3.4 Deployment Using Grunt Plugins



This section will guide you to:

* Prepare your projects for deployment using Grunt Plugins

This lab has four subsections, namely:

3.4.1 Creating deployment packages

3.4.2 Integrating Git

3.4.3 Moving files to production

3.4.4 Pushing the code to your GitHub repositories

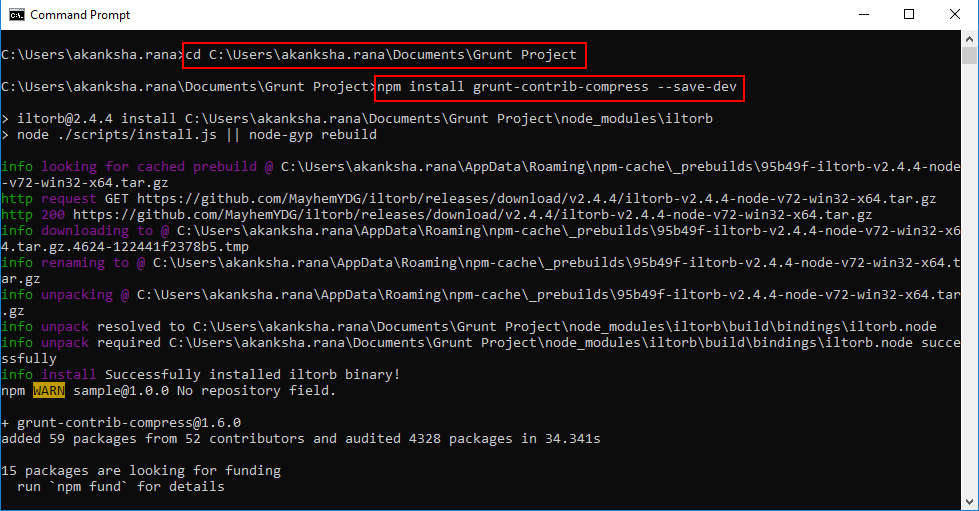
**Step 3.4.1:** Creating deployment packages

We will create a production-ready zip package using grunt-contrib-compress plugin.

* Install the grunt compress plugin using the following commands in the command-line interface:

**cd <path of your root project folder>**

**npm install grunt-contrib-compress --save-dev**

****

* Copy and paste the following code snippet in the gruntfile.js file as shown in the screenshot below:

**compress: {**

**main: {**

**options: {**

**archive: 'site.zip'**

**},**

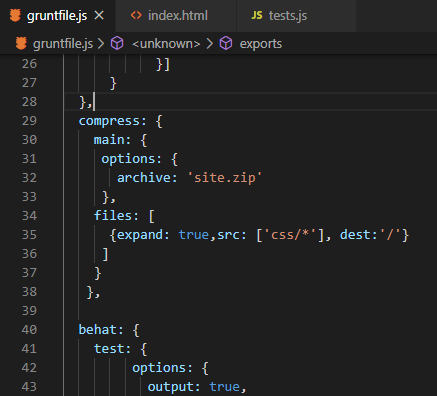
**files: [**

**{expand: true,src: ['css/\*'], dest:'/'}**

**]**

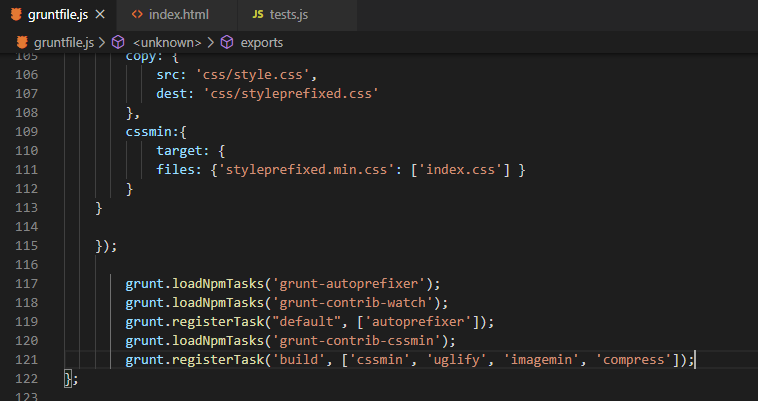
**}**

**},**



* Copy and paste the following code snippet in the gruntfile.js file as shown in the screenshot:

**grunt.registerTask('build', ['cssmin', 'uglify', 'imagemin', 'compress']);**

****

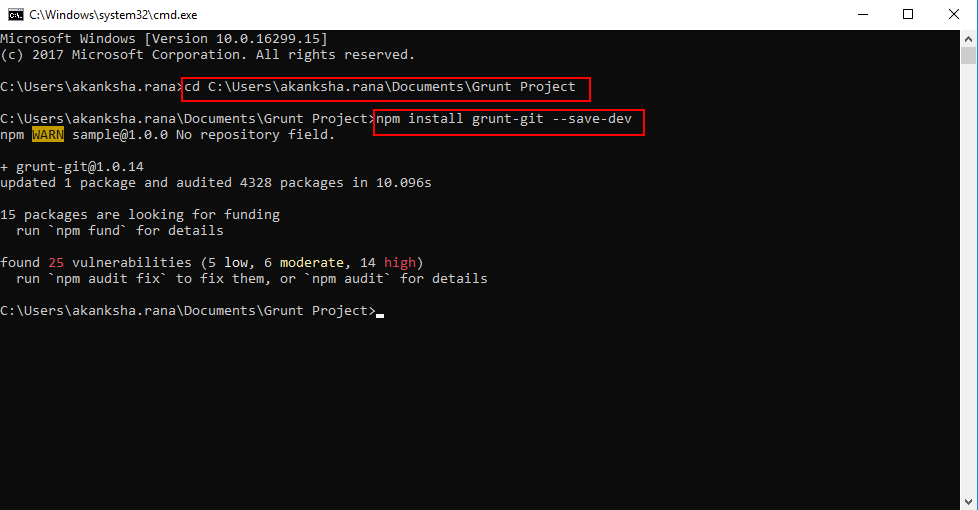
* Click on **Teminal** option on the top menu and click on **Run Tasks**. Select **build.**

**Step 3.4.2:** Integrating Git

* Install the git plugin using the following commands in the command-line interface:

**cd <path of the root project directory>**

**npm install grunt-git --save-dev**

****

* Copy and paste the following code snippet in the gruntfile.js file as shown in the screenshot below:

**gitcommit: {**

**task: {**

**options: {**

**message: 'Testing',**

**noVerify: true,**

**noStatus: false**

**},**

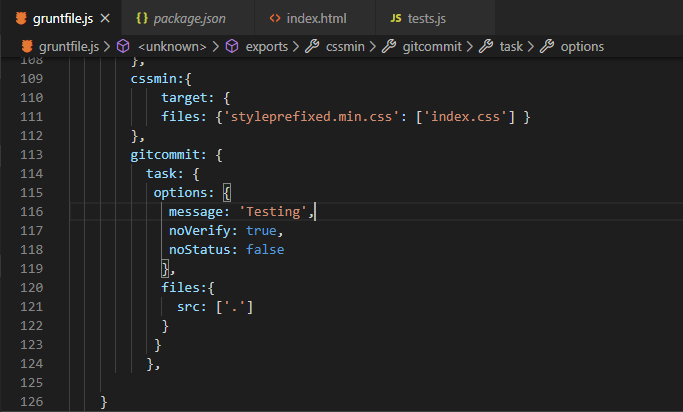
**files:{**

**src: ['.']**

**}**

**}**

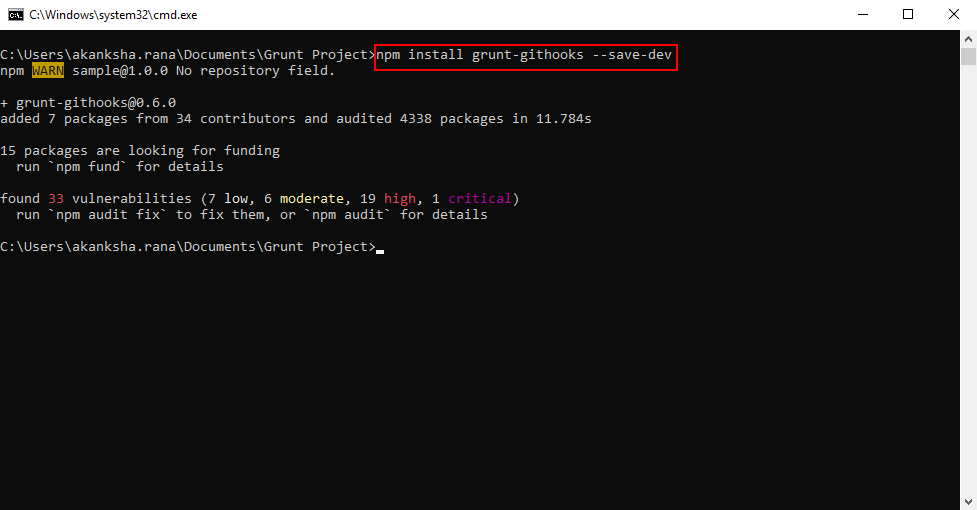
**},**



* Install githooks plugin using the following commands in the command-line interface:

**cd <path of the root project directory>**

**npm install grunt-githooks --save-dev**

****

* Copy and paste the following code snippet in the gruntfile.js file as shown in the screenshot below:

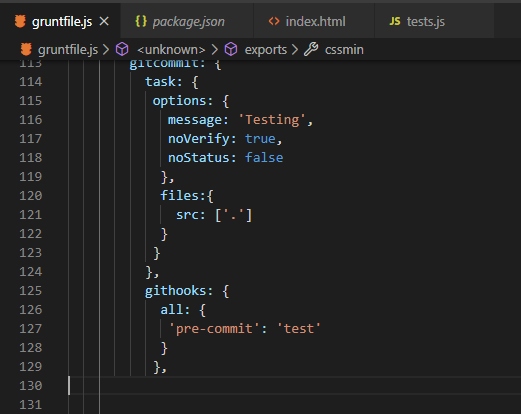
**githooks: {**

**all: {**

**'pre-commit': 'test'**

**}**

**},**



* Click on **Teminal** option on the top menu and click on **Run Tasks**. Select **githooks.**

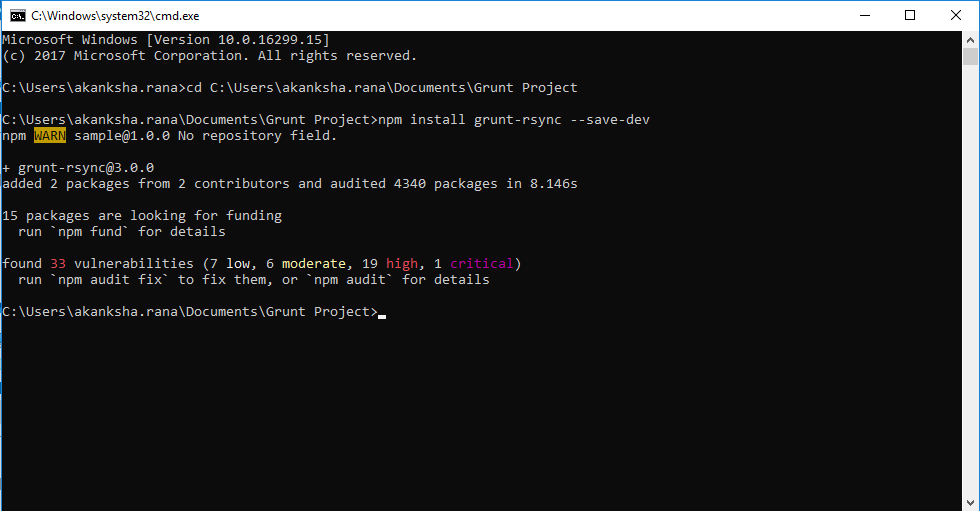
**Step 3.4.3:** Moving files to production

We will use rsync plugin to move files to a production server.

* Install the rsync plugin using the following commands in the command-line interface:

**cd <path of the root project folder>**

**npm install grunt-rsync --save-dev**

****

* Copy and paste the following code snippet in the gruntfie.js file as shown in the screenshot below:

**rsync: {**

**options: {**

**exclude: [".git\*", "\*.scss", "node\_mpdules"],**

**reclusive: true**

**},**

**prod: {**

**options: {**

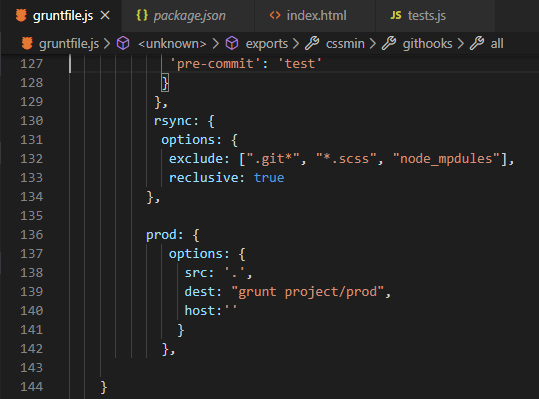
**src: ‘.’,**

**dest: "grunt project/prod",**

**host:’’**

**}**

**},**



**Note:** In the host field, you have to provide the hostname of the server where you want to move the files, whereas, in the destination field, you have to provide the directory structure where the files will be saved.

* Click on **Teminal** option on the top menu and click on **Run Tasks**. Select **prod.**

**Step 3.4.4:** Pushing the code 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 created initially using the following command:

**git push -u origin master**