

Module 2: Modules and Components

Demo Document 1: Create a new Angular application

edureka!

edureka!

© Brain4ce Education Solutions Pvt. Ltd.

Create a new Angular Application:

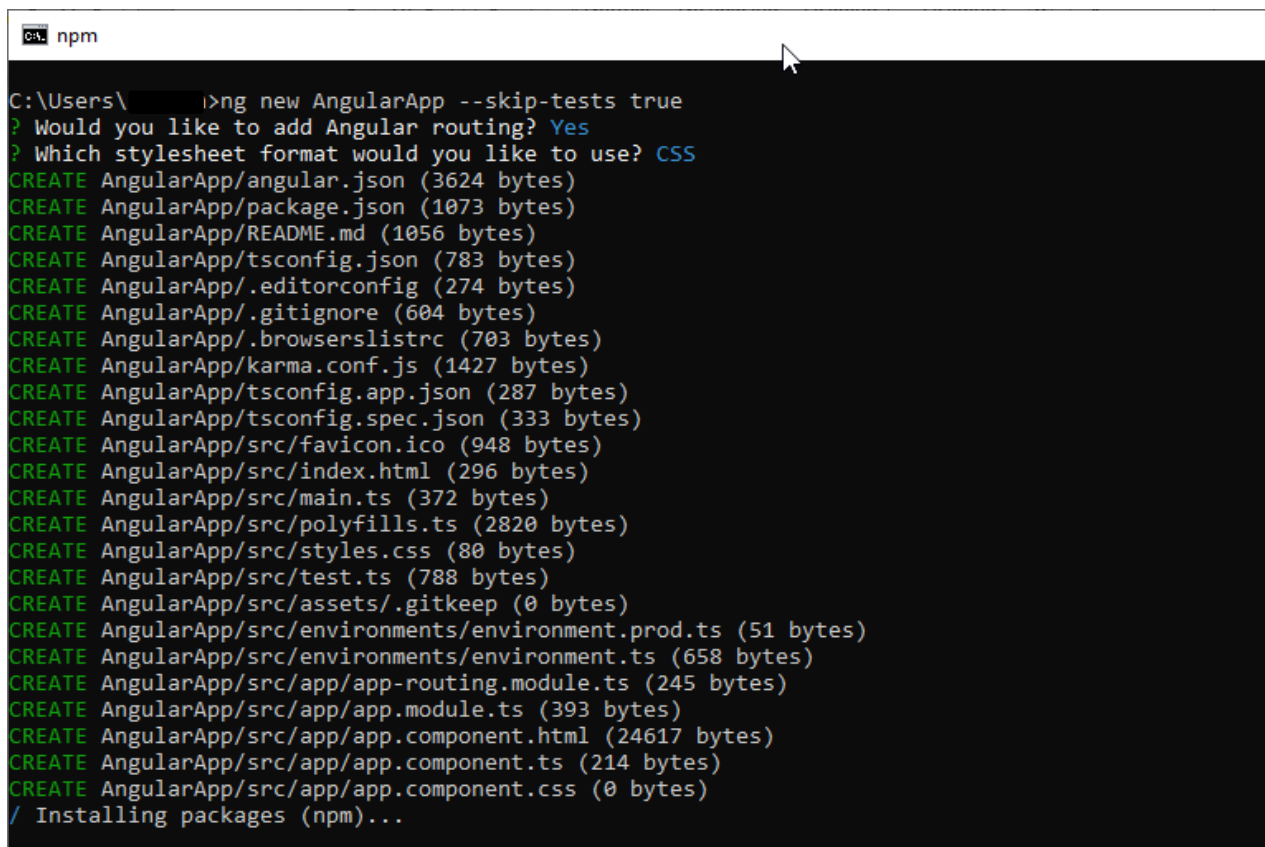
Step 1: Open the command prompt and check the latest version of NodeJS **node -v**

Step 2: Install the latest version of Angular CLI by **npm install -g @angular/cli**

Step 3:

Type the command **ng new 'application_name'** to create a new Angular application

Select the routing and the stylesheet format **CSS**



```
npm
C:\Users\>ng new AngularApp --skip-tests true
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use? CSS
CREATE AngularApp/angular.json (3624 bytes)
CREATE AngularApp/package.json (1073 bytes)
CREATE AngularApp/README.md (1056 bytes)
CREATE AngularApp/tsconfig.json (783 bytes)
CREATE AngularApp/.editorconfig (274 bytes)
CREATE AngularApp/.gitignore (604 bytes)
CREATE AngularApp/.browserslistrc (703 bytes)
CREATE AngularApp/karma.conf.js (1427 bytes)
CREATE AngularApp/tsconfig.app.json (287 bytes)
CREATE AngularApp/tsconfig.spec.json (333 bytes)
CREATE AngularApp/src/favicon.ico (948 bytes)
CREATE AngularApp/src/index.html (296 bytes)
CREATE AngularApp/src/main.ts (372 bytes)
CREATE AngularApp/src/polyfills.ts (2820 bytes)
CREATE AngularApp/src/styles.css (80 bytes)
CREATE AngularApp/src/test.ts (788 bytes)
CREATE AngularApp/src/assets/.gitkeep (0 bytes)
CREATE AngularApp/src/environments/environment.prod.ts (51 bytes)
CREATE AngularApp/src/environments/environment.ts (658 bytes)
CREATE AngularApp/src/app/app-routing.module.ts (245 bytes)
CREATE AngularApp/src/app/app.module.ts (393 bytes)
CREATE AngularApp/src/app/app.component.html (24617 bytes)
CREATE AngularApp/src/app/app.component.ts (214 bytes)
CREATE AngularApp/src/app/app.component.css (0 bytes)
/ Installing packages (npm)...
```

Note: It will take a while to create the application

Step 4: Once 'Successfully initialized git' message is prompted, move to the application directory

```
Command Prompt

The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/app/app.component.html.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/app/app.component.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/app/app.module.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/environments/environment.prod.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/environments/environment.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/index.html.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/main.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/polyfills.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/styles.css.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/test.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in tsconfig.app.json.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in tsconfig.json.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in tsconfig.spec.json.
The file will have its original line endings in your working directory
Successfully initialized git.

C:\Users\>
```

Step 5:

Move to the application directory by typing `cd 'application_name'`

```
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/polyfills.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/styles.css.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in src/test.ts.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in tsconfig.app.json.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in tsconfig.json.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in tsconfig.spec.json.
The file will have its original line endings in your working directory
Successfully initialized git.

C:\Users\>cd AngularApp

C:\Users\AngularApp>
```

Step 6:

Type the command “code .” to open the application in visual studio code

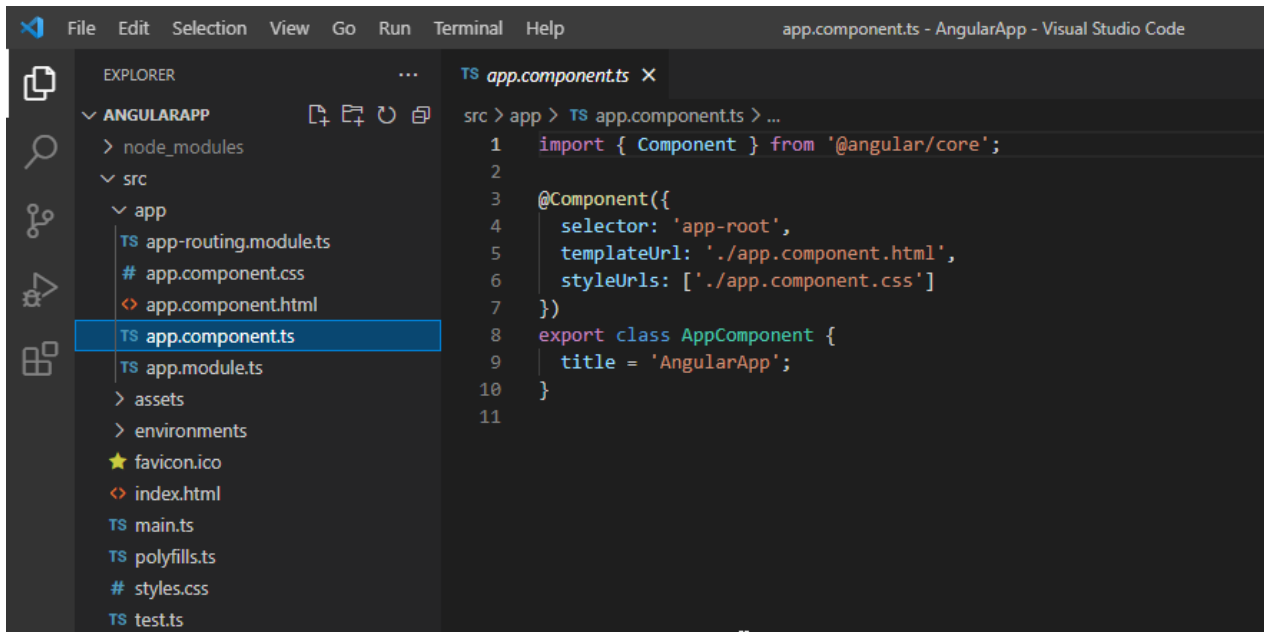
```
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in tsconfig.spec.json.
The file will have its original line endings in your working directory
Successfully initialized git.

C:\Users\>cd AngularApp

C:\Users\AngularApp>code .

C:\Users\AngularApp>
```

Step 7:

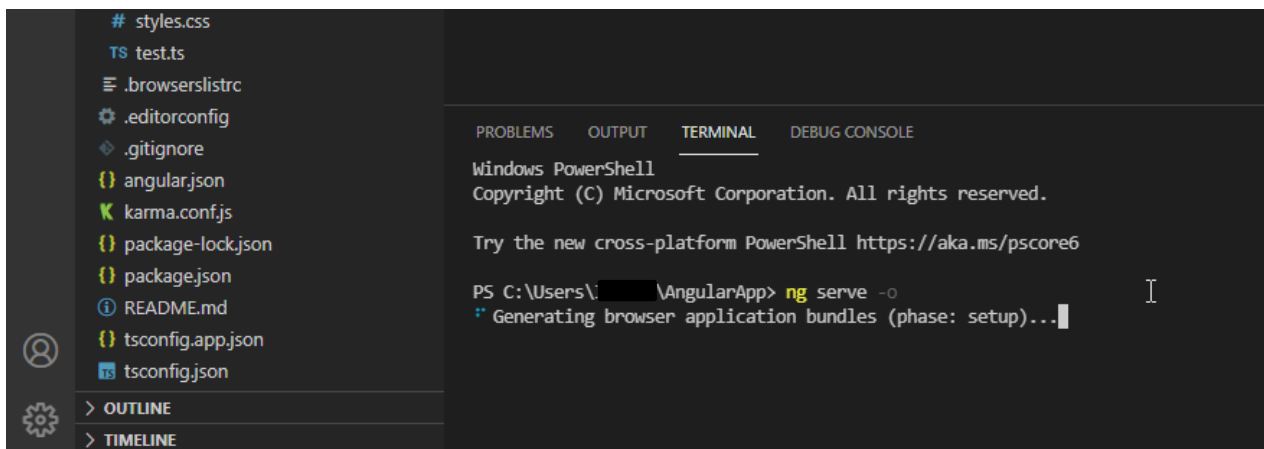


The screenshot shows the Visual Studio Code interface. On the left, the Explorer sidebar displays the project structure for 'ANGULARAPP'. The 'src' directory is expanded, showing 'app' and 'assets'. The 'app' directory is further expanded, listing 'app-routing.module.ts', 'app.component.css', 'app.component.html', 'app.component.ts' (selected), 'app.module.ts', and 'test.ts'. The main editor area shows the content of 'app.component.ts'.

```
src > app > TS app.component.ts > ...
1  import { Component } from '@angular/core';
2
3  @Component({
4    selector: 'app-root',
5    templateUrl: './app.component.html',
6    styleUrls: ['./app.component.css']
7  })
8  export class AppComponent {
9    title = 'AngularApp';
10 }
11
```

Step 8:

Launch the application by typing 'ng serve -o'

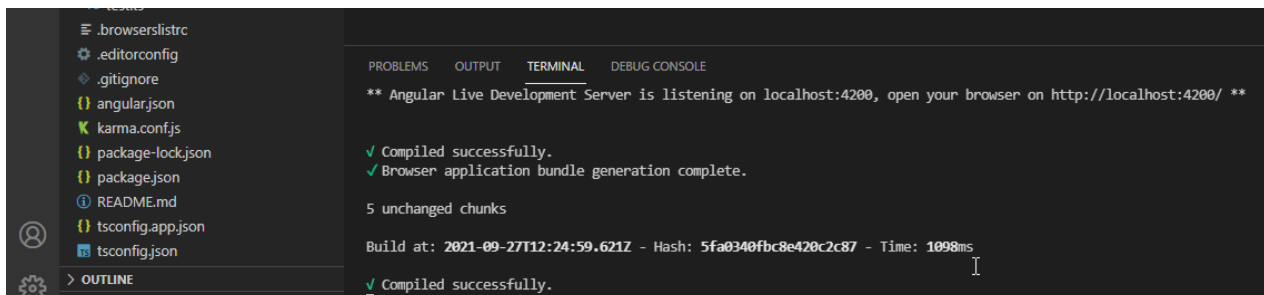


The screenshot shows the Visual Studio Code interface with the Explorer sidebar on the left. The 'src' directory is expanded, showing 'app' and 'assets'. The 'app' directory is further expanded, listing 'app-routing.module.ts', 'app.component.css', 'app.component.html', 'app.component.ts' (selected), 'app.module.ts', and 'test.ts'. The main editor area shows the content of 'app.component.ts'. The terminal window at the bottom shows the command 'ng serve -o' being executed.

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\...\AngularApp> ng serve -o
* Generating browser application bundles (phase: setup)...
```

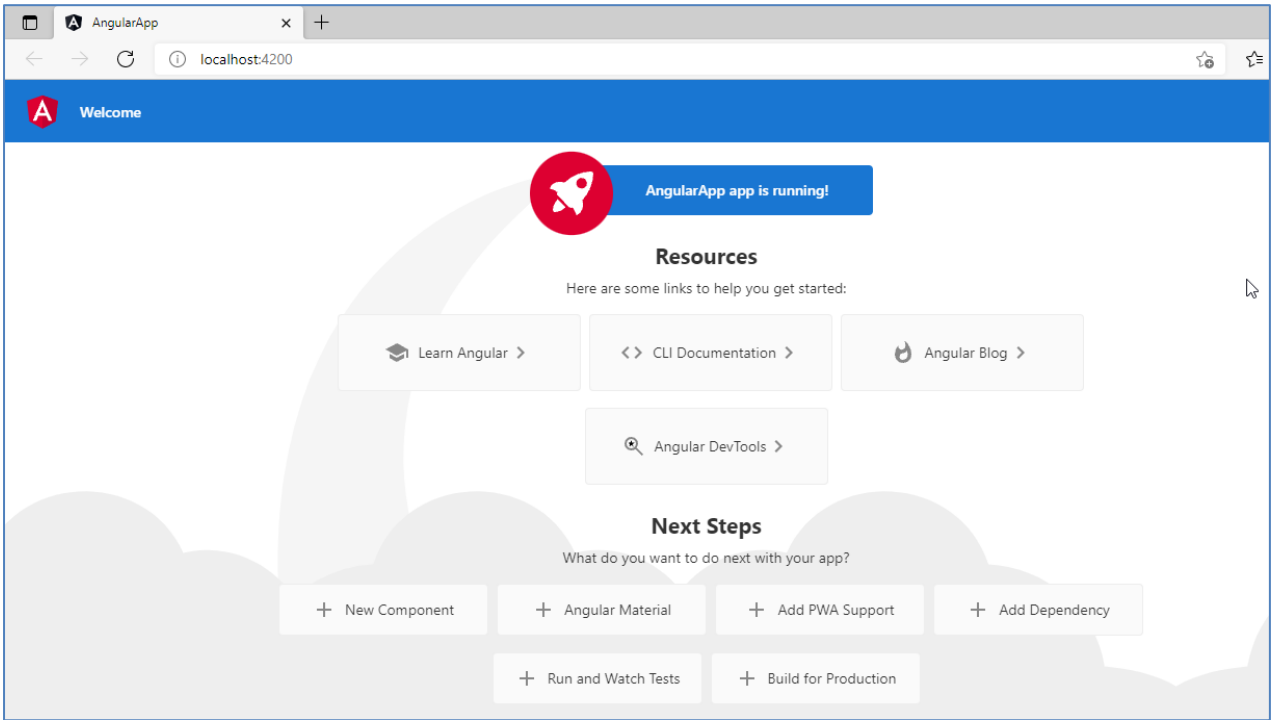


The screenshot shows the Visual Studio Code interface with the Explorer sidebar on the left. The 'src' directory is expanded, showing 'app' and 'assets'. The 'app' directory is further expanded, listing 'app-routing.module.ts', 'app.component.css', 'app.component.html', 'app.component.ts' (selected), 'app.module.ts', and 'test.ts'. The main editor area shows the content of 'app.component.ts'. The terminal window at the bottom shows the output of the 'ng serve -o' command.

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE
** Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ **

✓ Compiled successfully.
✓ Browser application bundle generation complete.
5 unchanged chunks

Build at: 2021-09-27T12:24:59.621Z - Hash: 5fa0340fbc8e420c2c87 - Time: 1098ms
✓ Compiled successfully.
```



edureka!