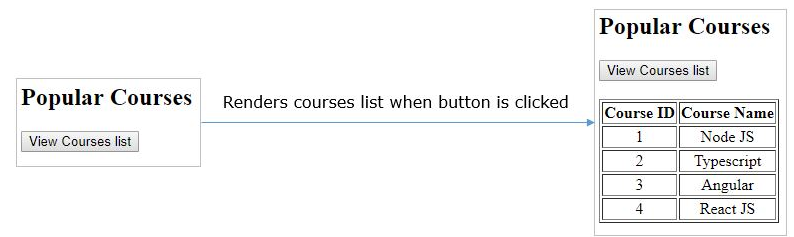
Highlights:

* Creating a nested component
* Loading nested component in the container component

**Demo Steps:**

**Problem Statement:**Loading CourseslistComponent in the root component when the user clicks on the **View courses list**button as shown below:



1. Create a component coursesList using the following CLI command
2. ng generate component coursesList

1. Above command will create a folder with name courses-list with the following files

* courses-list.component.ts
* courses-list.component.html
* courses-list.component.css
* courses-list.component.spec.ts

1. CoursesListComponent class will be added in **app.module.ts**file
2. import { BrowserModule } from '@angular/platform-browser';
3. import { NgModule } from '@angular/core';
4. import { AppComponent } from './app.component';
5. import { CoursesListComponent } from './courses-list/courses-list.component';
6. @NgModule({
7. declarations: [
8. AppComponent,
9. CoursesListComponent
10. ],
11. imports: [
12. BrowserModule
13. ],
14. providers: [],
15. bootstrap: [AppComponent]
16. })
17. export class AppModule { }

**Line 10:** **CoursesListComponent**is added to the declarations property to make it available to all other components in the module.

1. Write the below code in **courses-list.component.ts**
2. import { Component, OnInit } from '@angular/core';
3. @Component({
4. selector: 'app-courses-list',
5. templateUrl: './courses-list.component.html',
6. styleUrls: ['./courses-list.component.css']
7. })
8. export class CoursesListComponent {
9. courses = [
10. { courseId: 1, courseName: "Node JS" },
11. { courseId: 2, courseName: "Typescript" },
12. { courseId: 3, courseName: "Angular" },
13. { courseId: 4, courseName: "React JS" }
14. ];
15. }

**Line 11-14:** courses is an array of objects where each object has properties courseId and courseName

1. Write the below code in **courses-list.component.html**
2. <table border="1">
3. <thead>
4. <tr>
5. <th>Course ID</th>
6. <th>Course Name</th>
7. </tr>
8. </thead>
9. <tbody>
10. <tr \*ngFor="let course of courses">
11. <td>{{course.courseId}}</td>
12. <td>{{course.courseName}}</td>
13. </tr>
14. </tbody>
15. </table>

**Line 9-12:** ngFor iterates over courses array and renders **courseId**and **courseName**values

1. Add the following code in **courses-list.component.css**
2. tr{
3. text-align:center;
4. }

**Line 1-3:** adds center alignment to the table row in HTML page

1. Write the below code in **app.component.html**
2. <h2> Popular Courses </h2>
3. <button (click)="show=true">View Courses list</button><br/><br/>
4. <div \*ngIf="show">
5. <app-courses-list></app-courses-list>
6. </div>

**Line 2:** click event is bound to button which will initialize show property to true when it is clicked

**Line 4:**It loads CoursesList component only if the show property value is true.

**IN app.component.ts** initialize show to false

show:boolean=false

1. Save the files and check the output in the browser

Next, we will learn how to exchange data between parent and child components.