1)

npm init

2.

Ad d this lines to package.json and enter npm install

*"dependencies"*: {

*"@angular/common"*: "^7.1.0",

*"@angular/compiler"*: "^7.1.0",

*"@angular/core"*: "^7.1.0",

*"@angular/forms"*: "^7.1.0",

*"@angular/http"*: "^7.1.0",

*"@angular/platform-browser"*: "^7.1.0",

*"@angular/platform-browser-dynamic"*: "^7.1.0",

*"@angular/router"*: "^7.1.0",

*"core-js"*: "^2.5.7",

*"rxjs"*: "^6.3.3",

*"zone.js"*: "^0.8.26"

},

*"devDependencies"*: {

*"@types/node"*: "^10.12.10",

*"angular2-template-loader"*: "^0.6.2",

*"awesome-typescript-loader"*: "^5.2.1",

*"css-loader"*: "^1.0.1",

*"extract-text-webpack-plugin"*: "^3.0.2",

*"file-loader"*: "^2.0.0",

*"html-webpack-plugin"*: "^3.2.0",

*"node-sass"*: "^4.10.0",

*"raw-loader"*: "^0.5.1",

*"sass-loader"*: "^7.1.0",

*"style-loader"*: "^0.23.1",

*"tslint"*: "^5.11.0",

*"typescript"*: "^3.1.6",

*"url-loader"*: "^1.1.2",

*"webpack"*: "^4.26.0",

*"webpack-dev-server"*: "^3.1.10"

}

3. tsc –init will create tslint.js file and add below lines to it

{

*"compilerOptions"*: {

*"target"*: "es5",

*"module"*: "commonjs",

*"moduleResolution"*: "node",

*"sourceMap"*: true,

*"emitDecoratorMetadata"*: true,

*"experimentalDecorators"*: true,

*"lib"*: [ "es2015", "dom" ],

*"noImplicitAny"*: true,

*"suppressImplicitAnyIndexErrors"*: true,

*"typeRoots"*: [

"node\_modules/@types/"

]

},

*"compileOnSave"*: true,

*"exclude"*: [

"node\_modules/\*",

"\*\*/\*-aot.ts"

]

}

**4.Add webpack config file**

var webpack = require('webpack');

var path = require('path');

var HtmlWebpackPlugin = require('html-webpack-plugin');

var ExtractTextPlugin = require('extract-text-webpack-plugin');

module.exports = {

entry: {

polyfills: './src/polyfills.ts',

vendor: './src/vendor.ts',

app: './src/main.ts'

},

output: {

path: path.resolve( \_\_dirname + '/dist'),

publicPath: '/',

filename: '[name].[hash].js',

chunkFilename: '[id].[hash].chunk.js'

},

resolve: {

extensions: ['.ts', '.js']

},

module: {

rules: [

{

test: /\.ts$/,

loaders: [

{

loader: 'awesome-typescript-loader',

options: { configFileName: path.resolve(\_\_dirname + '/tsconfig.json') }

} , 'angular2-template-loader'

]

},

{

test: /\.html$/,

loader: 'html-loader'

},

{

test: /\.css$/,

loader: 'raw-loader'

},

]

},

plugins: [

new HtmlWebpackPlugin({

template: 'src/index.html'

})

]

}

**Step-5 Create src Directory**

**5.1 index.html**

<!DOCTYPE html>

<html>

<head>

<base href="/">

<title>Angular With Webpack</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

</head>

<body>

<my-app>Loading...</my-app>

</body>

</html>

**5.2 polyfills.ts**

import 'core-js/es6';

import 'core-js/es7/reflect';

require('zone.js/dist/zone');

**5.3 vendor.ts**

import '@angular/platform-browser';

import '@angular/platform-browser-dynamic';

import '@angular/core';

**5.4 main.ts**

import { platformBrowserDynamic } from '@angular/platform-browser-dynamic';

import { enableProdMode } from '@angular/core';

import { AppModule } from './app/app.module';

platformBrowserDynamic().bootstrapModule(AppModule);

**Step-6 App Directory**

**6.1 app.module.ts**

import { NgModule } from '@angular/core';

import { BrowserModule } from '@angular/platform-browser';

import { AppComponent } from './app.component';

@NgModule({

imports: [

BrowserModule

],

declarations: [

AppComponent

],

bootstrap: [ AppComponent ]

})

export class AppModule { }

**6.2 app.component.ts**

import { Component } from '@angular/core';

@Component({

selector: 'my-app',

templateUrl: './app.component.html',

styleUrls: ['./app.component.css']

})

export class AppComponent { }

**6.3 app.component.html**

<h1> Minimal Angular Application is Ready </h1>

<p class="header"> This is how it is supposed to look </p>

**Step-7 Build using webpack**

webpack