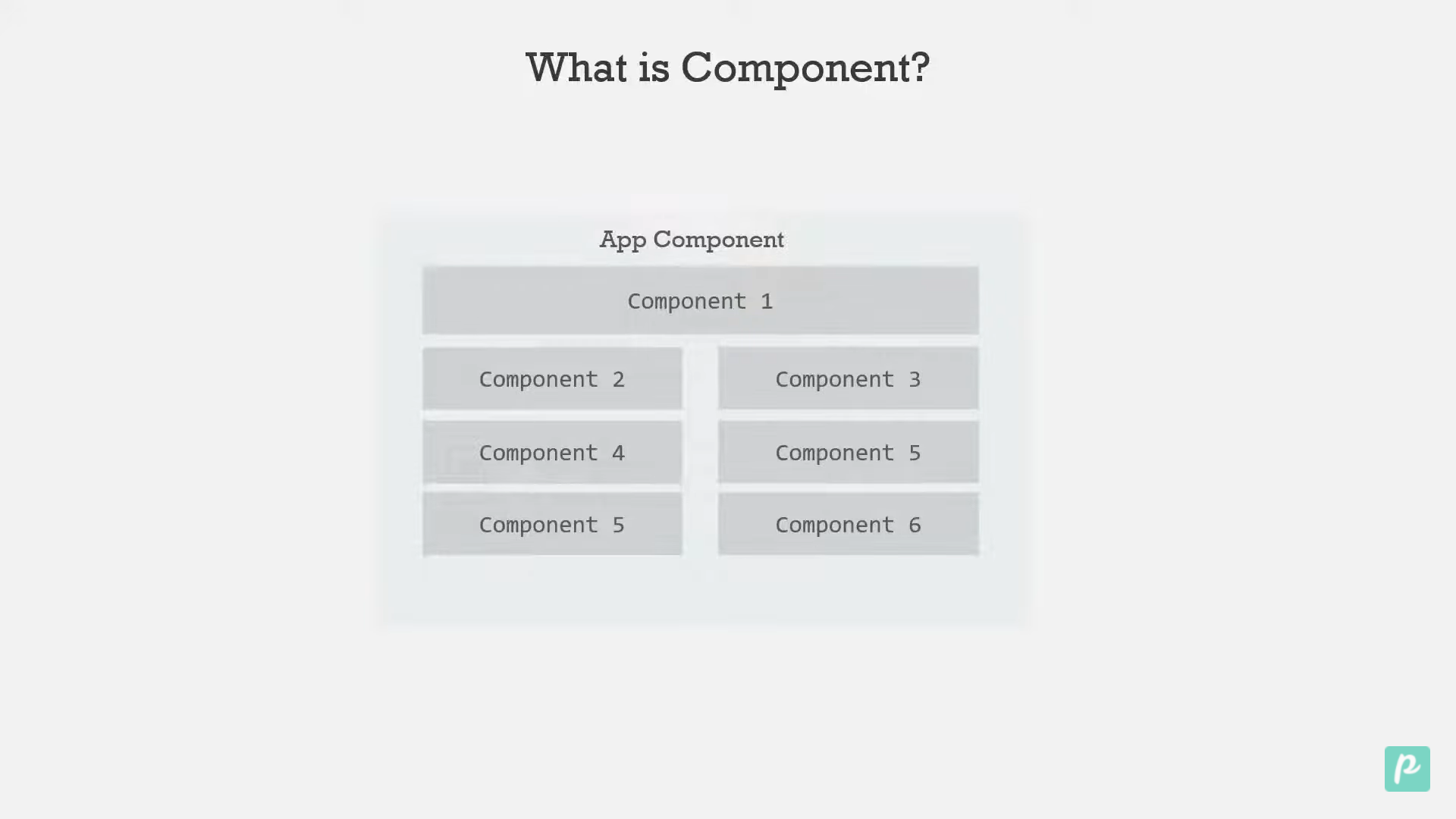
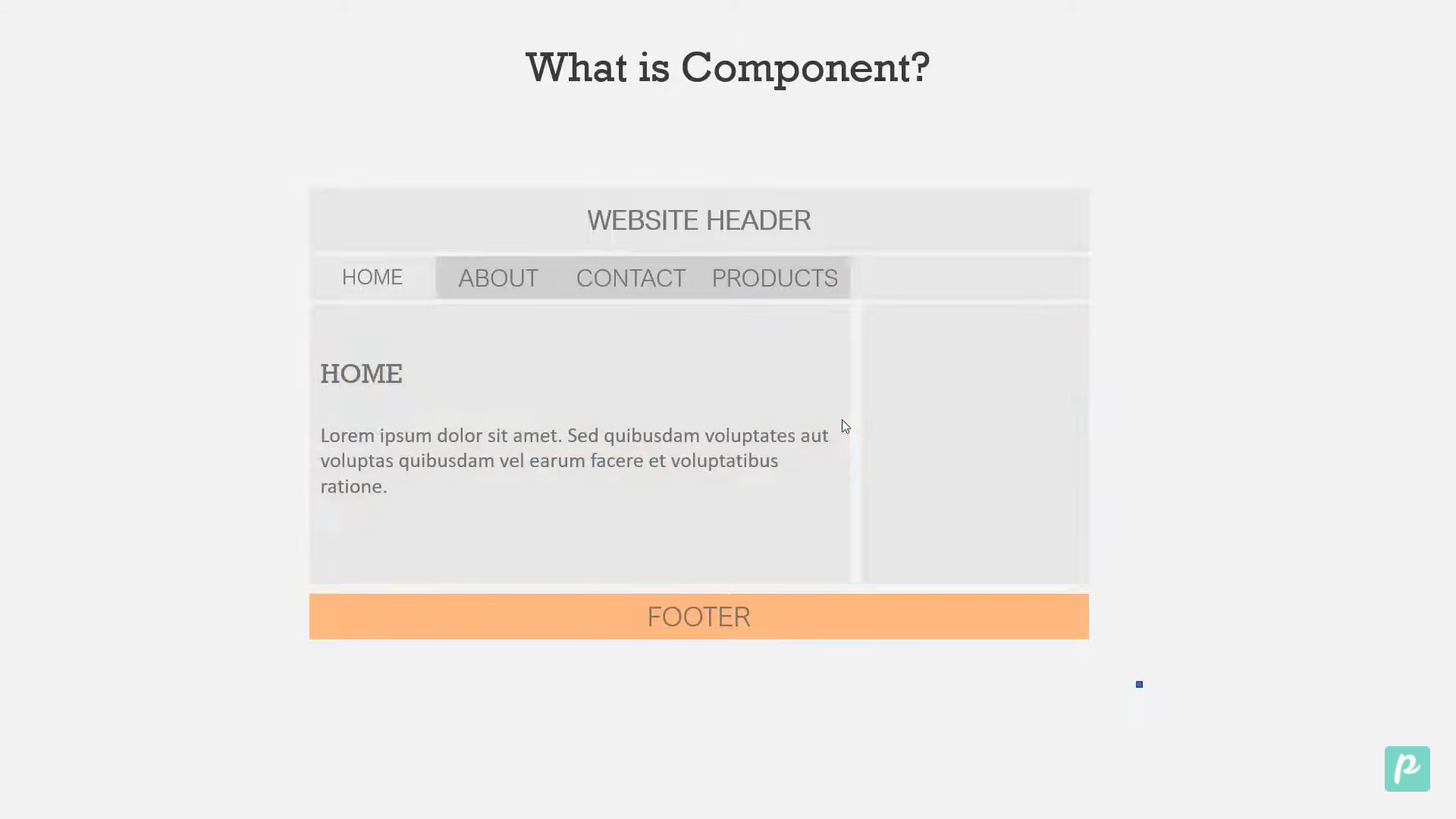
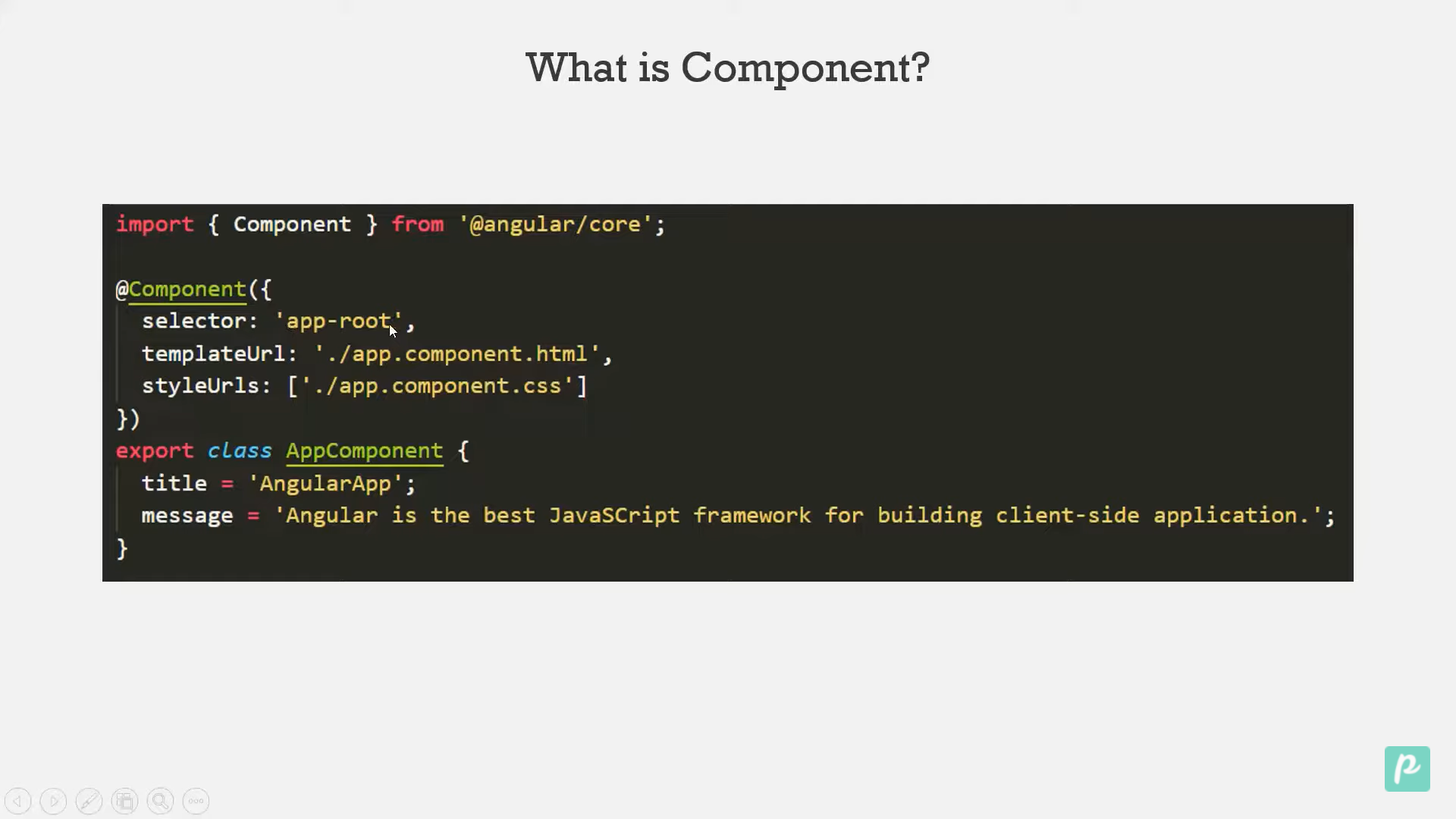
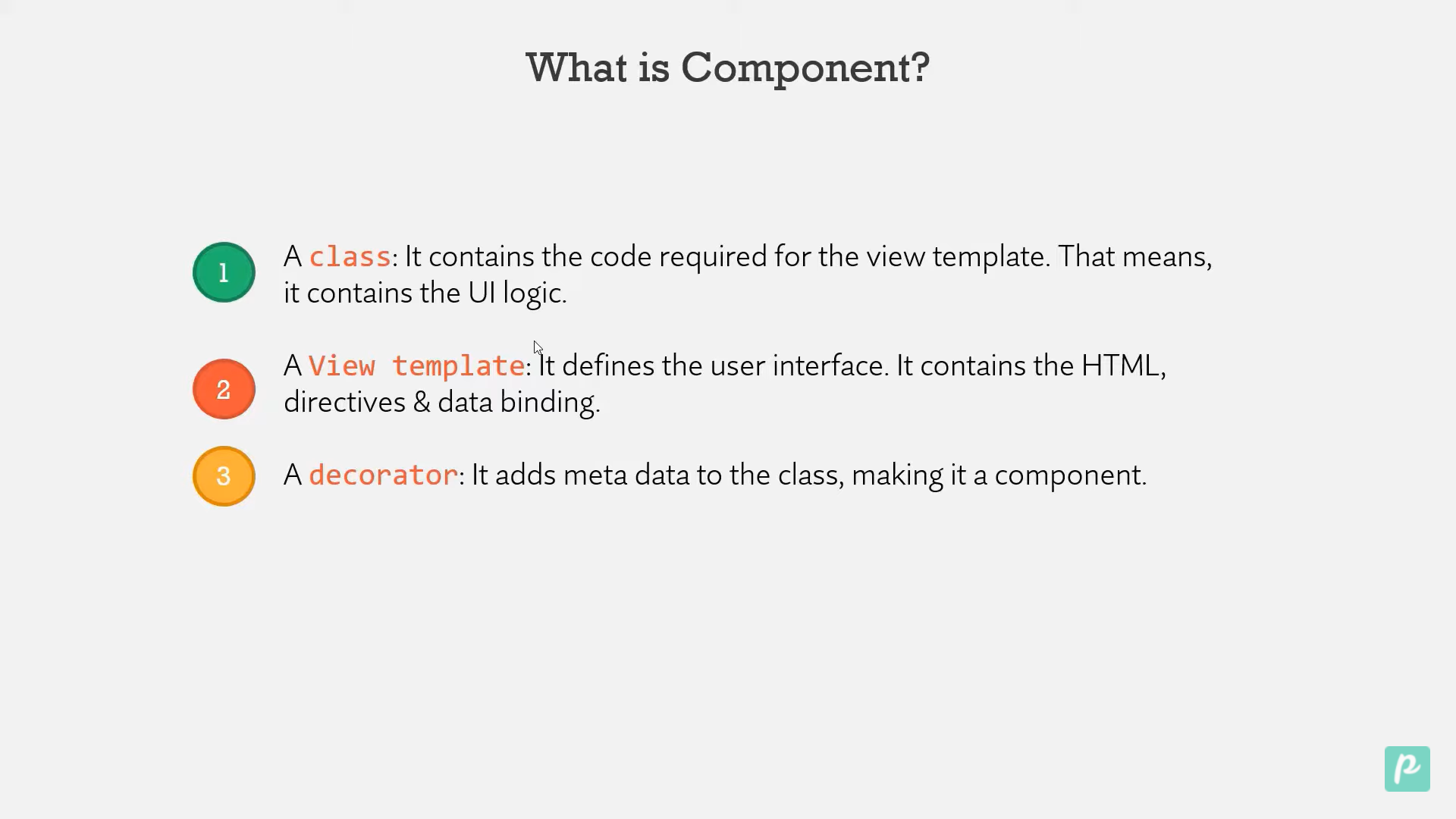


Main component is app component or root component.

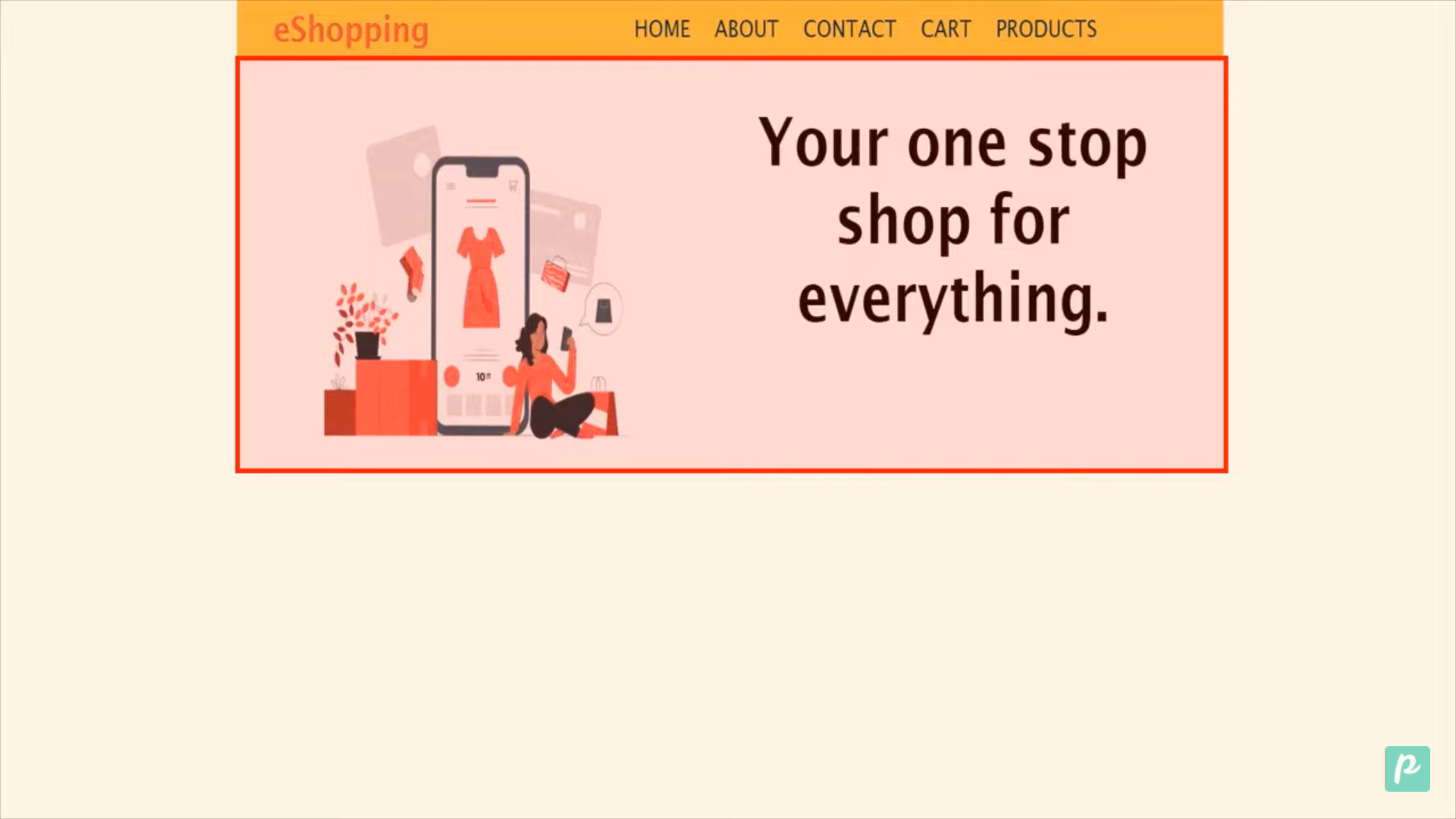






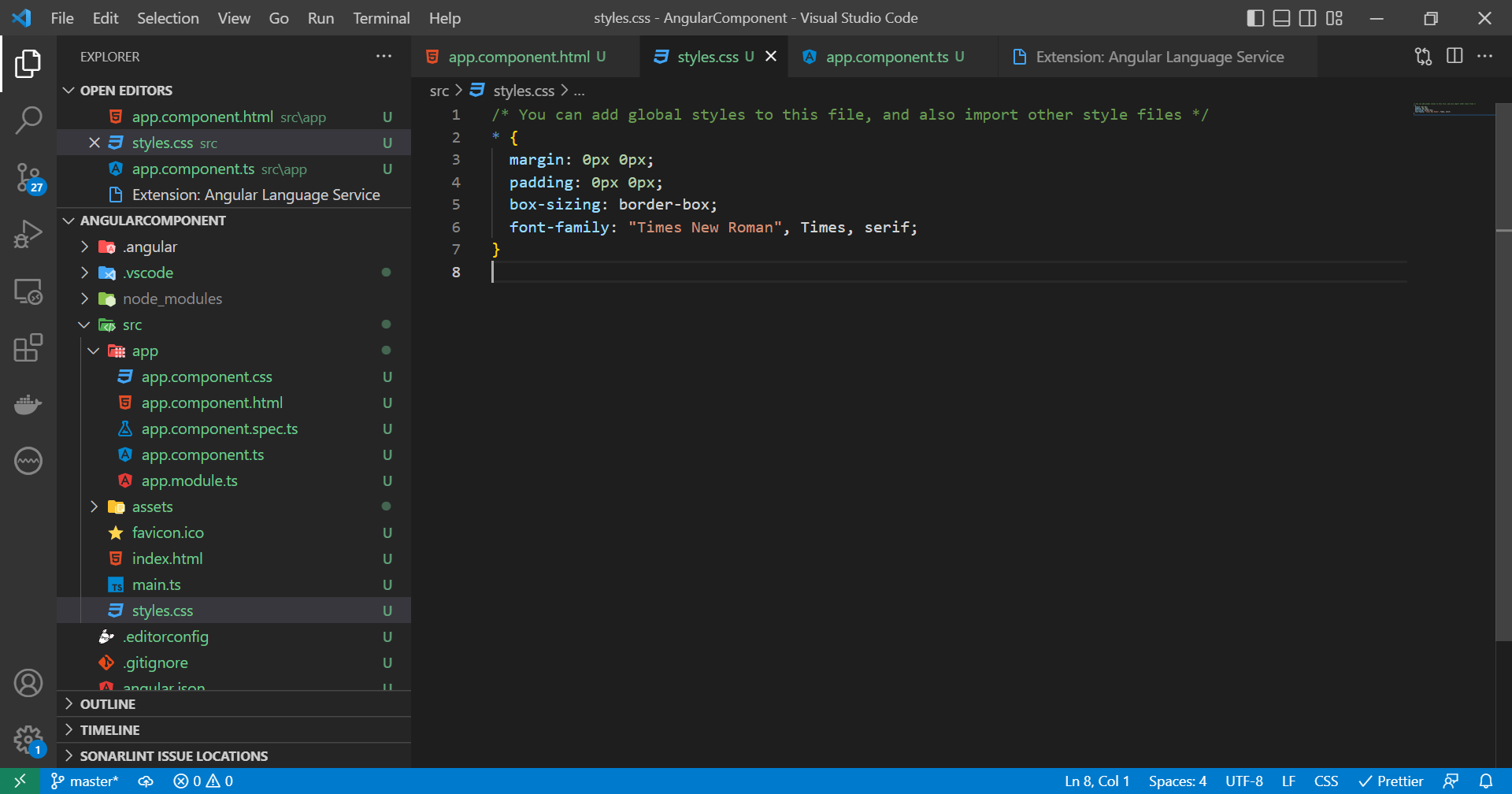


Creating the component



Apply global stylesheet

Step1: We are applying the style to every section of the page.



Step2 : Inside the app folder create one more folder Container. We will create all the component inside this. Inside the component create

1.container.component.ts

export class ContainerComponent{

}

2. we have to import the component and use annonation @component, and we have to provide the selector,templateurl and styleurls

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

@Component({

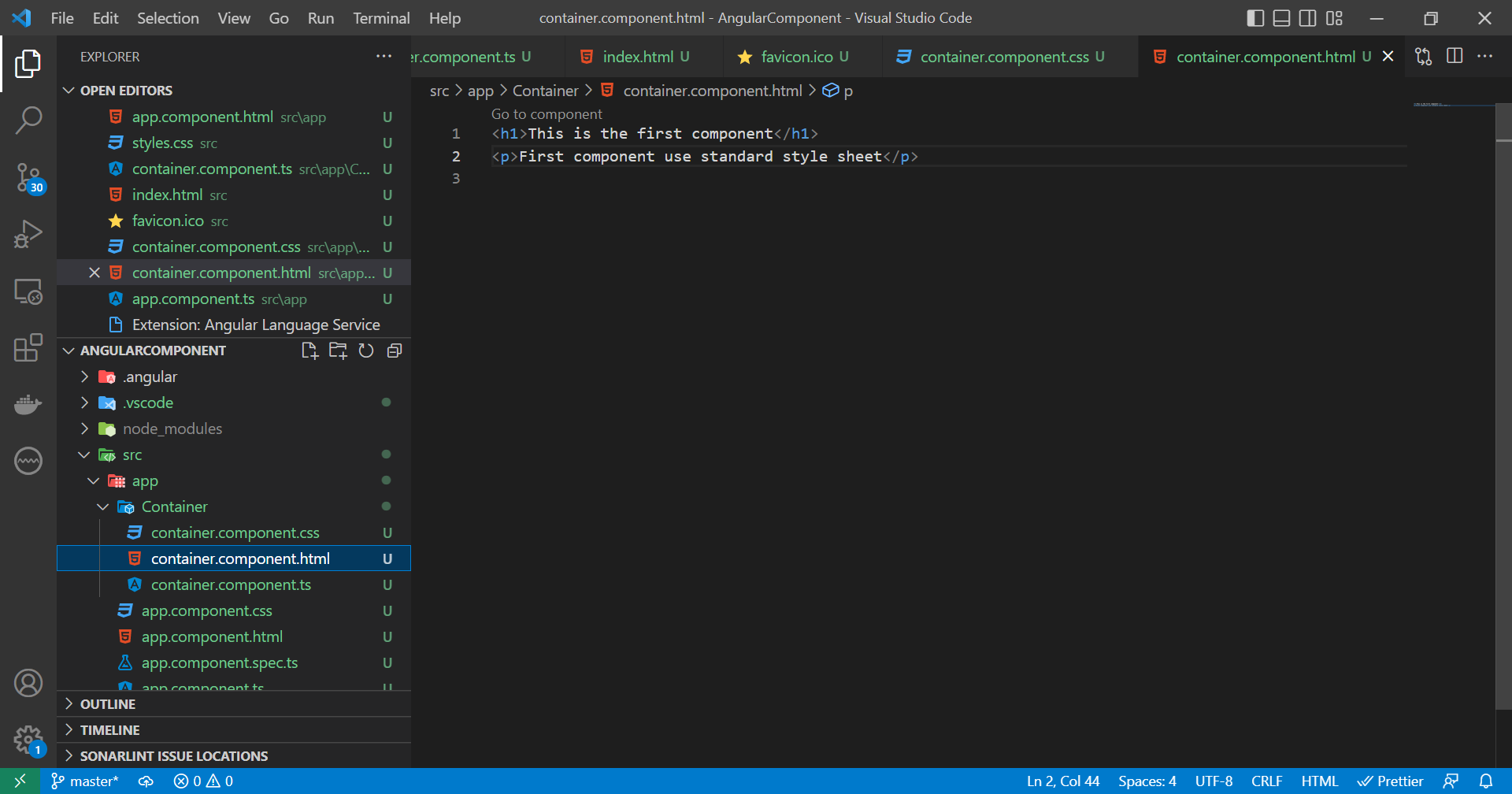
  selector: '',

  templateUrl: '',

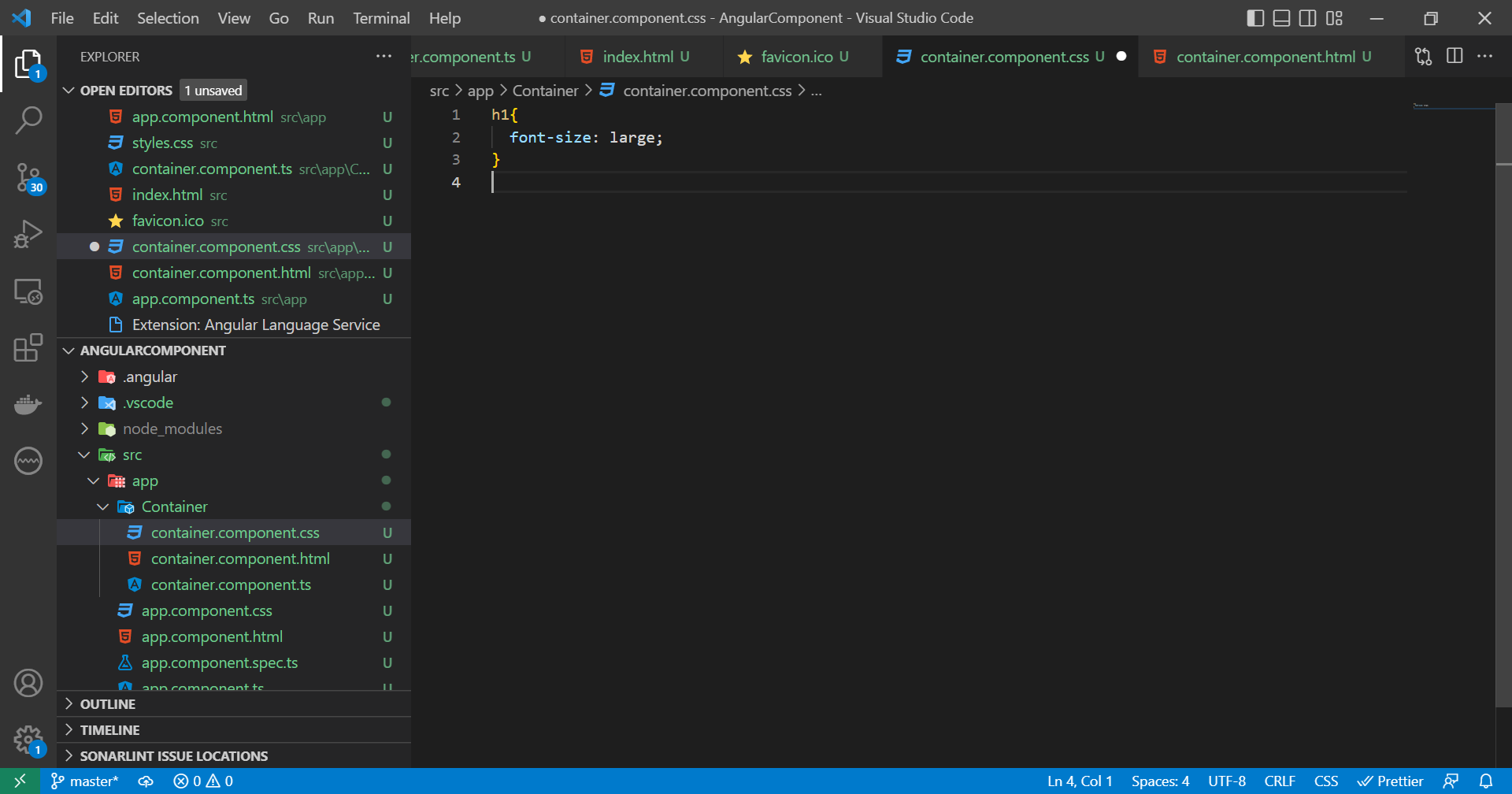
  styleUrls: [''],

})

Step3 : Now create a file container.component.html and write the code



Step4: Create style sheet file container.component.css



Step 5: register the component in app.module.ts . In decleration write your component name.

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

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

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

import { ContainerComponent } from './Container/container.component';

@NgModule({

  declarations: [AppComponent, ContainerComponent],

  imports: [BrowserModule],

  providers: [],

  bootstrap: [AppComponent],

})

export class AppModule {}

Step 6: use this component in app.component.html

<h1>Welcome</h1>

<p>{{ title }}</p>

<app-container></app-container>

We can generate component using command

ng generate component <nameof component>

ex ng generate component nav

Windows PowerShell

Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS E:\Angular\Angular\_Project\AngularComponent> ng generate component nav

We can use shortcut for generating the component: ng g c <name of component>

Ng g c header

Angular Template Property

To use Template property we will generate one more component notification.

If we don’t want to use html file instead of that we can use template property and the HTML code can be written inside that.

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

@Component({

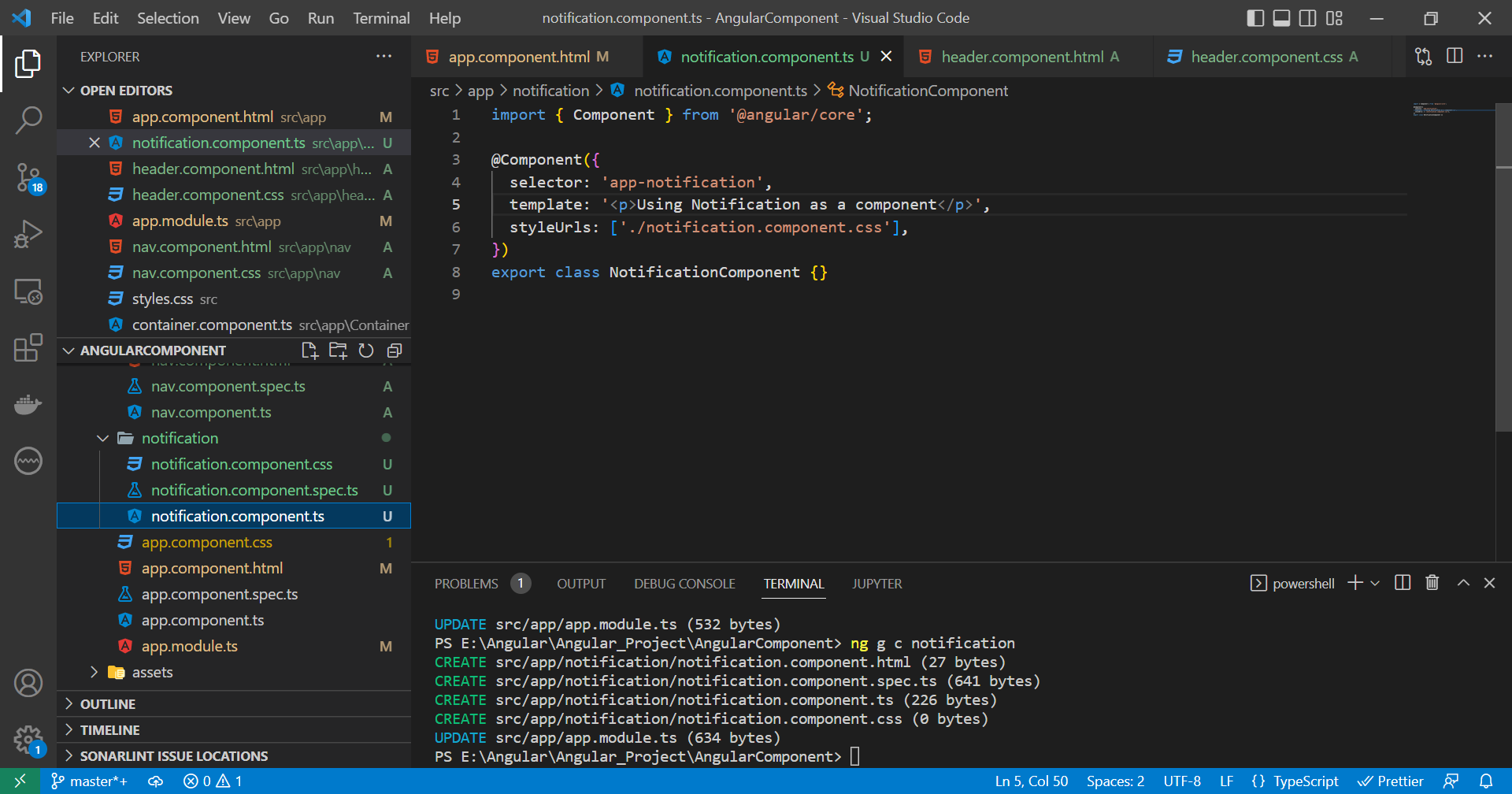
  selector: 'app-notification',

  template: '<p>Using Notification as a component</p>',

  styleUrls: ['./notification.component.css'],

})

export class NotificationComponent {}



If we have multiple lines of code then we can use ` back ticks

Styles Property: Instead of providing the external css file we can write inline css, styles:``

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

@Component({

  selector: 'app-notification',

  template: '<p>Using Notification as a component</p>',

  styles: [' p{ font-size: 18px; text-align: center;}',''],

})

export class NotificationComponent {}

BootStrap in Angular

Step 1: We have to install bootstrap and command is npm install - - save bootstrap. Since it is third party libraries file it will in node module folder.

Step 2: In angular.json file we have to add the path in style property.

 "styles": [

              "node\_modules/bootstrap/dist/css/bootstrap.min.css",

              "src/styles.css"

            ],

Save it .

Now we can use it in notification component

@Component({

  selector: 'app-notification',

  template: `<div class="alert alert-success">

    <p>This is notification</p>

  </div>`,

  styles: [' p{ font-size: 18px; text-align: center;}', ''],

})

export class NotificationComponent {}

in template we had use bootstrap.