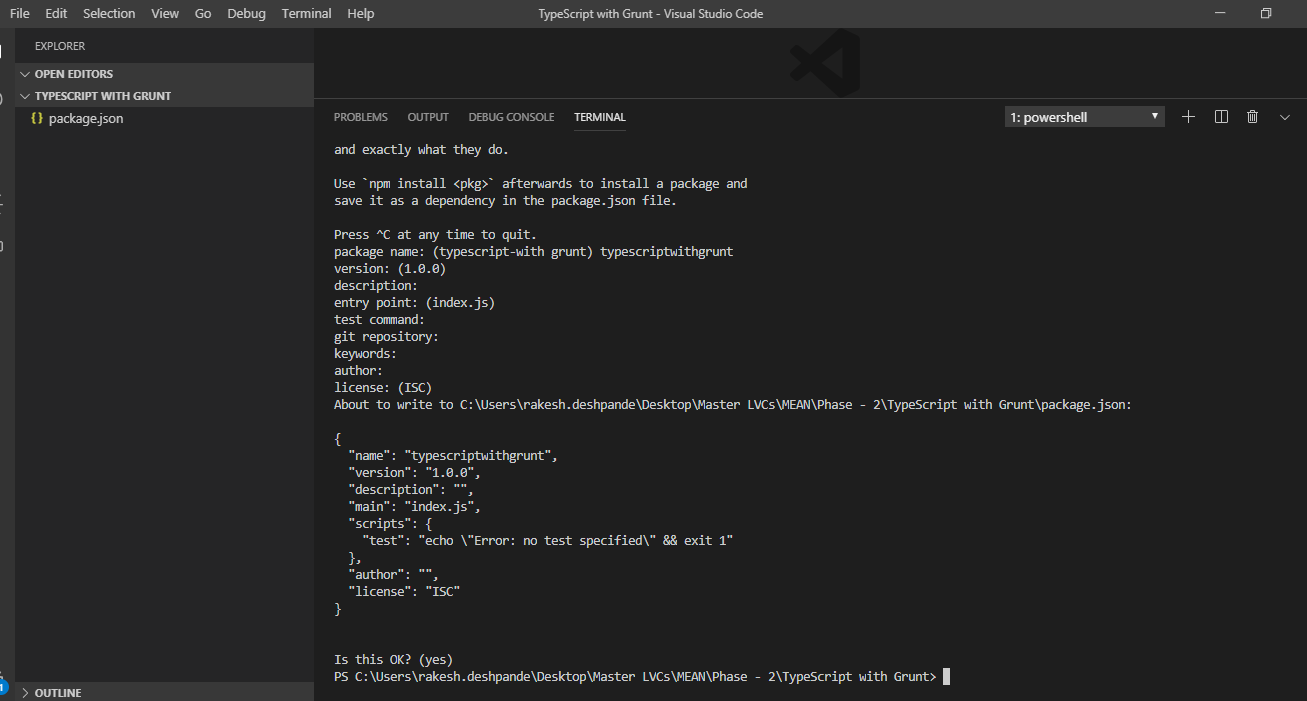
* Create a folder **TypeScript with Grunt** in your system (/lab).
* Open this folder in Visual Studio Code
* Node.js is already installed in your lab. (Refer MEAN: Lab Guide - Phase 2)
* Open the terminal navigating to **Terminal** > **New** **Terminal**
* You should be able to check the *npm* and *node* version in your terminal as shown in the screenshot (The versions may differ from the ones available in the screenshot)



* Run the following command to set up the package.json file

npm init

* Enter package name as **typescriptwithgrunt** and press [enter]
* Enter version number, description, entry point, test command, git repository, keywords, author, and license
* Check if the configuration you suggested is correct
* Press [enter] again. It should create a *package.json*file in the directory



* Run the following command to set up Grunt and TypeScript in your project:

*sudo npm install typescript --save-dev*

*sudo npm install grunt --save-dev*

*sudo npm install grunt-cli -g*

* Set up the grunt-ts by running the following command:

*sudo npm install grunt-ts*

**Step 1.1.2:** Generating a .js file with server.ts and Gruntfile.js

* Open Visual Studio Code
* Create two files, **server.ts** and **Gruntfile.js**
* Add the following source code in server.ts:

var sys = require("sys"),

my\_http = require("http");

my\_http.createServer(function(request,response){

sys.puts("I got kicked");

response.writeHeader(200, {"Content-Type": "text/plain"});

response.write("Hello World");

response.end();

}).listen(8080);

sys.puts("Server Running on 8080");

* Add the following source code in Gruntfile.js:

module.exports = function(grunt) {

grunt.initConfig({

ts: {

default : {

src: ["\*\*/\*.ts", "!node\_modules/\*\*/\*.ts"]

}

}

});

grunt.loadNpmTasks("grunt-ts");

grunt.registerTask("default", ["ts"]);

}

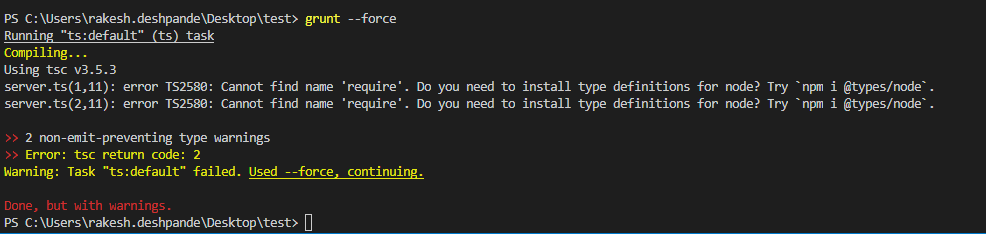
* Run the following command to execute Grunt on the .ts file:

*grunt --force*

* You should be able to find that the following files will be created with a few errors and warning at the command line:

*server.js*

*server.js.map*



**Step 1.1.3:** Pushing the codes to your GitHub repositories

Open your command prompt and navigate to the folder where you have created your files.

**cd <folder path>**

Initialize your repository using the following command:

**git init**

Add all the files to your git repository using the following command:

**git add .**

Commit the changes using the following command:

**git commit . -m “Changes have been committed.”**

Push the files to the folder you initially created using the following command:

**git push -u origin master**