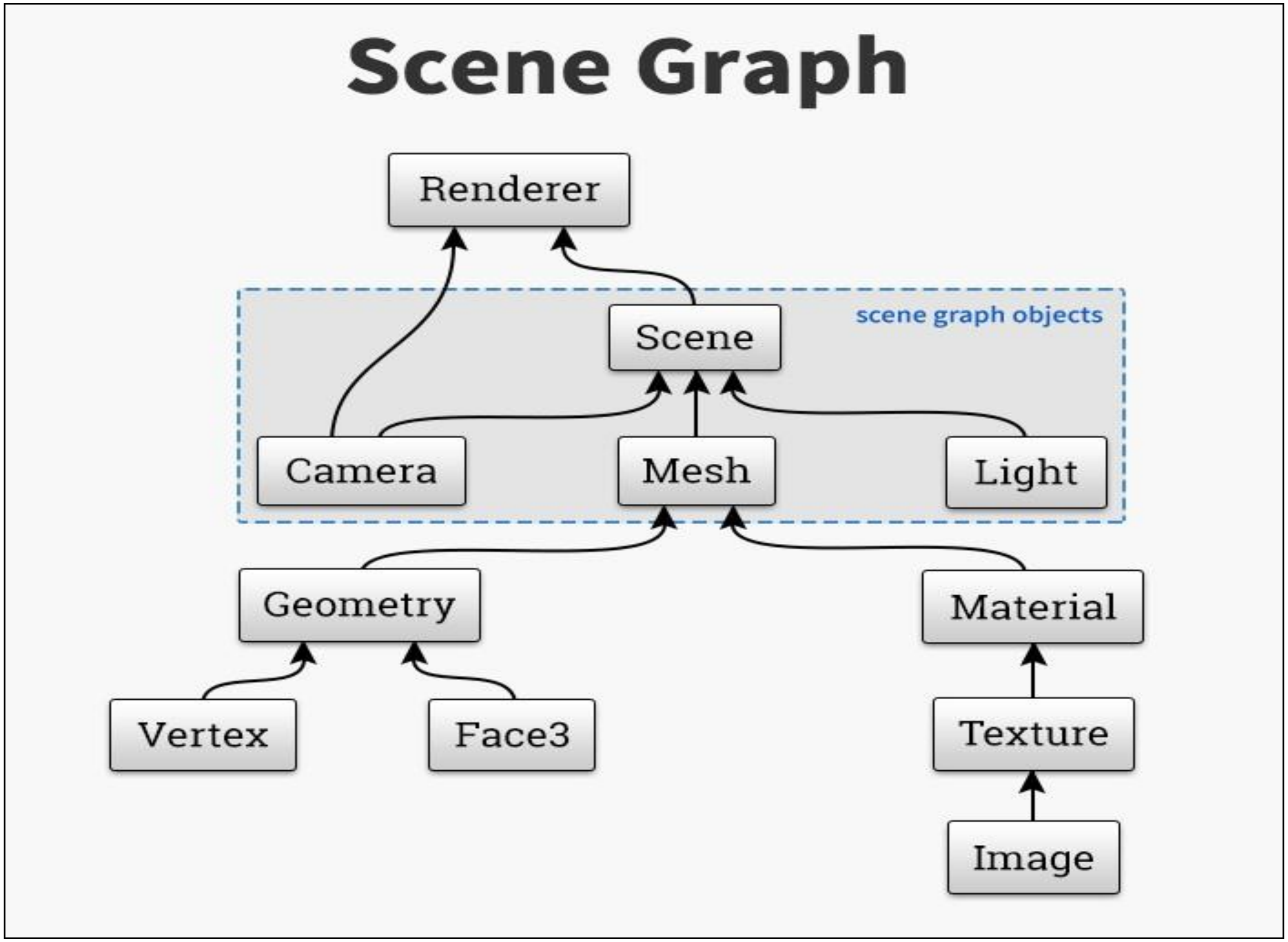
Lab 07- Lab 08
Set up node.js project to use Three.js library as module

Discussion is based on F.S. Hill Chapter 05,06

BSCS – 514 Computer Graphics
Course Supervisor Dr. Humera Tariq

Tree Structure followed by three.js and other graphics APIS



High level Summary

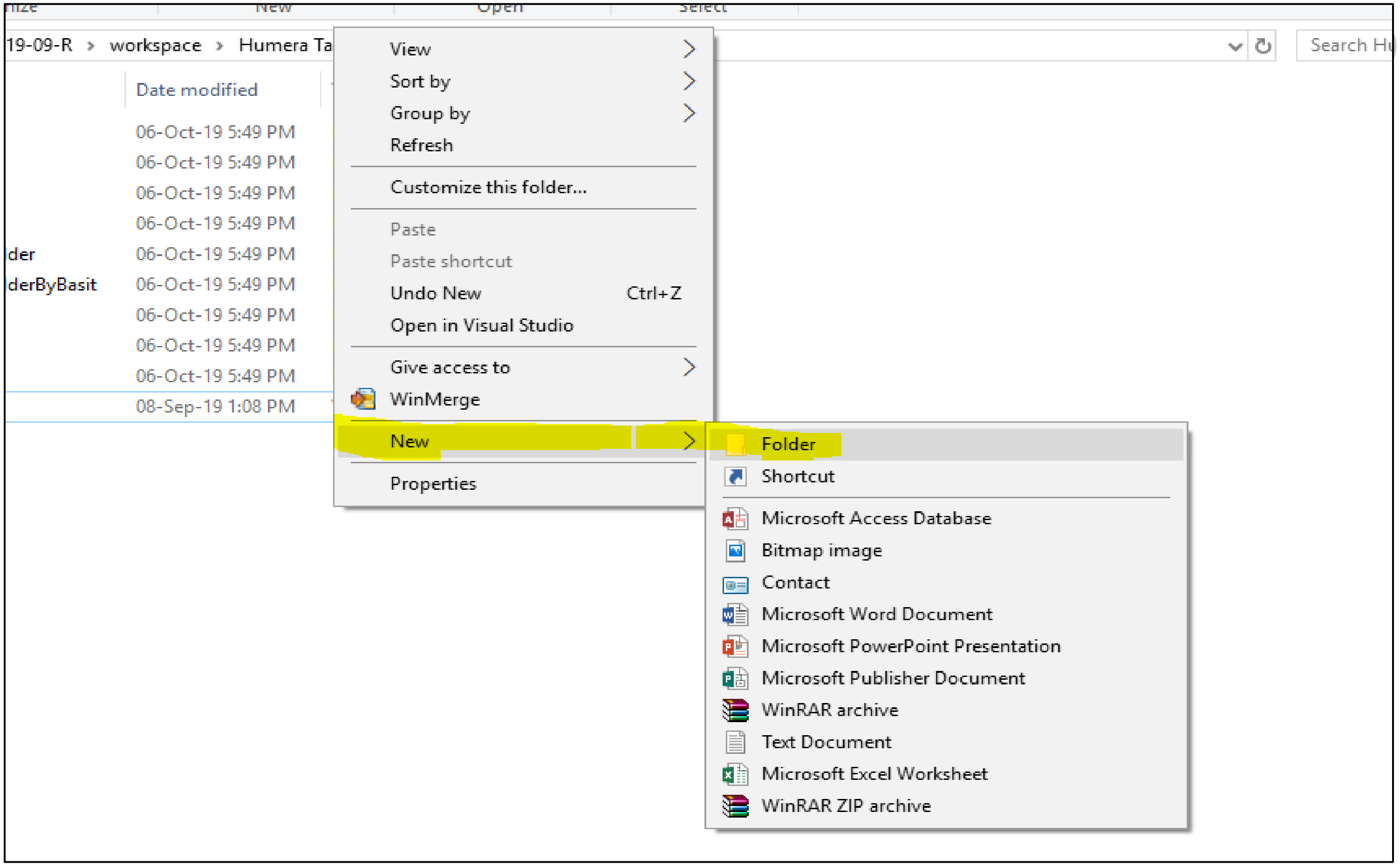
1) npm init
2) npm install –-save three
3) npm install –-save webpack
4) npm run build
5) Npm run build-dev-watch
6) Files to be used
   > package.json
   > webpack.config.js
   > index.html
   > index.js

Step 01:
- Create package.json File

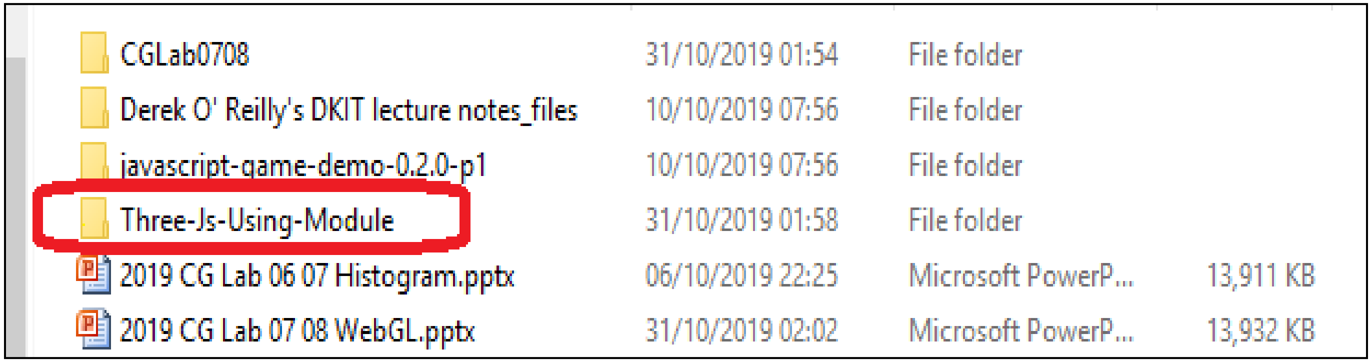
Project setup

Create a folder by right clicking on mouse.
Select New → Folder

1



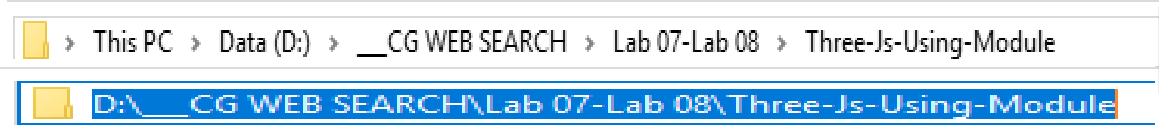
Give folder name of your choice. Make sure there will be no space in folder name. Otherwise when using npm init command. It will complain.



CGLab0708	31/10/2019 01:54	File folder	
Derek O' Reilly's DKIT lecture notes_files	10/10/2019 07:56	File folder	
javascript-game-demo-0.2.0-p1	10/10/2019 07:56	File folder	
Three-Js-Using-Module	31/10/2019 01:58	File folder	
2019 CG Lab 06 07 Histogram.pptx	06/10/2019 22:25	Microsoft PowerP...	13,911 KB
2019 CG Lab 07 08 WebGL.pptx	31/10/2019 02:02	Microsoft PowerP...	13,932 KB

2

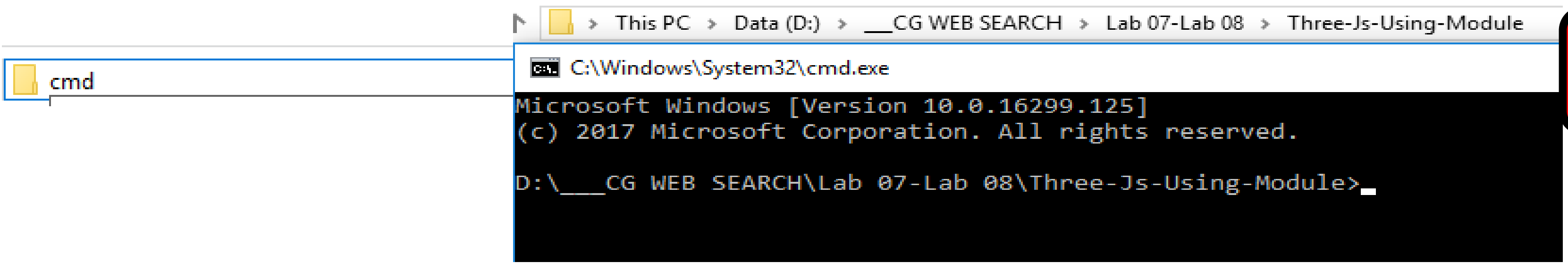
Now go to the create directory and open command prompt by typing cmd in search bar in the same folder



This PC › Data (D:) › __CG WEB SEARCH › Lab 07-Lab 08 › Three-Js-Using-Module

D:\__CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module

3



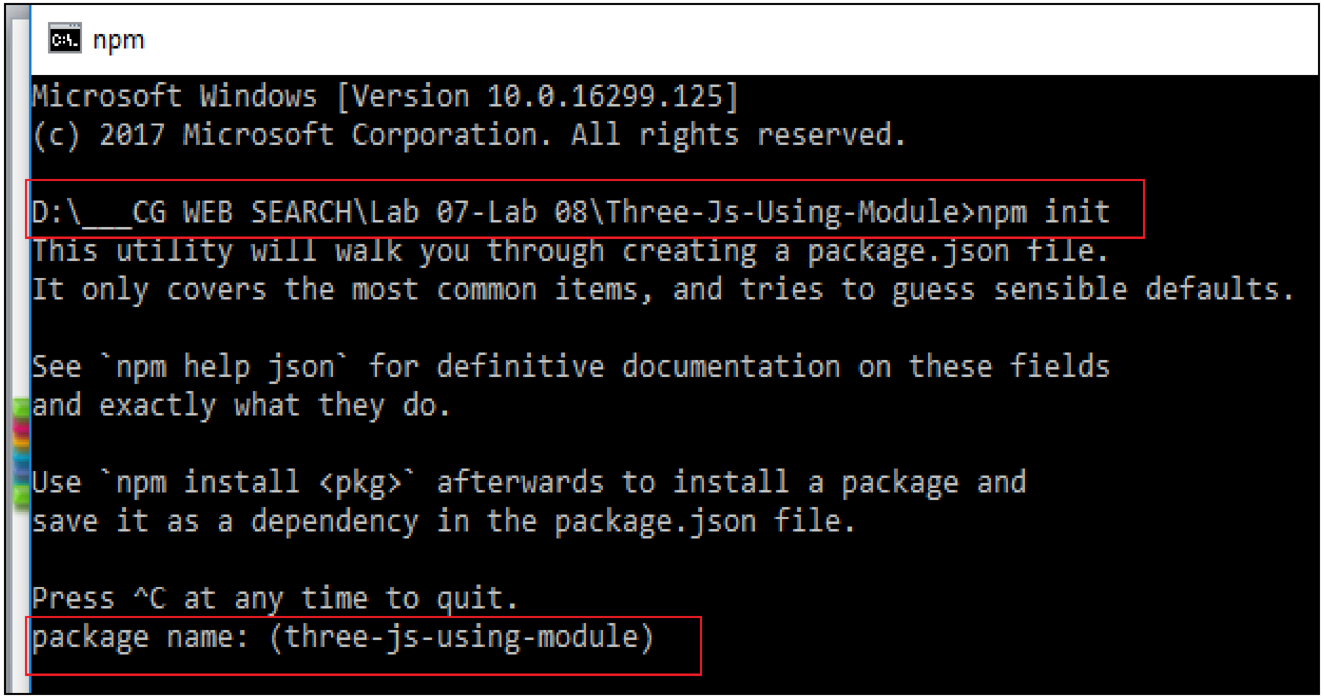
This PC › Data (D:) › __CG WEB SEARCH › Lab 07-Lab 08 › Three-Js-Using-Module

cmd

C:\Windows\System32\cmd.exe

```
Microsoft Windows [Version 10.0.16299.125]
(c) 2017 Microsoft Corporation. All rights reserved.

D:\___CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module>_
```

4



```
Microsoft Windows [Version 10.0.16299.125]
(c) 2017 Microsoft Corporation. All rights reserved.

D:\___CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module>npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help json` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (three-js-using-module)
```

5

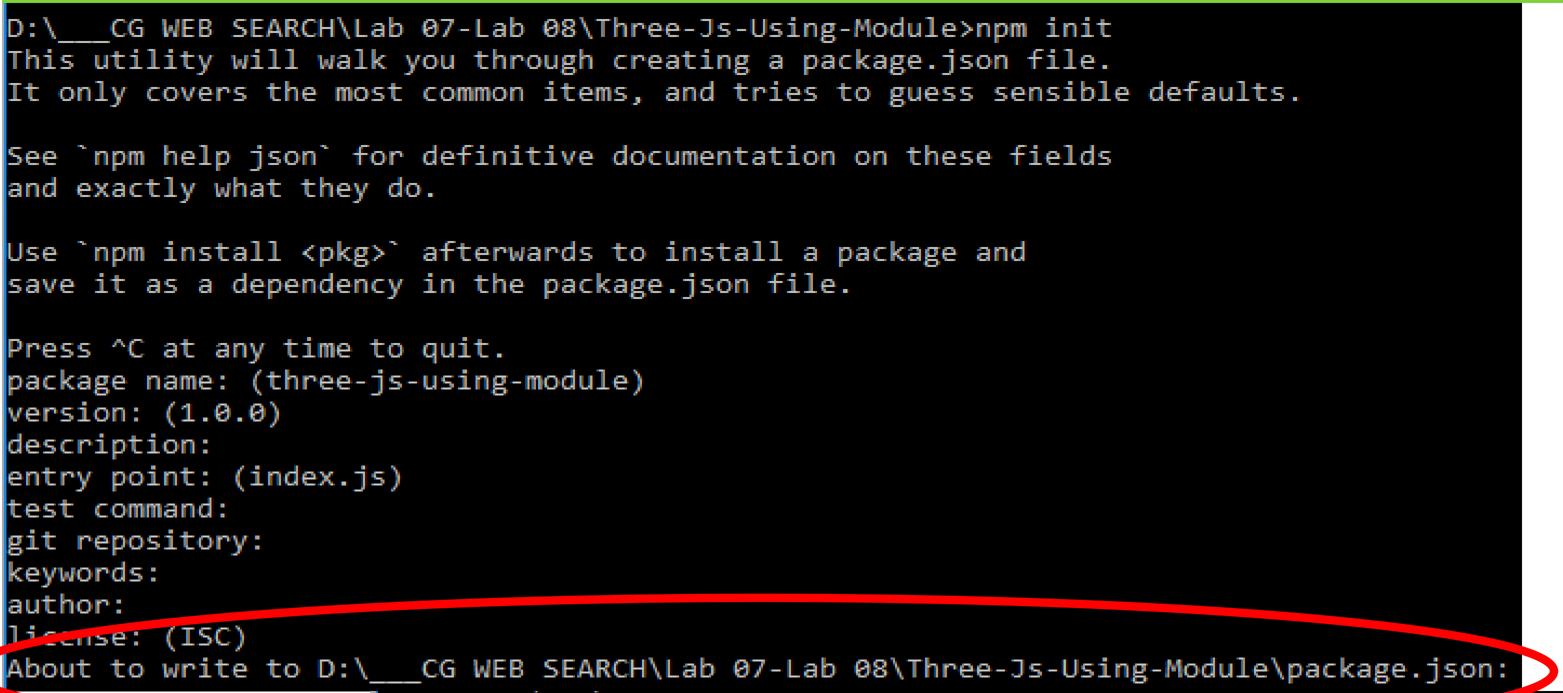
Now type the command npm init. This will kick off a series of questions that will help set up Node.js on our project. The first question will ask you to specify your project name. Hitting Enter will allow you to specify the default value that has already been selected for you. That is all great, but the default name is our project folder, which is Three-JS-Using-Module. If you hit Enter, because it contains capital letters, it will throw an error.

Either accept default by pressing enter or go ahead and enter the lowercase version of the name, three-js-using-module. Once you've done that, press Enter.

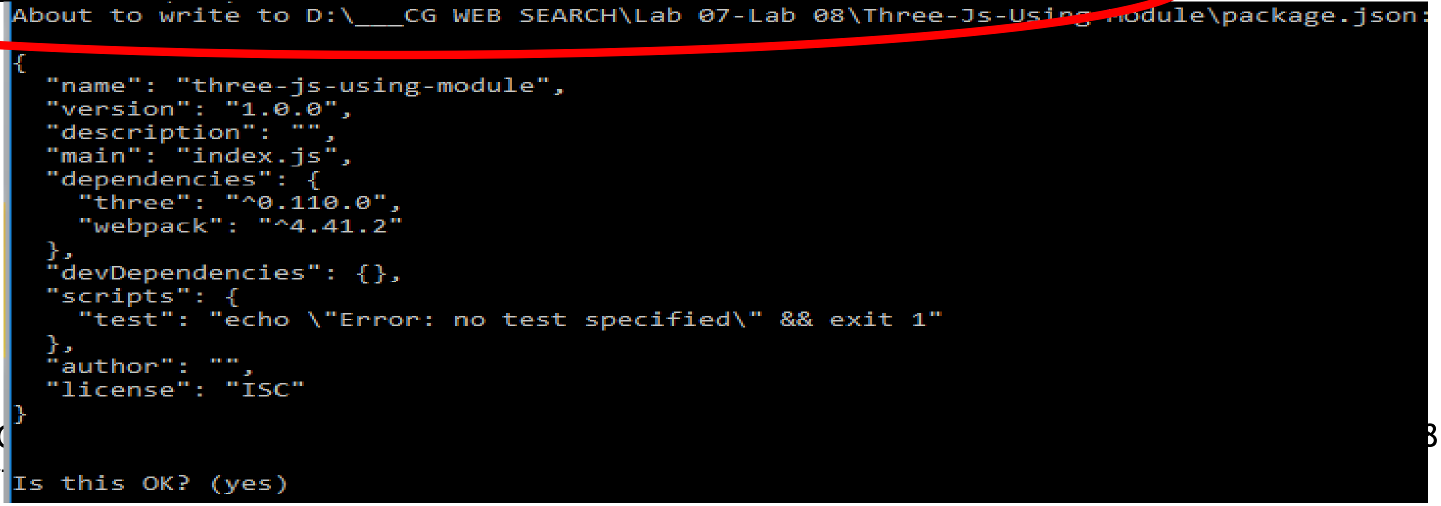
For the remaining questions, just hit Enter to accept all the default values. The end result of all of this is a new file called package.json that will be created in your Three-JS-Using-Module folder.

Simple press Enter again and again and say yes to generate package.json file

6



```
D:\___CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module>npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help json` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (three-js-using-module)
version: (1.0.0)
description:
entry point: (index.js)
test command:
git repository:
keywords:
author:
license: (ISC)
About to write to D:\___CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module\package.json:
```



```
About to write to D:\___CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module\package.json:

{
  "name": "three-js-using-module",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "dependencies": {
    "three": "^0.110.0",
    "webpack": "^4.41.2"
  },
  "devDependencies": {},
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}


Is this OK? (yes)
```

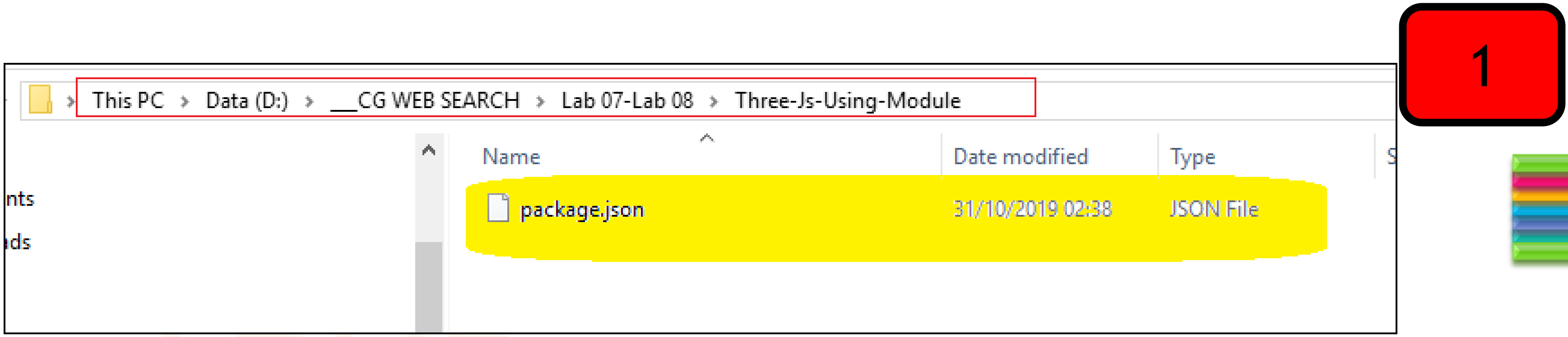
Step 02:
- Install Three.js and Webpack

Here is the link that I got help

https://webpack.js.org/guides/getting-started/

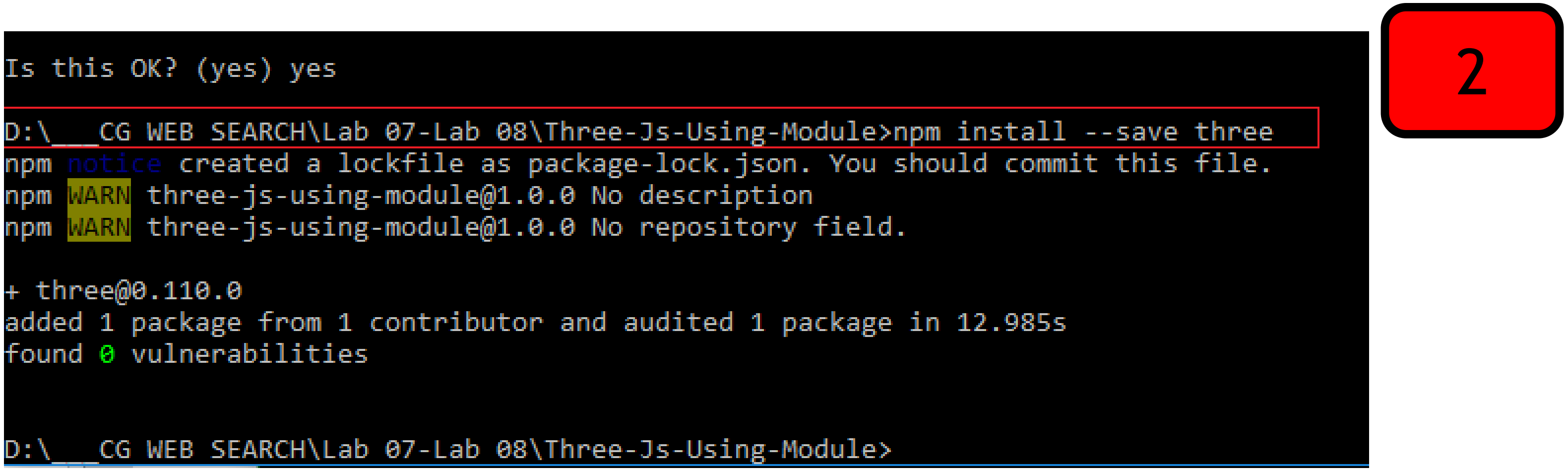
-

Package.json is ready and its time to install Three.js


1
This PC › Data (D:) › __CG WEB SEARCH › Lab 07-Lab 08 › Three-Js-Using-Module
Name
Date modified
Type
nts
ids

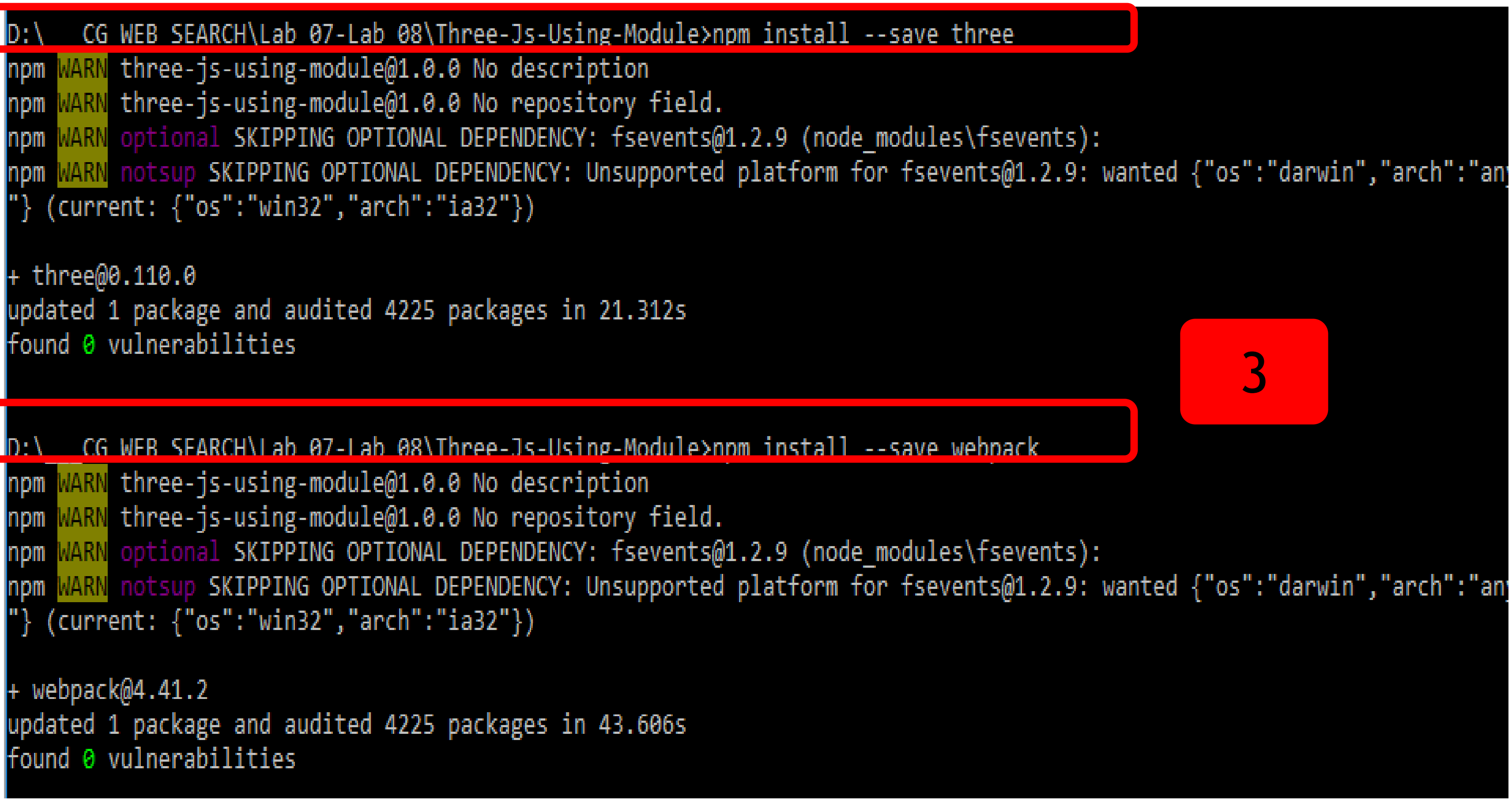
Name	Date modified	Type
package.json	31/10/2019 02:38	JSON File

Now we will install three.js library. Type the following command and press enter npm install --save three



```
Is this OK? (yes) yes

D:\___CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module>npm install --save three
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN three-js-using-module@1.0.0 No description
npm WARN three-js-using-module@1.0.0 No repository field.

+ three@0.110.0
added 1 package from 1 contributor and audited 1 package in 12.985s
found 0 vulnerabilities


D:\___CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module>
```

Install Three.js and WebPack



```
D:\   CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module>npm install --save three
npm WARN three-js-using-module@1.0.0 No description
npm WARN three-js-using-module@1.0.0 No repository field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.9 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.9: wanted {"os":"darwin","arch":"an
"} (current: {"os":"win32","arch":"ia32"})

+ three@0.110.0
updated 1 package and audited 4225 packages in 21.312s
found 0 vulnerabilities


D:\   CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module>npm install --save webpack
npm WARN three-js-using-module@1.0.0 No description
npm WARN three-js-using-module@1.0.0 No repository field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.9 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.9: wanted {"os":"darwin","arch":"an
"} (current: {"os":"win32","arch":"ia32"})

+ webpack@4.41.2
updated 1 package and audited 4225 packages in 43.606s
found 0 vulnerabilities
```

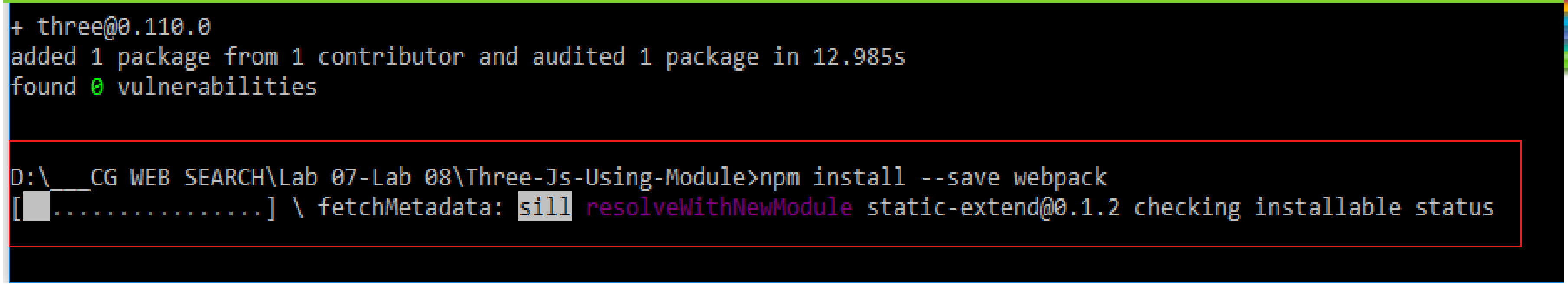
3

After Installing webpack Observe Project Structure

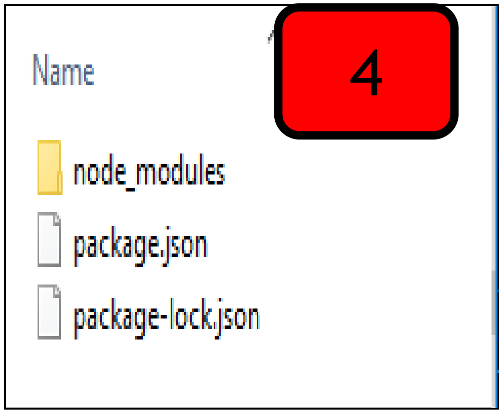
We have used following command to install webpack

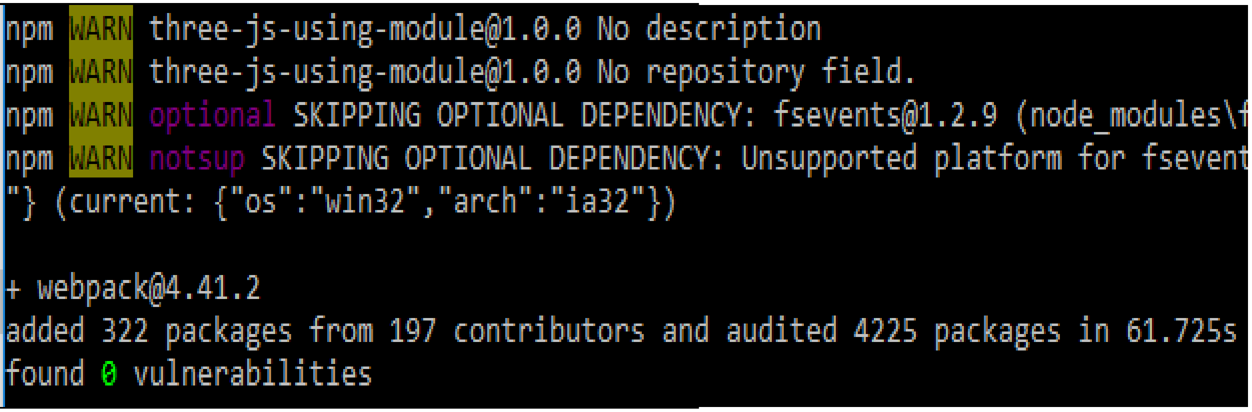
npm install --save webpack

This will take a few moments while the webpack package (and its large list of dependencies) gets downloaded and placed into our node_modules folder.



```
+ three@0.110.0
added 1 package from 1 contributor and audited 1 package in 12.985s
found 0 vulnerabilities

D:\___CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module>npm install --save webpack
[█...............] \ fetchMetadata: sill resolveWithNewModule static-extend@0.1.2 checking installable status
```

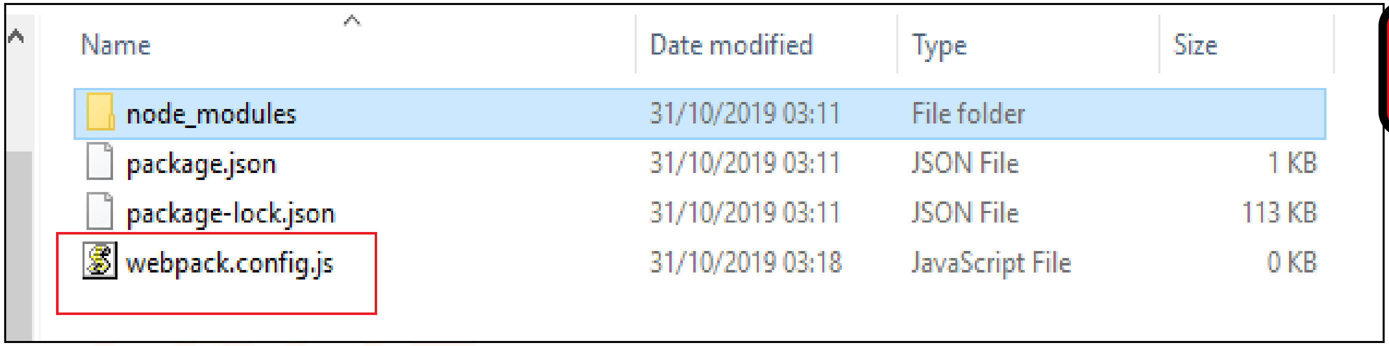




```
npm WARN three-js-using-module@1.0.0 No description
npm WARN three-js-using-module@1.0.0 No repository field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.9 (node_modules\f
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevent
"} (current: {"os":"win32","arch":"ia32"})

+ webpack@4.41.2
added 322 packages from 197 contributors and audited 4225 packages in 61.725s
found 0 vulnerabilities
```

Step 03:
Create index.html and index.js

After you've done this, we need to add a configuration file to specify how webpack will work with our current project. Using your code editor, add a file called webpack.config.js inside our Three-JS-Using-Module folder.

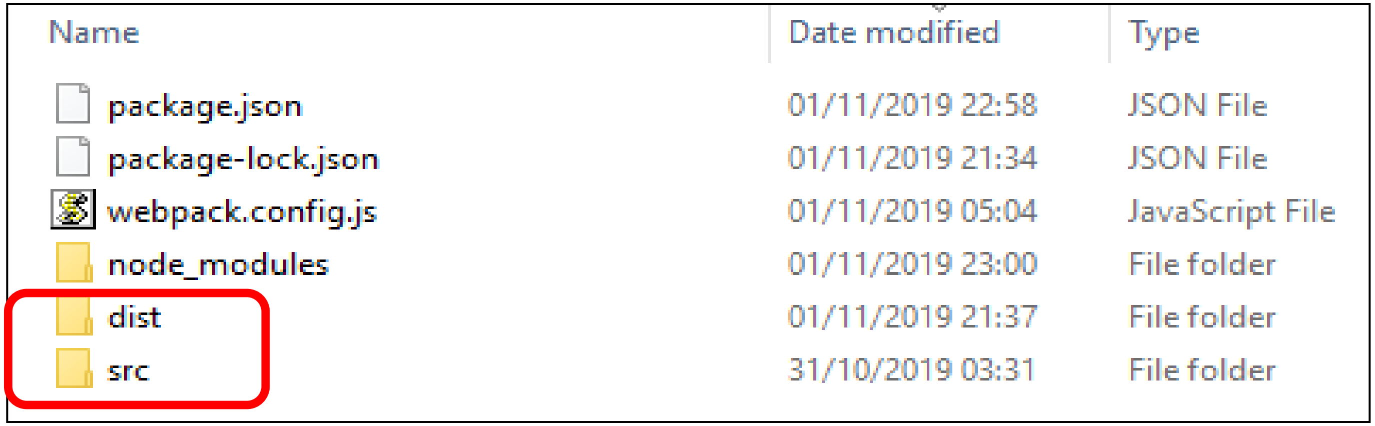


Name	Date modified	Type	Size
node_modules	31/10/2019 03:11	File folder	
package.json	31/10/2019 03:11	JSON File	1 KB
package-lock.json	31/10/2019 03:11	JSON File	113 KB
webpack.config.js	31/10/2019 03:18	JavaScript File	0 KB

1

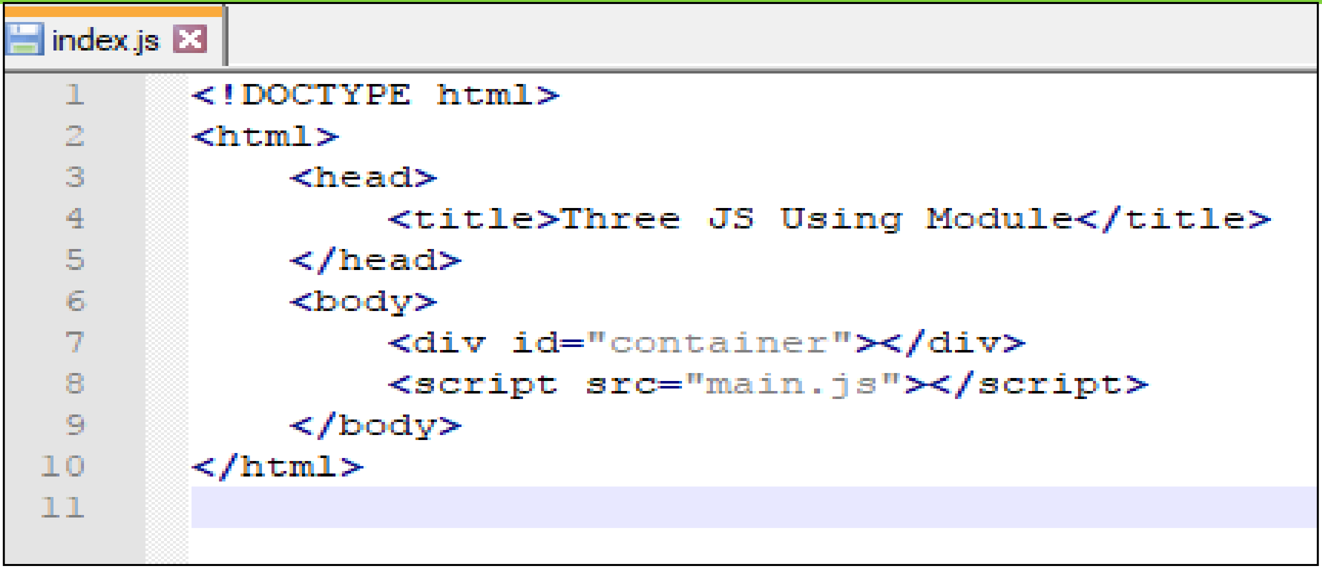
Now create a src folder and dist folder.

2



Name	Date modified	Type
package.json	01/11/2019 22:58	JSON File
package-lock.json	01/11/2019 21:34	JSON File
webpack.config.js	01/11/2019 05:04	JavaScript File
node_modules	01/11/2019 23:00	File folder
dist	01/11/2019 21:37	File folder
src	31/10/2019 03:31	File folder

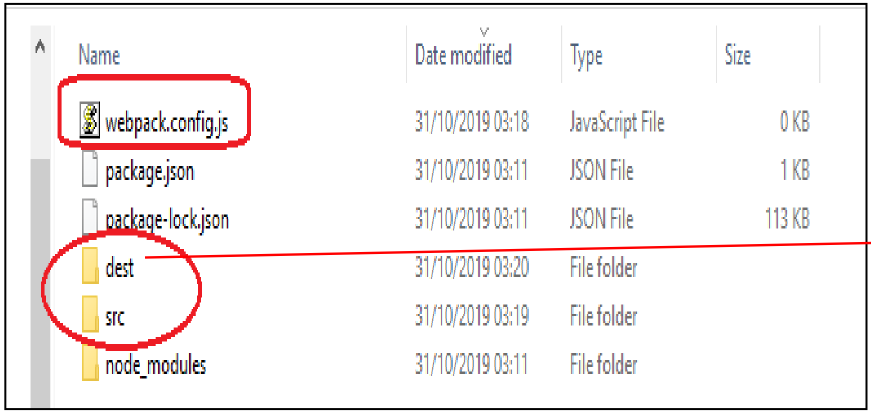
Create index.html into distribution folder i.e. dist



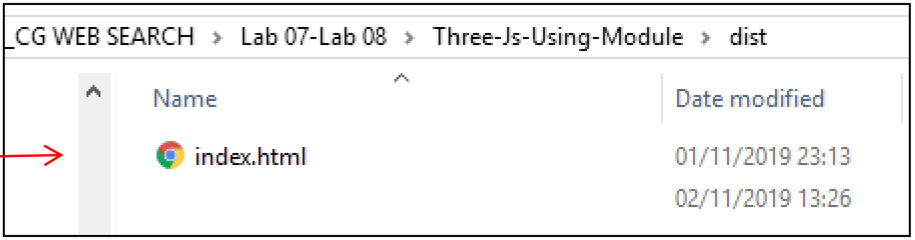
index.js

```html
<!DOCTYPE html>
<html>
    <head>
        <title>Three JS Using Module</title>
    </head>
    <body>
        <div id="container"></div>
        <script src="main.js"></script>
    </body>
</html>

```

3



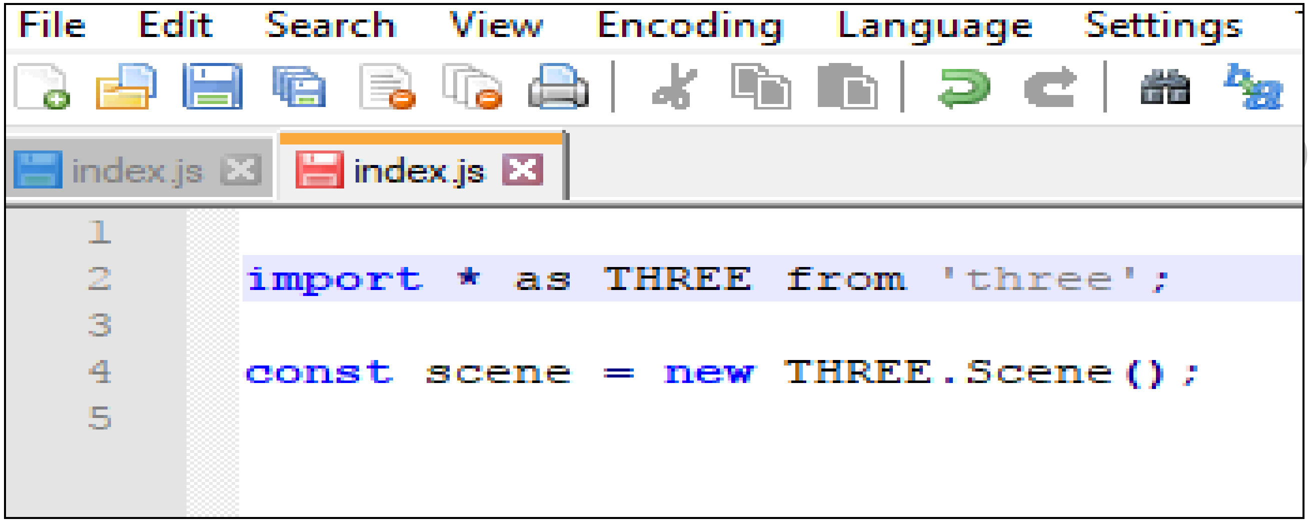
Name	Date modified	Type	Size
webpack.config.js	31/10/2019 03:18	JavaScript File	0 KB
package.json	31/10/2019 03:11	JSON File	1 KB
package-lock.json	31/10/2019 03:11	JSON File	113 KB
dest	31/10/2019 03:20	File folder	
src	31/10/2019 03:19	File folder	
node_modules	31/10/2019 03:11	File folder	



_CG WEB SEARCH › Lab 07-Lab 08 › Three-Js-Using-Module › dist

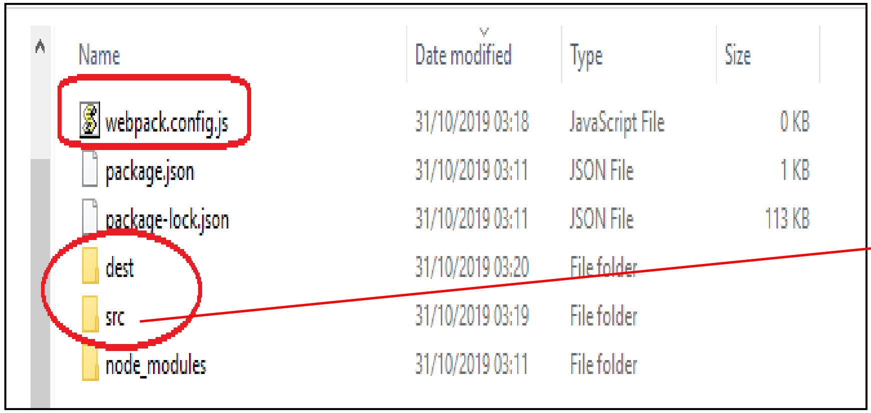
Name	Date modified
index.html	01/11/2019 23:13
	02/11/2019 13:26

Create index.js and put it into source folder i.e. src

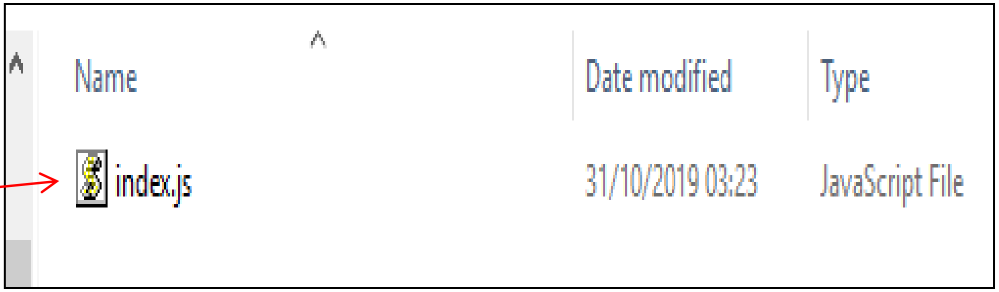


```
import * as THREE from 'three';

const scene = new THREE.Scene();
```

4



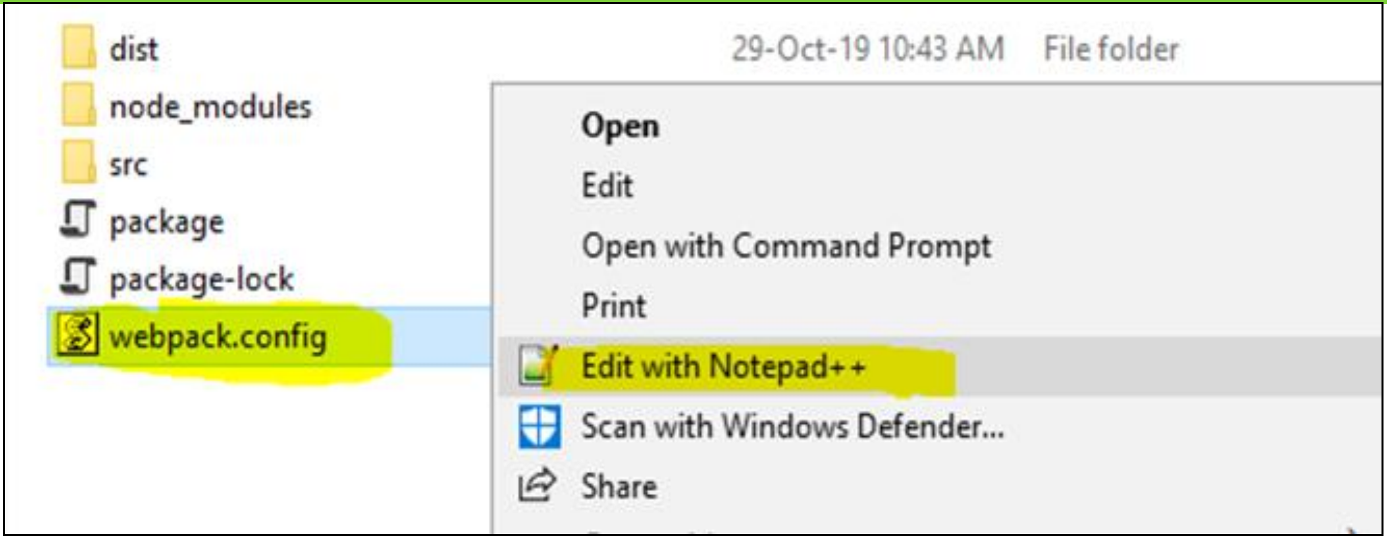
Name	Date modified	Type	Size
webpack.config.js	31/10/2019 03:18	JavaScript File	0 KB
package.json	31/10/2019 03:11	JSON File	1 KB
package-lock.json	31/10/2019 03:11	JSON File	113 KB
dest	31/10/2019 03:20	File folder	
src	31/10/2019 03:19	File folder	
node_modules	31/10/2019 03:11	File folder	



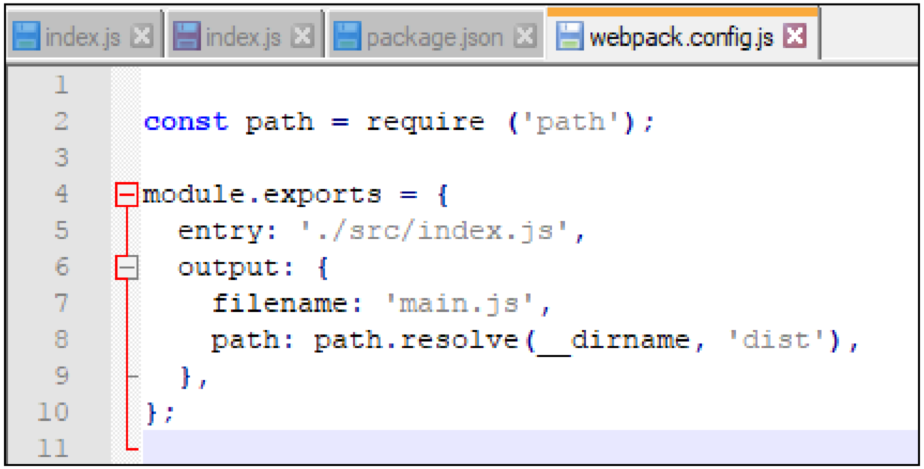
Name	Date modified	Type
index.js	31/10/2019 03:23	JavaScript File

Step 05:
Build and Run with webpack.config.js and package.json

Now open webpack.config.js file and write given content into it.



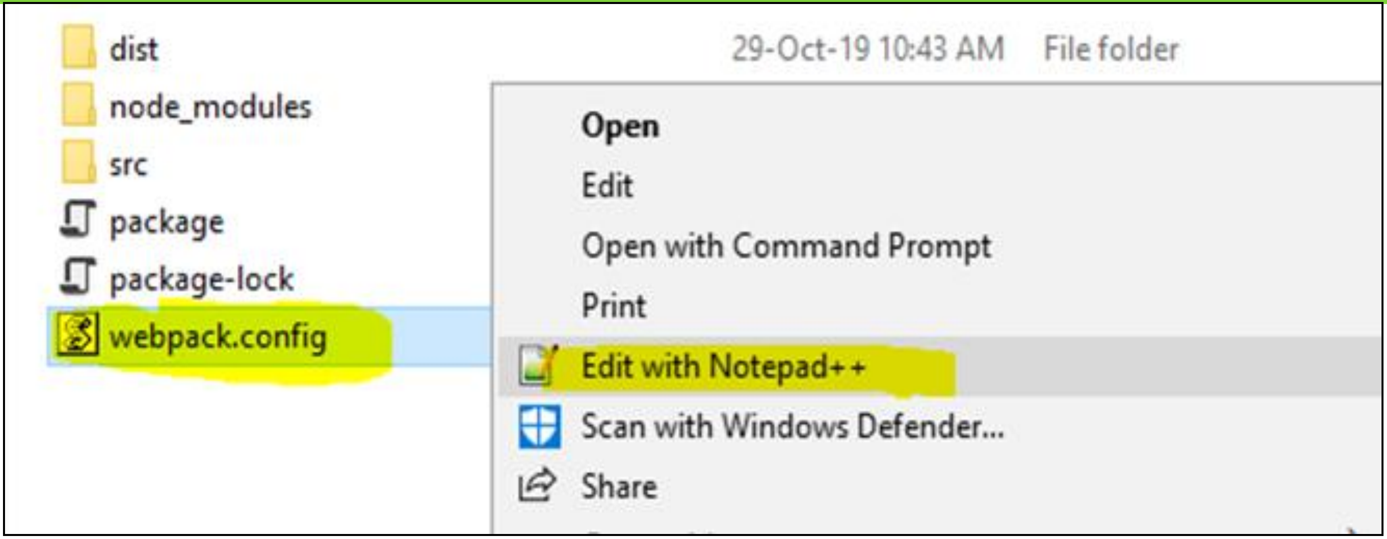
1



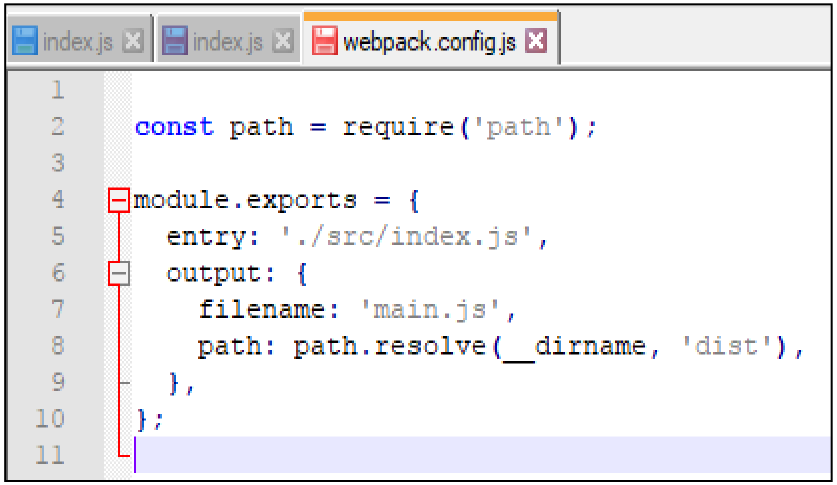
```

const path = require ('path');

module.exports = {
  entry: './src/index.js',
  output: {
    filename: 'main.js',
    path: path.resolve(__dirname, 'dist'),
  },
};

```

2

Now open webpack.config.js file and write given content into it.



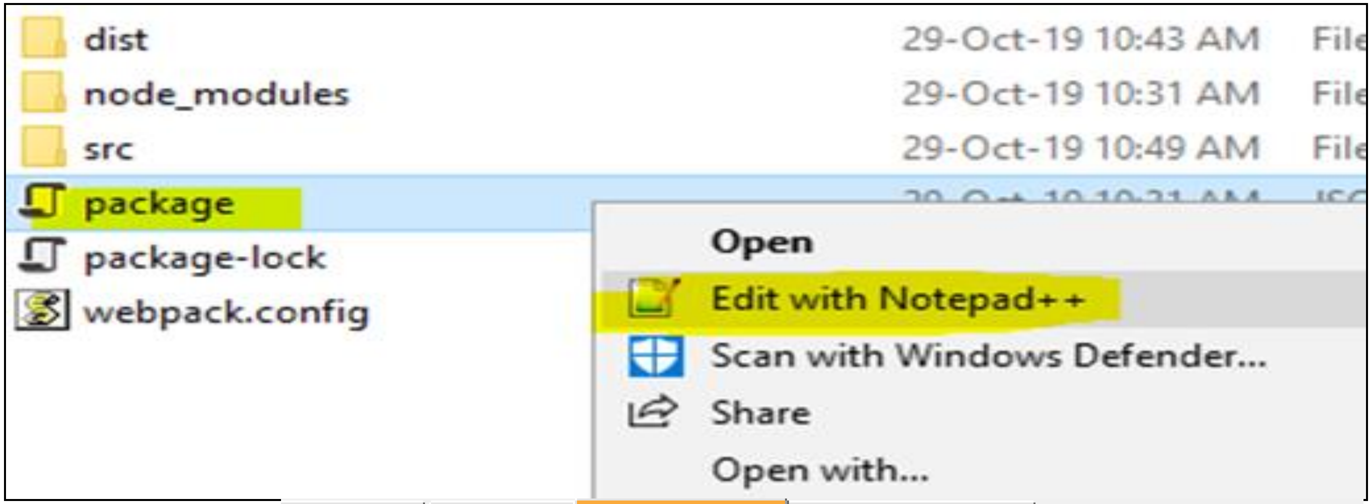
1



```

const path = require('path');

module.exports = {
  entry: './src/index.js',
  output: {
    filename: 'main.js',
    path: path.resolve(__dirname, 'dist'),
  },
};

```

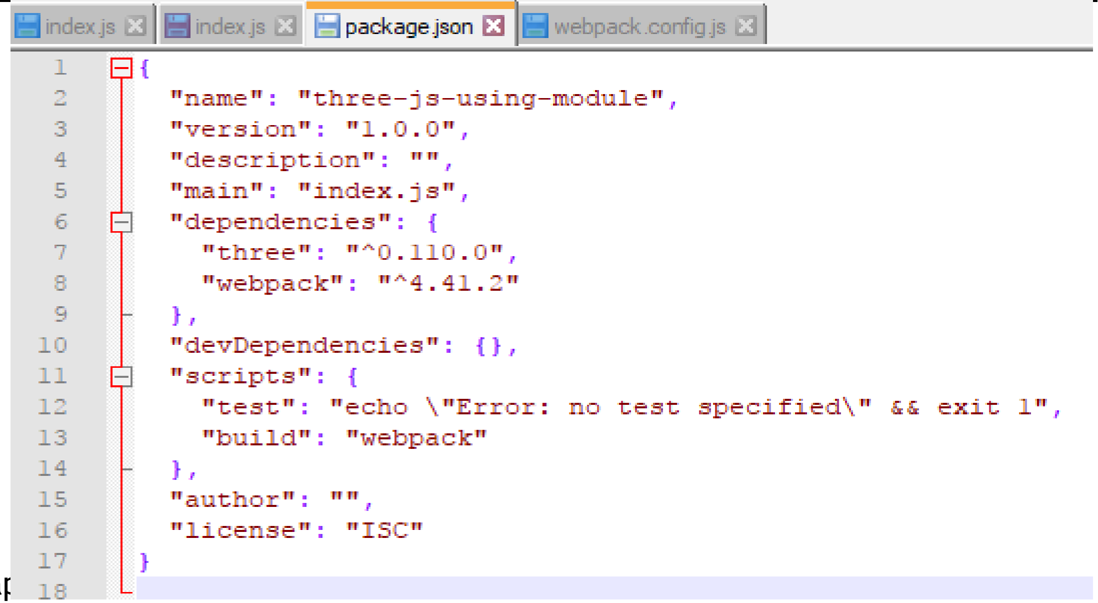
2

Now open package.json file



```
dist               29-Oct-19 10:43 AM   File
node_modules       29-Oct-19 10:31 AM   File
src                29-Oct-19 10:49 AM   File
package
package-lock
webpack.config
```

3



```json
{
  "name": "three-js-using-module",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "dependencies": {
    "three": "^0.110.0",
    "webpack": "^4.41.2"
  },
  "devDependencies": {},
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "build": "webpack"
  },
  "author": "",
  "license": "ISC"
}
```

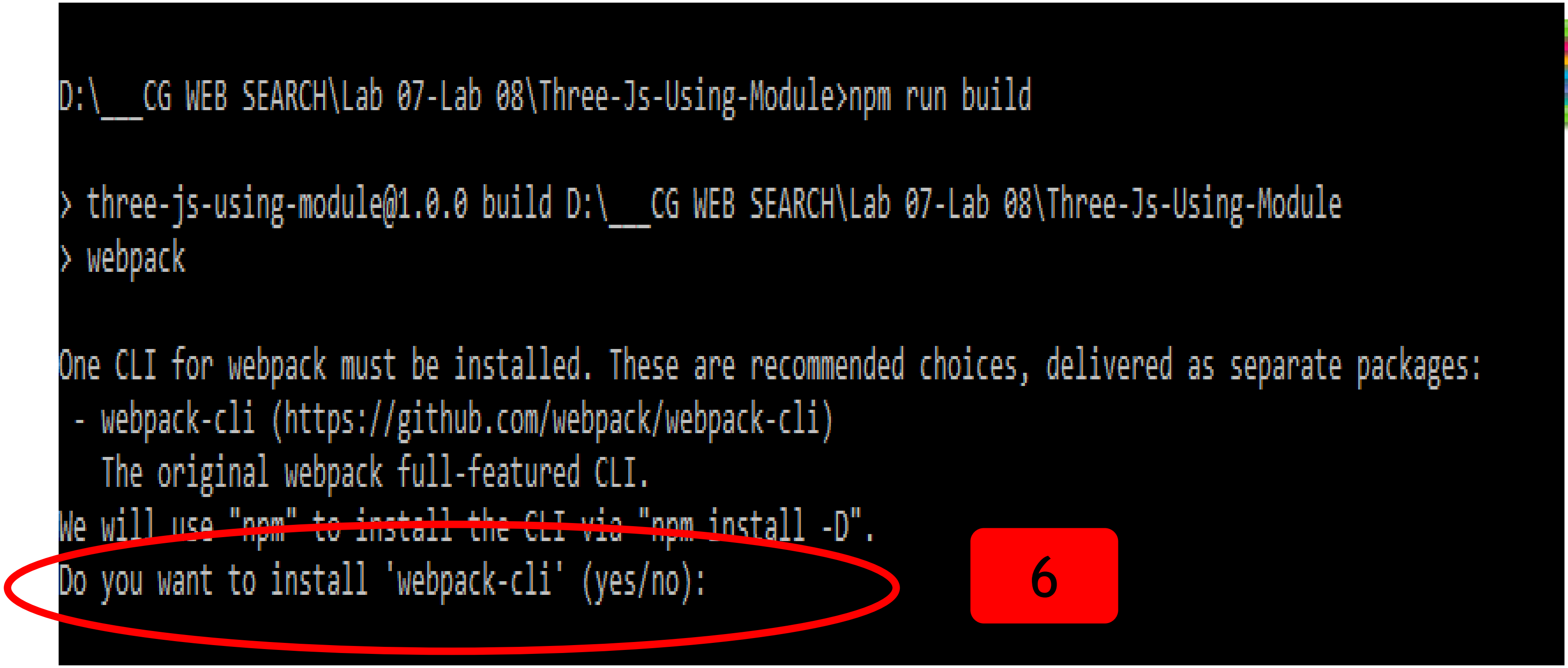
4

Add the line "build": "webpack"

Now run the following command. If webpack cli is not installed. It will ask for it. Type yes and press enter

```
npm run build
```

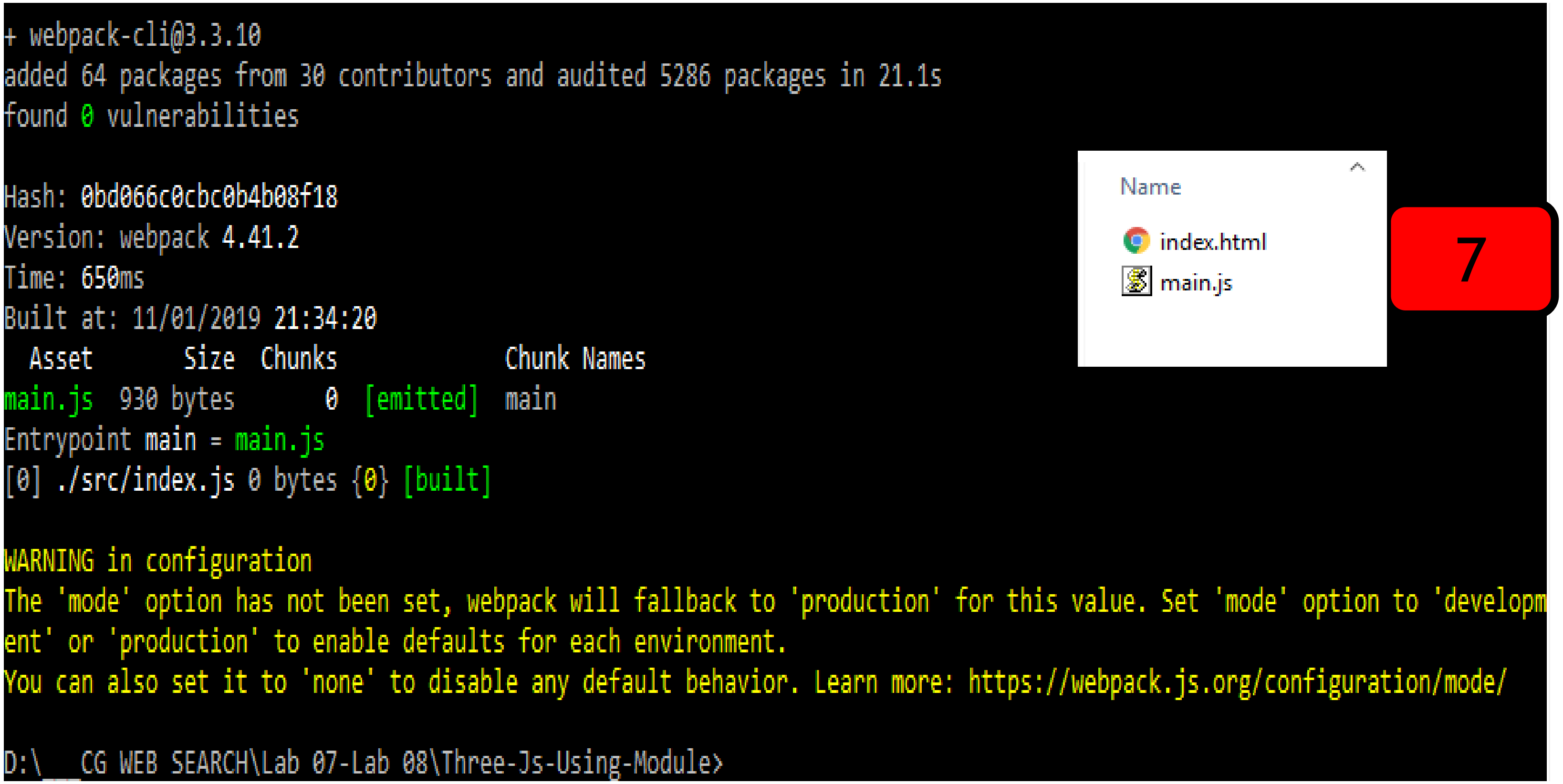
5



```
D:\__CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module>npm run build

> three-js-using-module@1.0.0 build D:\__CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module
> webpack

One CLI for webpack must be installed. These are recommended choices, delivered as separate packages:
 - webpack-cli (https://github.com/webpack/webpack-cli)
   The original webpack full-featured CLI.
We will use "npm" to install the CLI via "npm install -D".
Do you want to install 'webpack-cli' (yes/no):
```

6

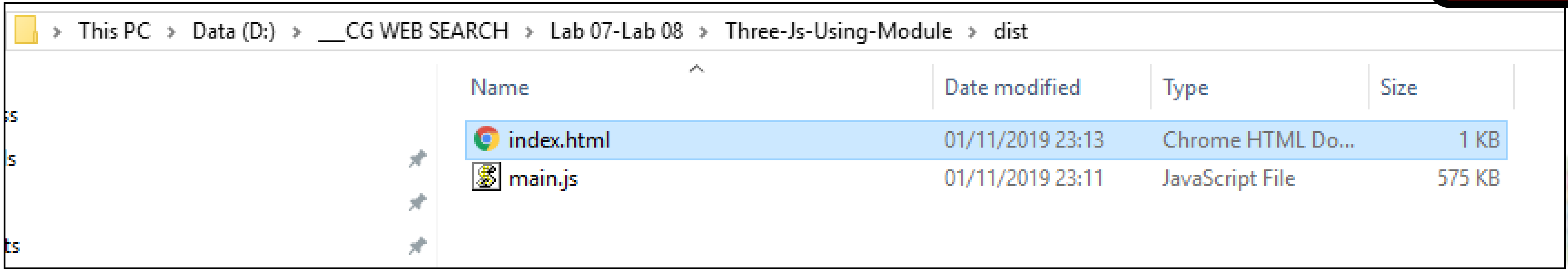
Cli create main.js



```
+ webpack-cli@3.3.10
added 64 packages from 30 contributors and audited 5286 packages in 21.1s
found 0 vulnerabilities

Hash: 0bd066c0cbc0b4b08f18
Version: webpack 4.41.2
Time: 650ms
Built at: 11/01/2019 21:34:20
  Asset       Size  Chunks             Chunk Names
main.js  930 bytes       0  [emitted]  main
Entrypoint main = main.js
[0] ./src/index.js 0 bytes {0} [built]

WARNING in configuration
The 'mode' option has not been set, webpack will fallback to 'production' for this value. Set 'mode' option to 'developm
ent' or 'production' to enable defaults for each environment.
You can also set it to 'none' to disable any default behavior. Learn more: https://webpack.js.org/configuration/mode/

D:\   CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module>
```

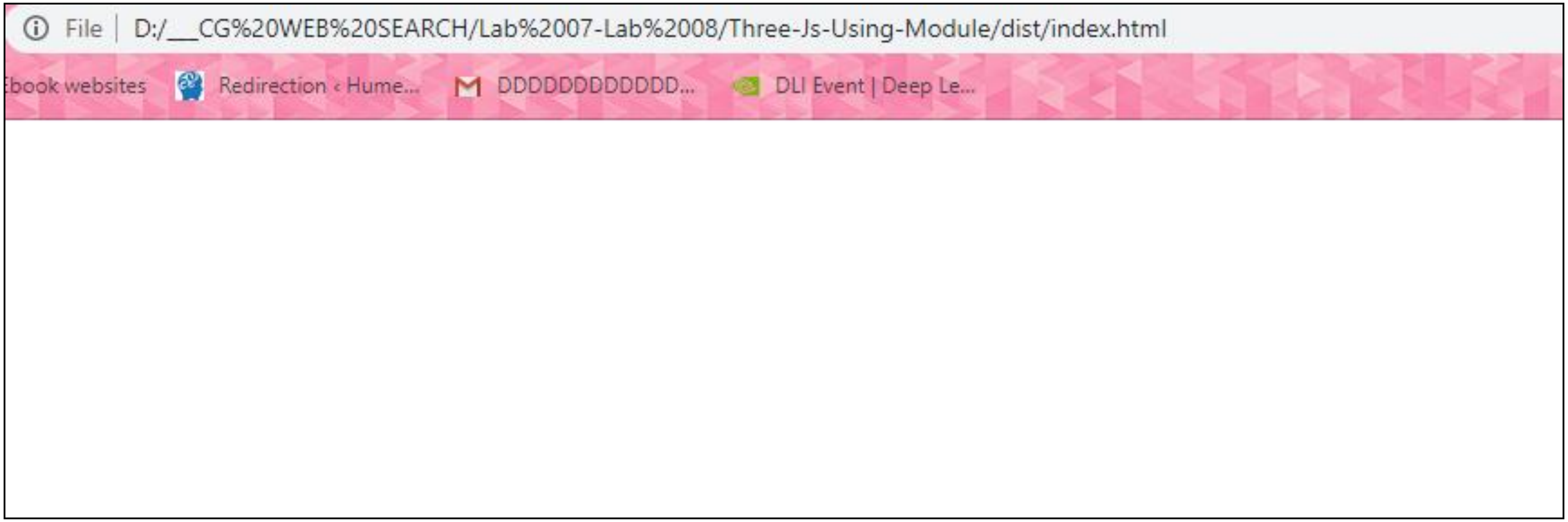
Run index.html but no output bcoz our index.js is almost empty

8



This PC › Data (D:) › __CG WEB SEARCH › Lab 07-Lab 08 › Three-Js-Using-Module › dist

Name	Date modified	Type	Size
index.html	01/11/2019 23:13	Chrome HTML Do...	1 KB
main.js	01/11/2019 23:11	JavaScript File	575 KB



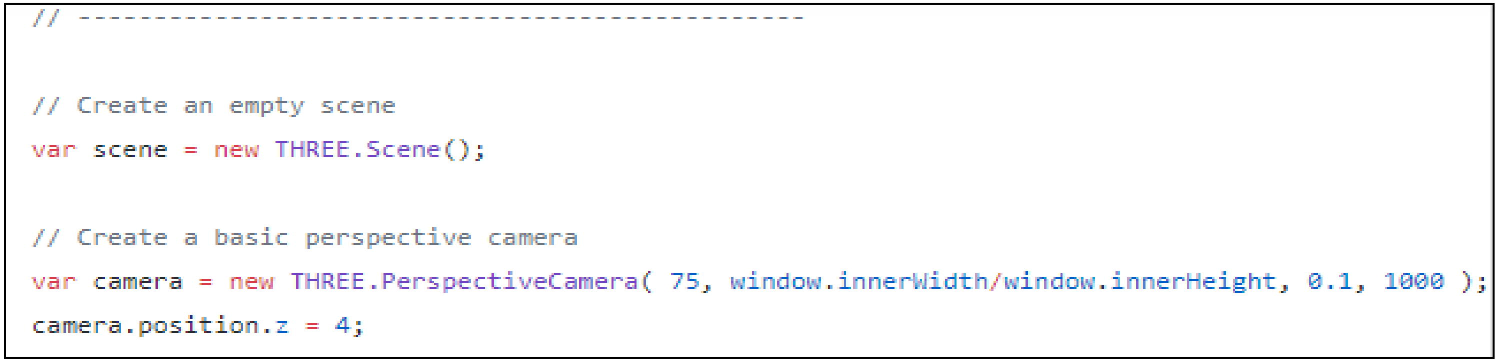
File | D:/__CG%20WEB%20SEARCH/Lab%2007-Lab%2008/Three-Js-Using-Module/dist/index.html

Go to given link and copy code to paste into our empty index.js

9

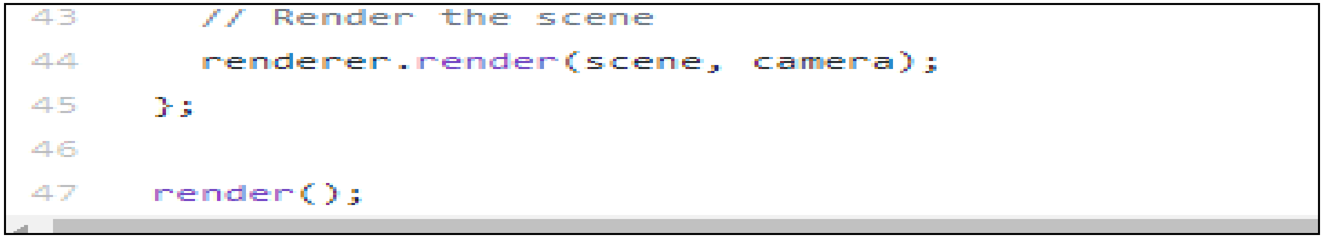
https://medium.com/@necsoft/three-js-101-hello-world-part-1-443207b1ebe1

Start of code to copy



```
// ------------------------------------------------


// Create an empty scene
var scene = new THREE.Scene();

// Create a basic perspective camera
var camera = new THREE.PerspectiveCamera( 75, window.innerWidth/window.innerHeight, 0.1, 1000 );
camera.position.z = 4;
```

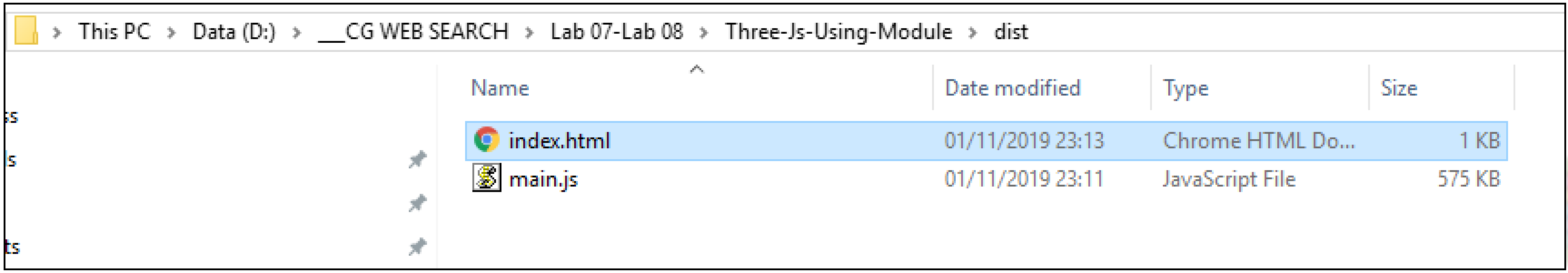
end of code to copy



```
        // Render the scene
        renderer.render(scene, camera);
    };

    render();
```

Run index.html but build first using npm run build

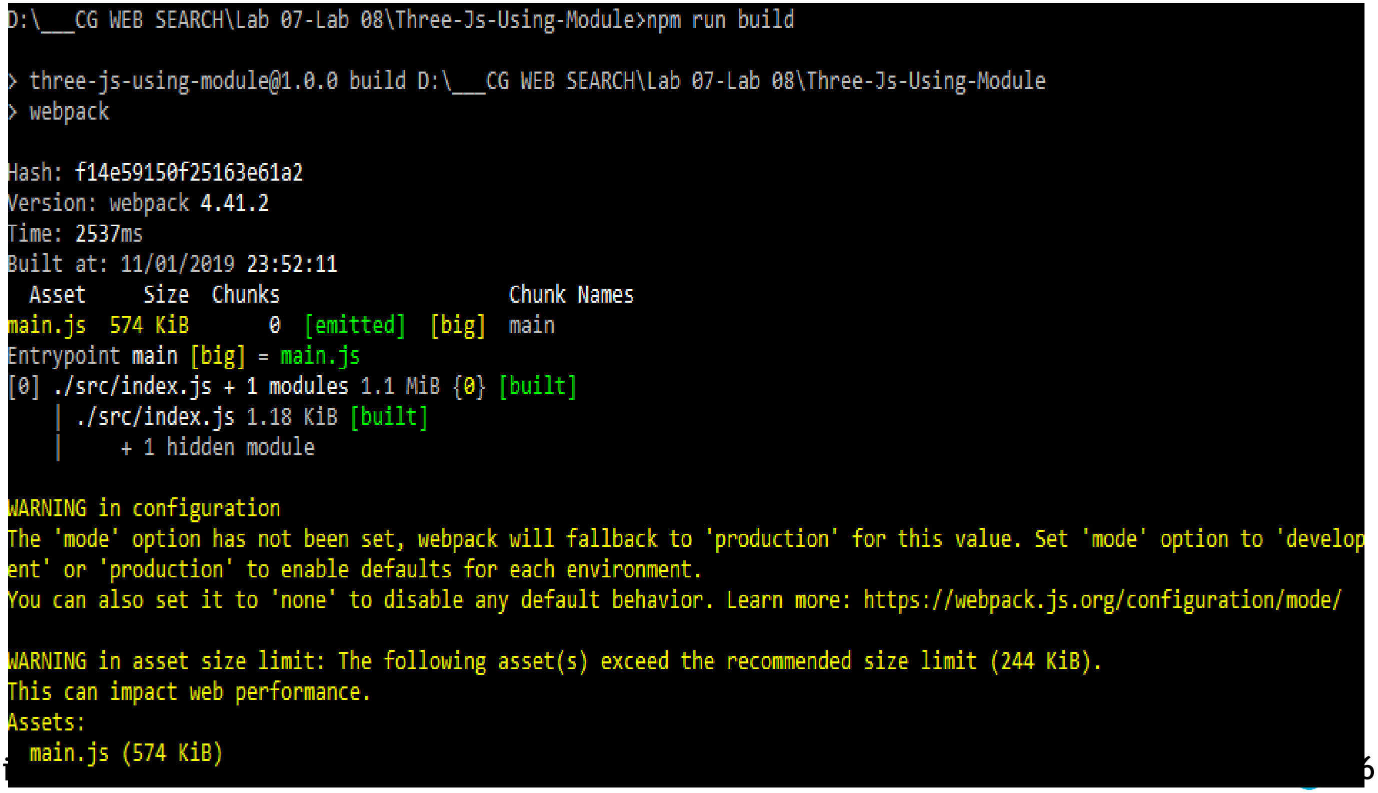
10



This PC › Data (D:) › __CG WEB SEARCH › Lab 07-Lab 08 › Three-Js-Using-Module › dist

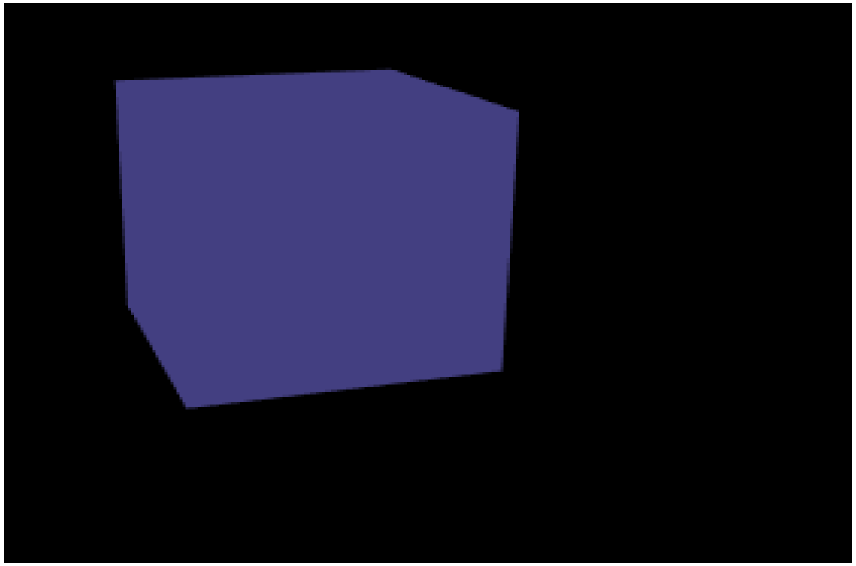
Name	Date modified	Type	Size
index.html	01/11/2019 23:13	Chrome HTML Do...	1 KB
main.js	01/11/2019 23:11	JavaScript File	575 KB

Since you changed index.js so you need to run npm run build again and refresh



```
D:\__CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module>npm run build

> three-js-using-module@1.0.0 build D:\__CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module
> webpack

Hash: f14e59150f25163e61a2
Version: webpack 4.41.2
Time: 2537ms
Built at: 11/01/2019 23:52:11
  Asset     Size  Chunks                    Chunk Names
main.js  574 KiB       0  [emitted]  [big]  main
Entrypoint main [big] = main.js
[0] ./src/index.js + 1 modules 1.1 MiB {0} [built]
    | ./src/index.js 1.18 KiB [built]
    |     + 1 hidden module

WARNING in configuration
The 'mode' option has not been set, webpack will fallback to 'production' for this value. Set 'mode' option to 'develop
ent' or 'production' to enable defaults for each environment.
You can also set it to 'none' to disable any default behavior. Learn more: https://webpack.js.org/configuration/mode/

WARNING in asset size limit: The following asset(s) exceed the recommended size limit (244 KiB).
This can impact web performance.
Assets:
  main.js (574 KiB)
```

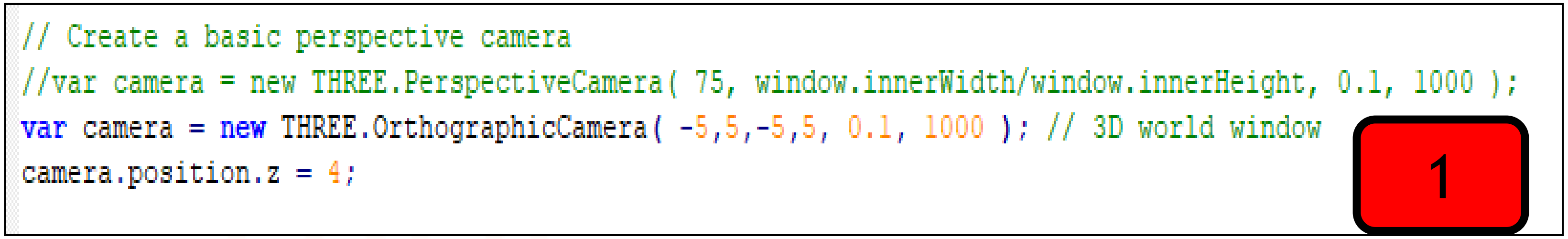
First Three.js Output : A rotating cube perspective view



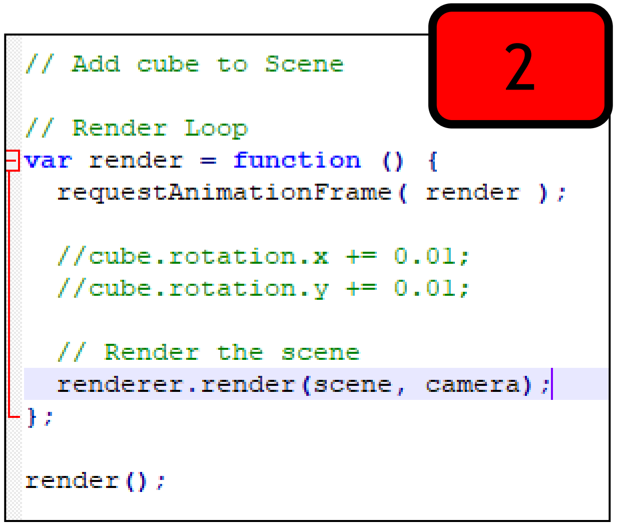
11

Transformation Basic code

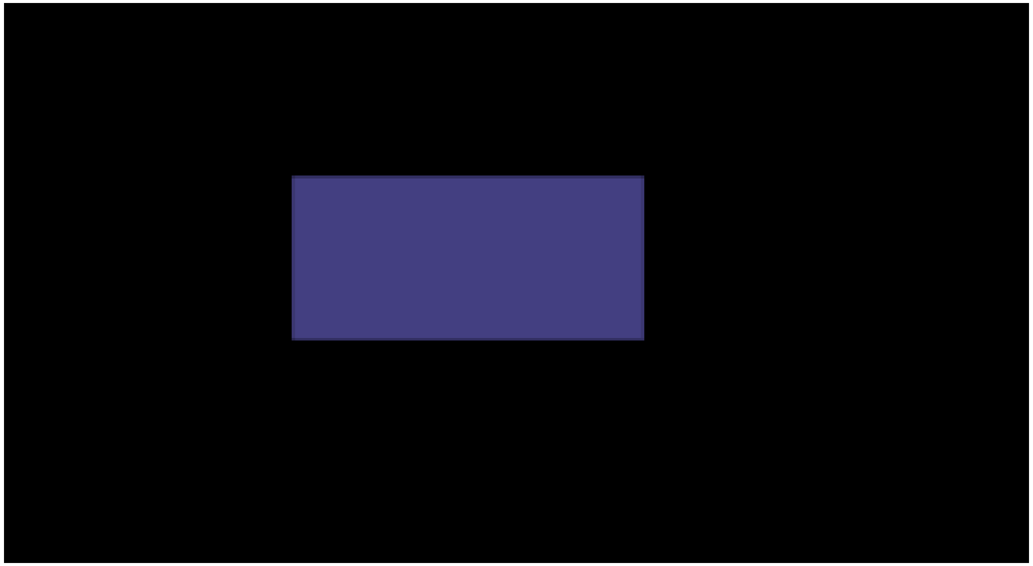
Modify code to set 2D Camera and stop rotation



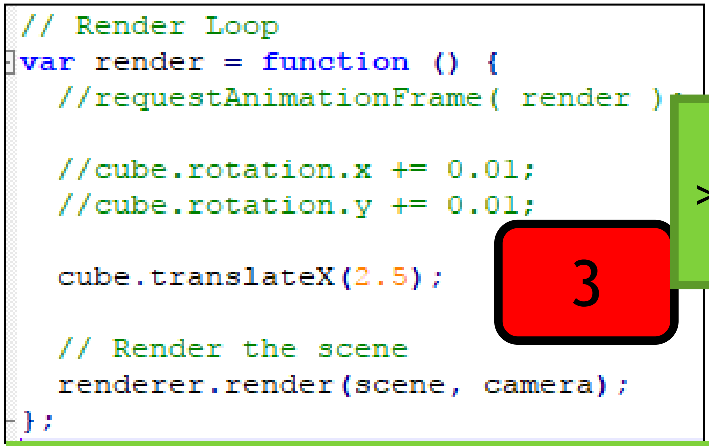
```
// Create a basic perspective camera
//var camera = new THREE.PerspectiveCamera( 75, window.innerWidth/window.innerHeight, 0.1, 1000 );
var camera = new THREE.OrthographicCamera( -5,5,-5,5, 0.1, 1000 ); // 3D world window
camera.position.z = 4;
```



```
// Add cube to Scene

// Render Loop
var render = function () {
  requestAnimationFrame( render );

  //cube.rotation.x += 0.01;
  //cube.rotation.y += 0.01;

  // Render the scene
  renderer.render(scene, camera);
};

render();
```



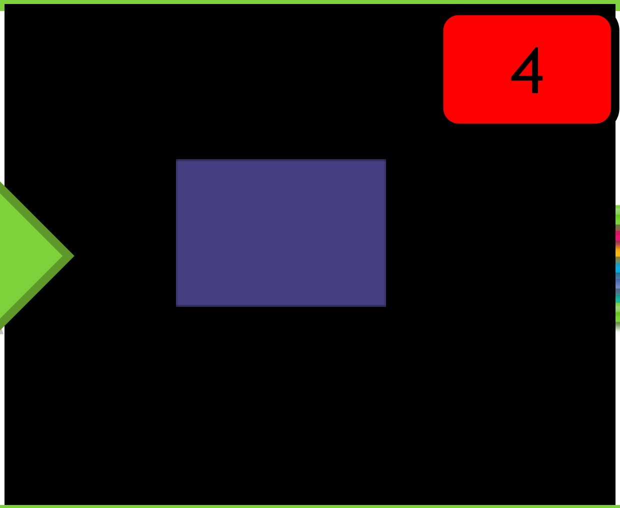
Stop Timer to understand transformations



```
// Render Loop
var render = function () {
  //requestAnimationFrame( render )

  //cube.rotation.x += 0.01;
  //cube.rotation.y += 0.01;

  cube.translateX(2.5);

  // Render the scene
  renderer.render(scene, camera);
};
```

3

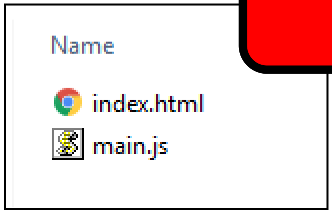
> npm run build



4

To avoid build again and again use watch

5: delete main.js in dist folder

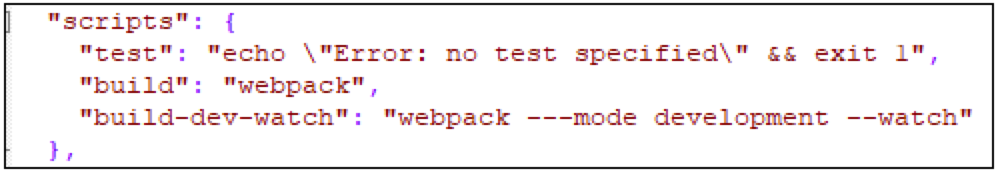


Name

index.html

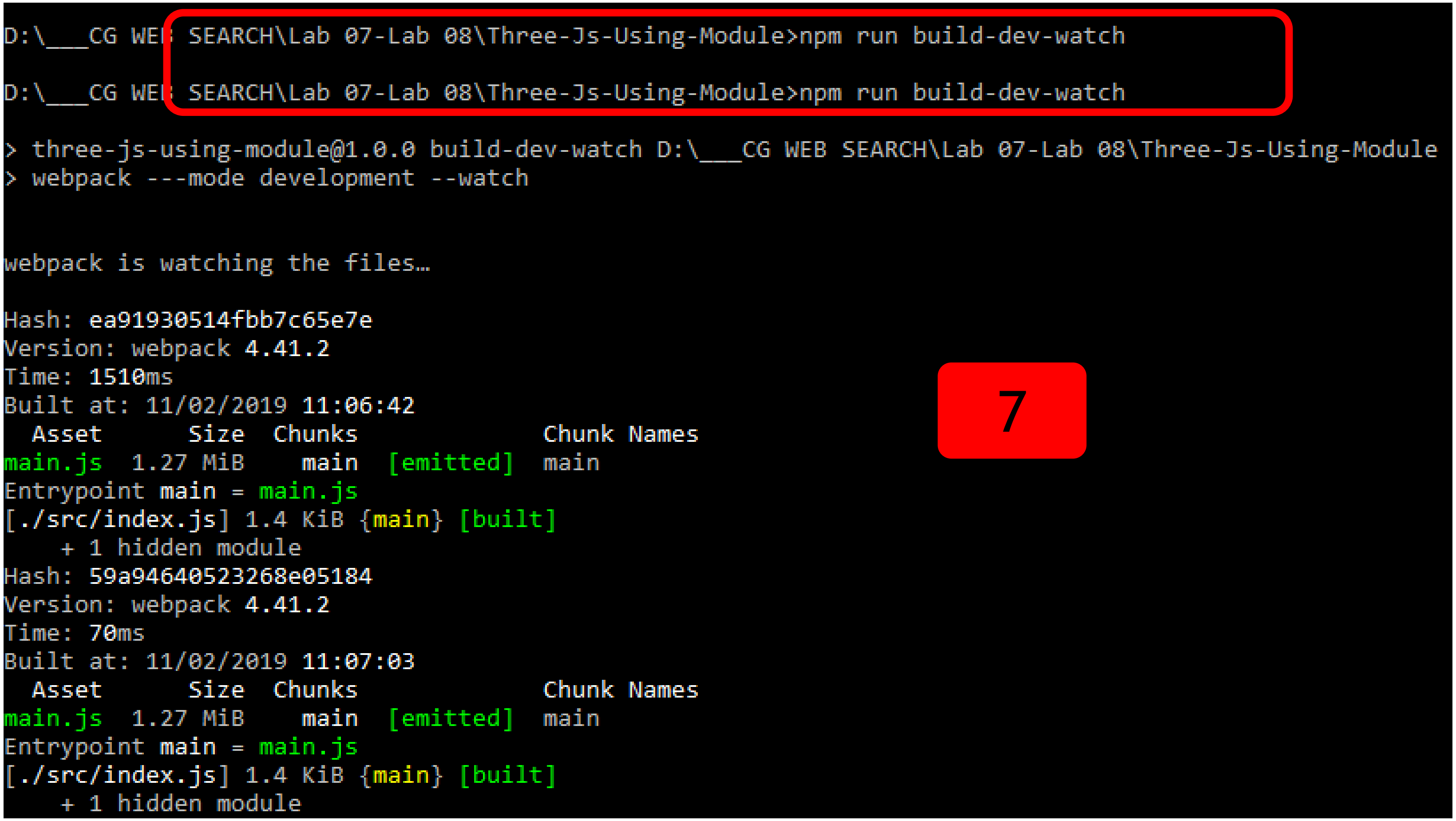
main.js

6: Add watch command in package.json



```
"scripts": {
  "test": "echo \"Error: no test specified\" && exit 1",
  "build": "webpack",
  "build-dev-watch": "webpack ---mode development --watch"
},
```

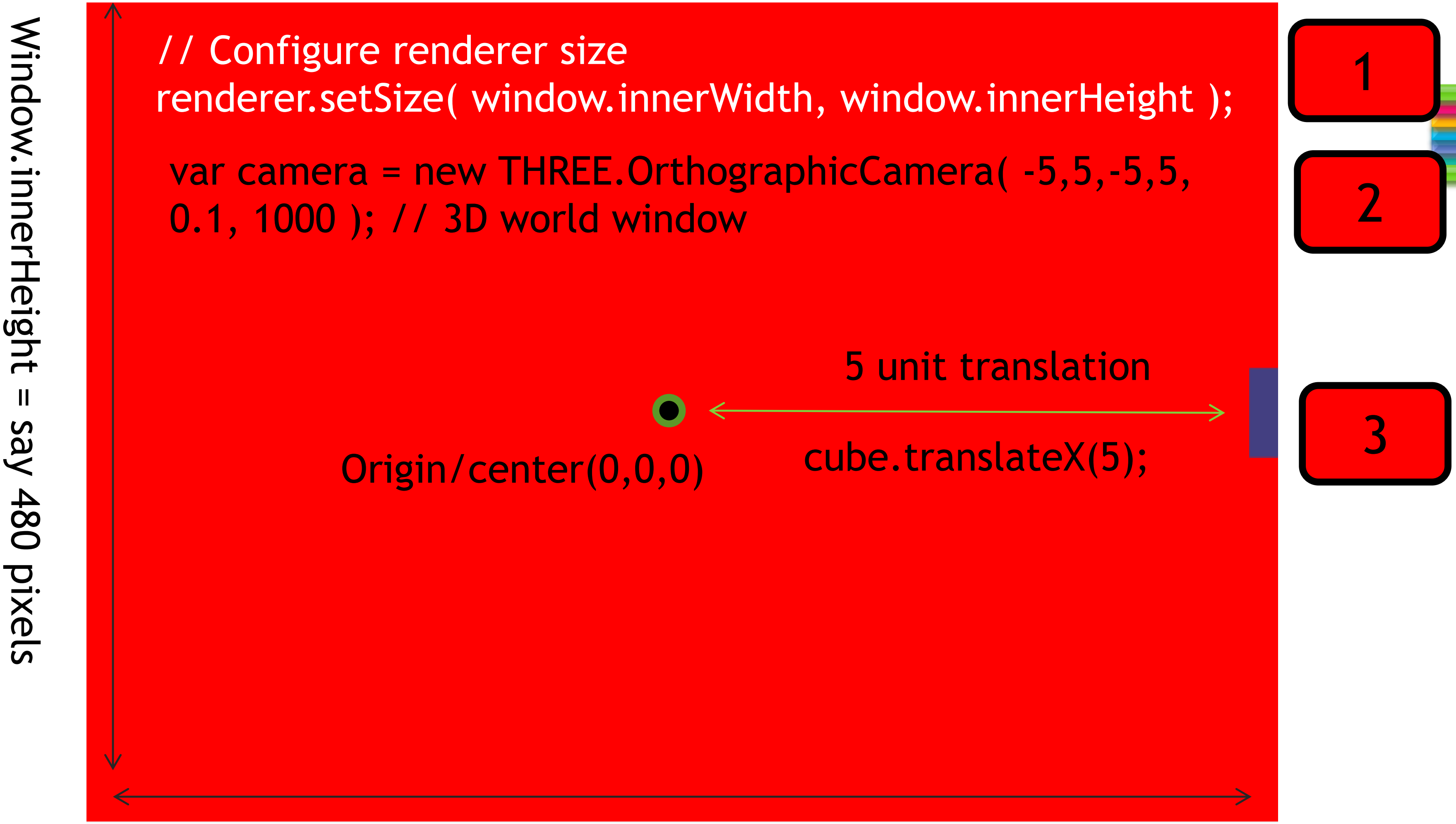
Now run watch ,do changes and just refresh html



```
D:\___CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module>npm run build-dev-watch

D:\___CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module>npm run build-dev-watch

> three-js-using-module@1.0.0 build-dev-watch D:\___CG WEB SEARCH\Lab 07-Lab 08\Three-Js-Using-Module
> webpack ---mode development --watch


webpack is watching the files…

Hash: ea91930514fbb7c65e7e
Version: webpack 4.41.2
Time: 1510ms
Built at: 11/02/2019 11:06:42
  Asset      Size  Chunks             Chunk Names
main.js  1.27 MiB    main  [emitted]  main
Entrypoint main = main.js
[./src/index.js] 1.4 KiB {main} [built]
    + 1 hidden module
Hash: 59a94640523268e05184
Version: webpack 4.41.2
Time: 70ms
Built at: 11/02/2019 11:07:03
  Asset      Size  Chunks             Chunk Names
main.js  1.27 MiB    main  [emitted]  main
Entrypoint main = main.js
[./src/index.js] 1.4 KiB {main} [built]
    + 1 hidden module
```

7

World Window to VP Mapping concept



```
// Configure renderer size
renderer.setSize( window.innerWidth, window.innerHeight );

var camera = new THREE.OrthographicCamera( -5,5,-5,5,
0.1, 1000 ); // 3D world window

cube.translateX(5);
```

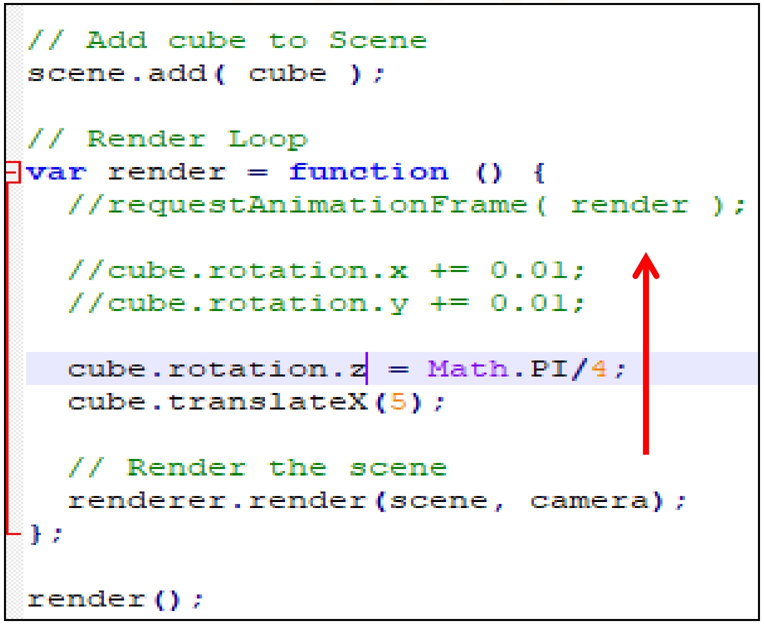
Window.innerHeight = say 480 pixels
1
2
5 unit translation
3
Origin/center(0,0,0)
Window.innerWidth = say 640 pixels

Object Transformation T + R (Rotation followed by Translation)

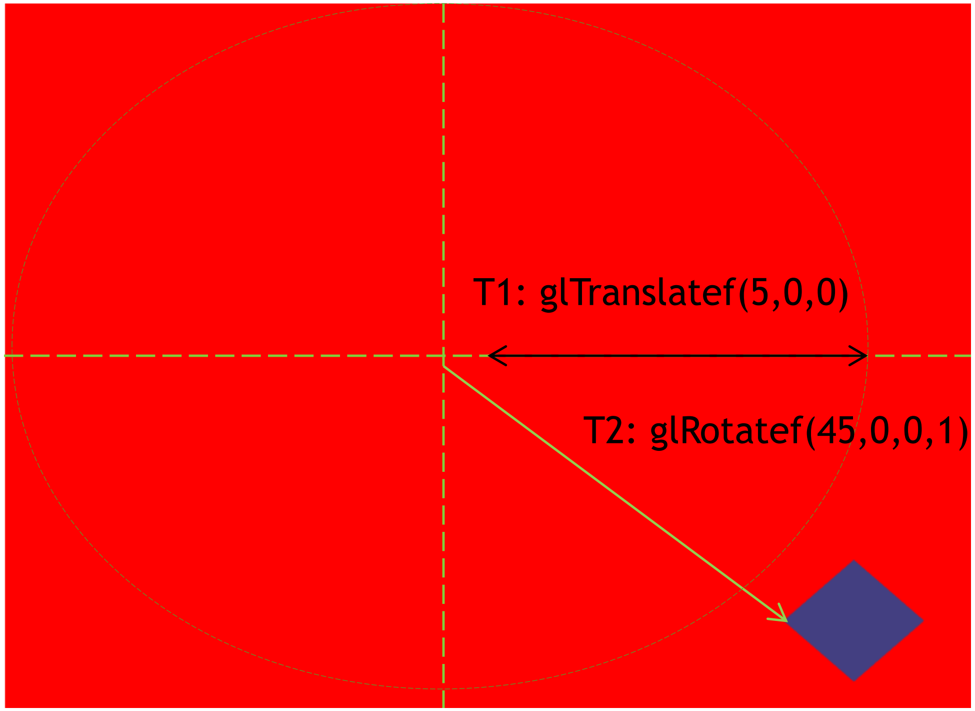
Composite Matrix = Rotation Matrix * Translation Matrix

M = R *T vs. M = T * R

M = T2 * T1 = R * T // object transformation (Read Right to Left)



```
// Add cube to Scene
scene.add( cube );

// Render Loop
var render = function () {
   //requestAnimationFrame( render );

   //cube.rotation.x += 0.01;
   //cube.rotation.y += 0.01;

   cube.rotation.z = Math.PI/4;
   cube.translateX(5);

   // Render the scene
   renderer.render(scene, camera);
};

render();
```

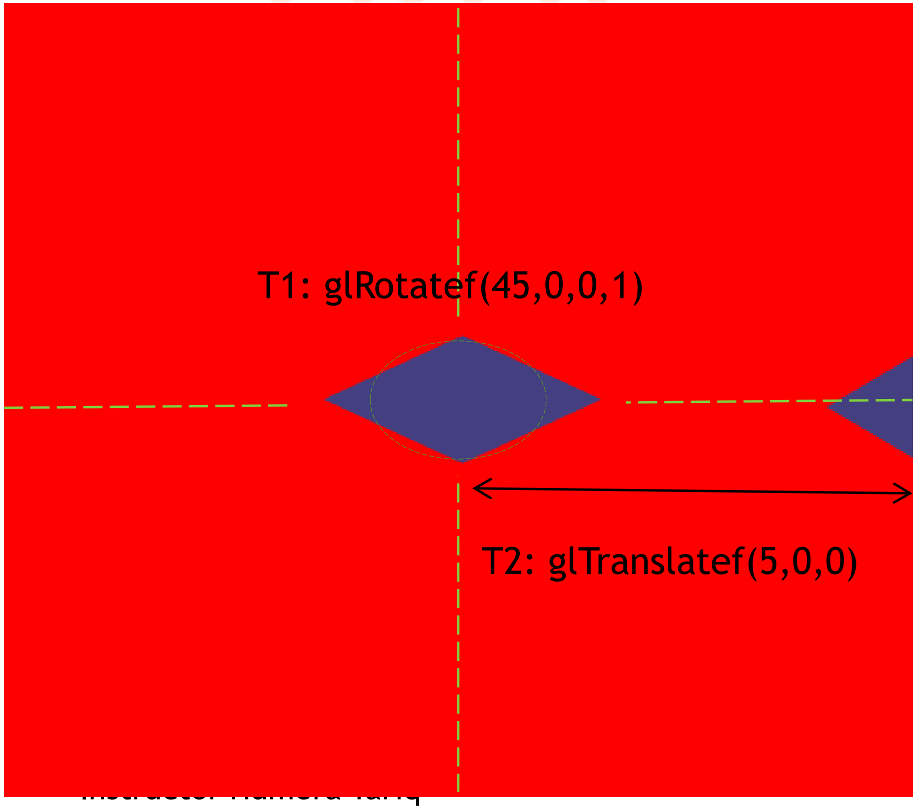


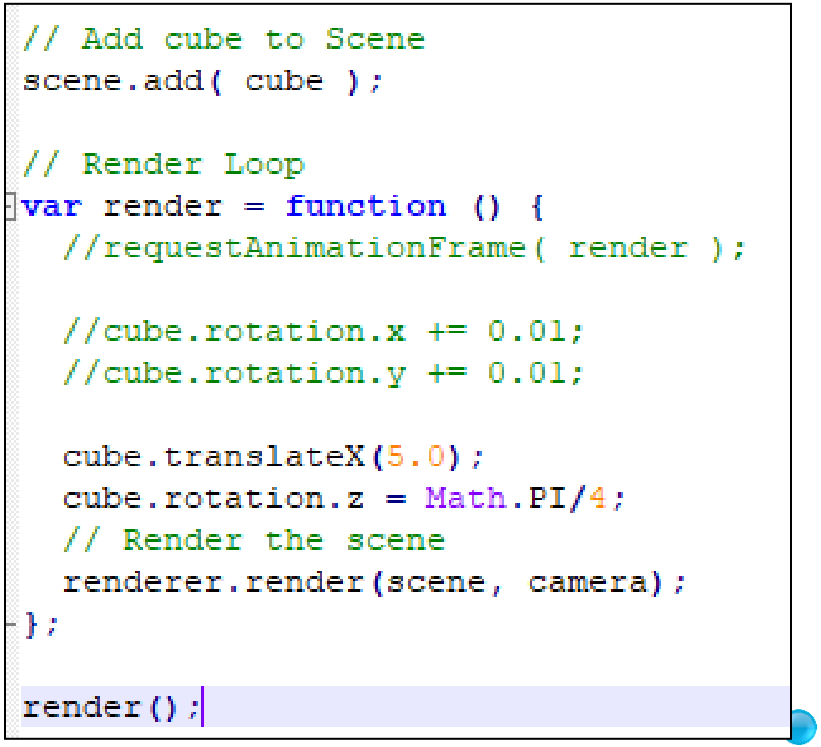
Object Transformation R + T (Translation followed by Rotation)

Composite Matrix = Rotation Matrix * Translation Matrix

M = R *T vs. M = T * R

M = T2 * T1 = R * T // object transformation (Read Right to Left)





```
// Add cube to Scene
scene.add( cube );

// Render Loop
var render = function () {
  //requestAnimationFrame( render );

  //cube.rotation.x += 0.01;
  //cube.rotation.y += 0.01;

  cube.translateX(5.0);
  cube.rotation.z = Math.PI/4;
  // Render the scene
  renderer.render(scene, camera);
};

render();
```

Practice Codes from Chap 05 Transformation of Objects

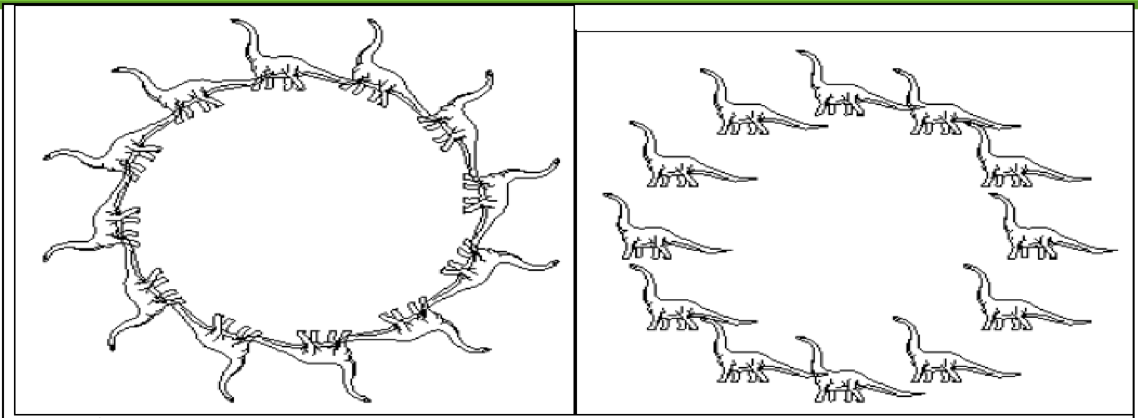
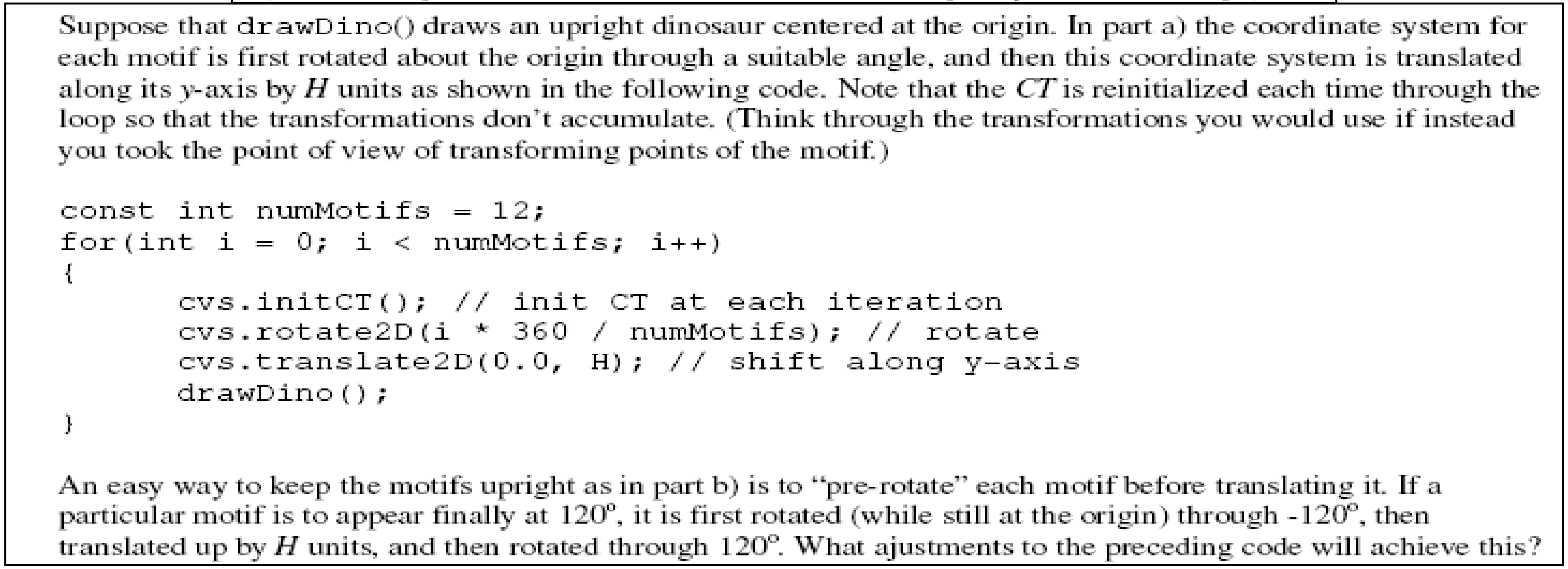


Figure 5.42. Two patterns based on a motif. a). each motif is rotated separately. b). all motifs are upright.



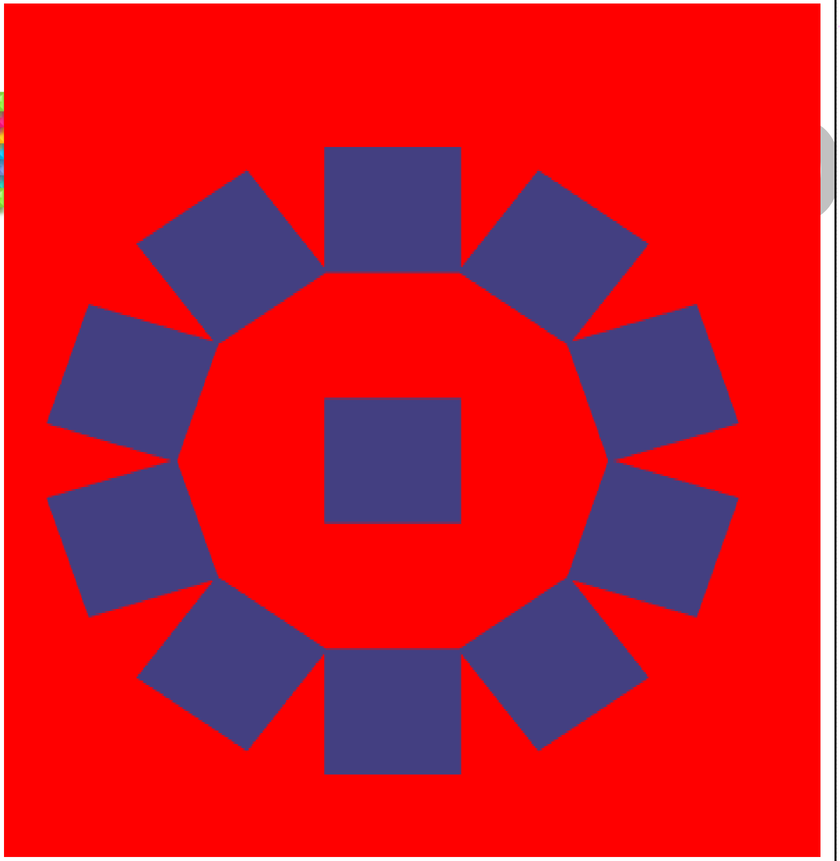
Suppose that drawDino() draws an upright dinosaur centered at the origin. In part a) the coordinate system for each motif is first rotated about the origin through a suitable angle, and then this coordinate system is translated along its *y*-axis by *H* units as shown in the following code. Note that the *CT* is reinitialized each time through the loop so that the transformations don't accumulate. (Think through the transformations you would use if instead you took the point of view of transforming points of the motif.)

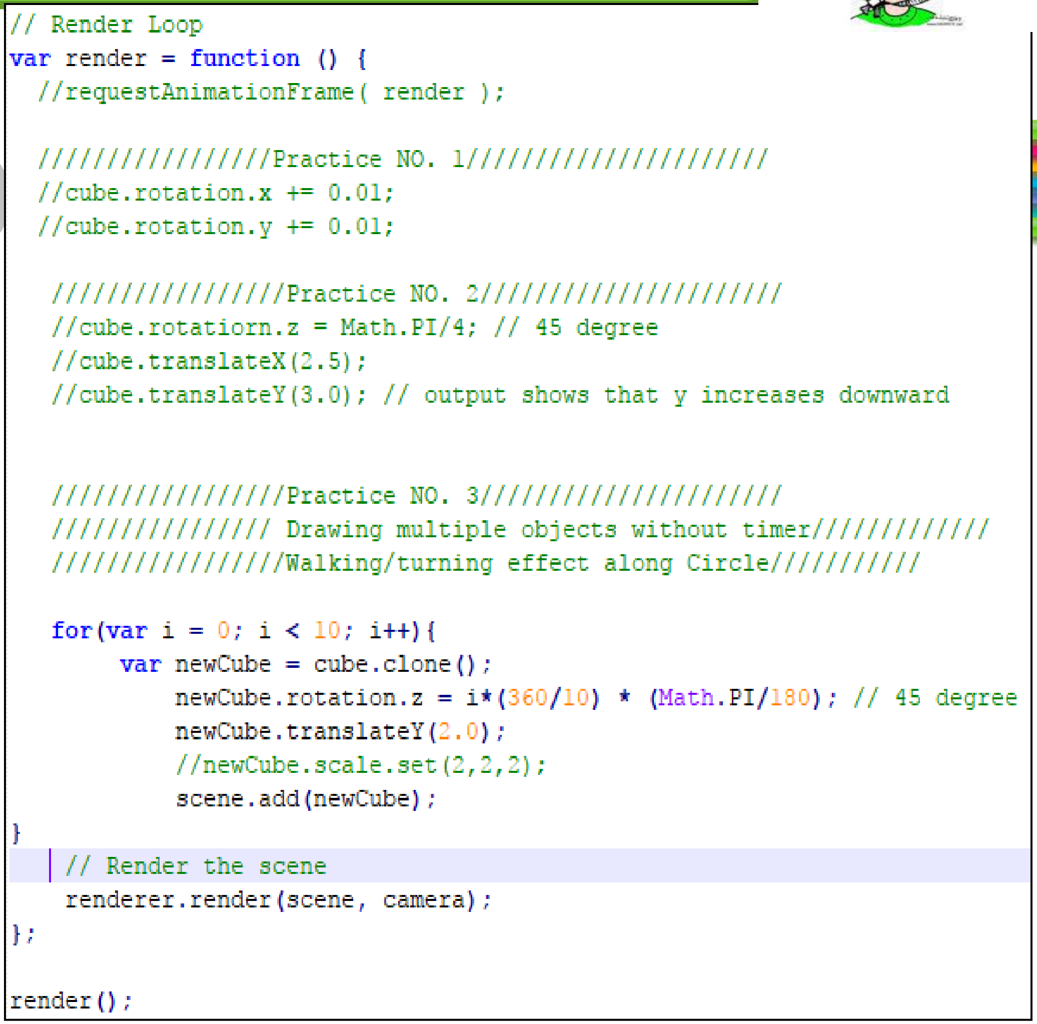
```
const int numMotifs = 12;
for(int i = 0; i < numMotifs; i++)
{
      cvs.initCT(); // init CT at each iteration
      cvs.rotate2D(i * 360 / numMotifs); // rotate
      cvs.translate2D(0.0, H); // shift along y-axis
      drawDino();
}
```

An easy way to keep the motifs upright as in part b) is to "pre-rotate" each motif before translating it. If a particular motif is to appear finally at 120°, it is first rotated (while still at the origin) through -120°, then translated up by *H* units, and then rotated through 120°. What ajustments to the preceding code will achieve this?

Code and Output of First Dino Pattern

Figure 5.42. Two patterns based on a motif. a). each motif is rotated separately





```
// Render Loop
var render = function () {
  //requestAnimationFrame( render );

  /////////////////Practice NO. 1//////////////////////
  //cube.rotation.x += 0.01;
  //cube.rotation.y += 0.01;

   /////////////////Practice NO. 2//////////////////////
   //cube.rotatiorn.z = Math.PI/4; // 45 degree
   //cube.translateX(2.5);
   //cube.translateY(3.0); // output shows that y increases downward


   /////////////////Practice NO. 3//////////////////////
   //////////////// Drawing multiple objects without timer/////////////
   /////////////////Walking/turning effect along Circle//////////

   for(var i = 0; i < 10; i++){
        var newCube = cube.clone();
            newCube.rotation.z = i*(360/10) * (Math.PI/180); // 45 degree
            newCube.translateY(2.0);
            //newCube.scale.set(2,2,2);
            scene.add(newCube);
}
    // Render the scene
    renderer.render(scene, camera);
};

render();
```