

Lesson 03 Demo 03

Creating Services for the Data in the Application

Objective: To create services for the data in the Application

Tool required: Visual Studio Code

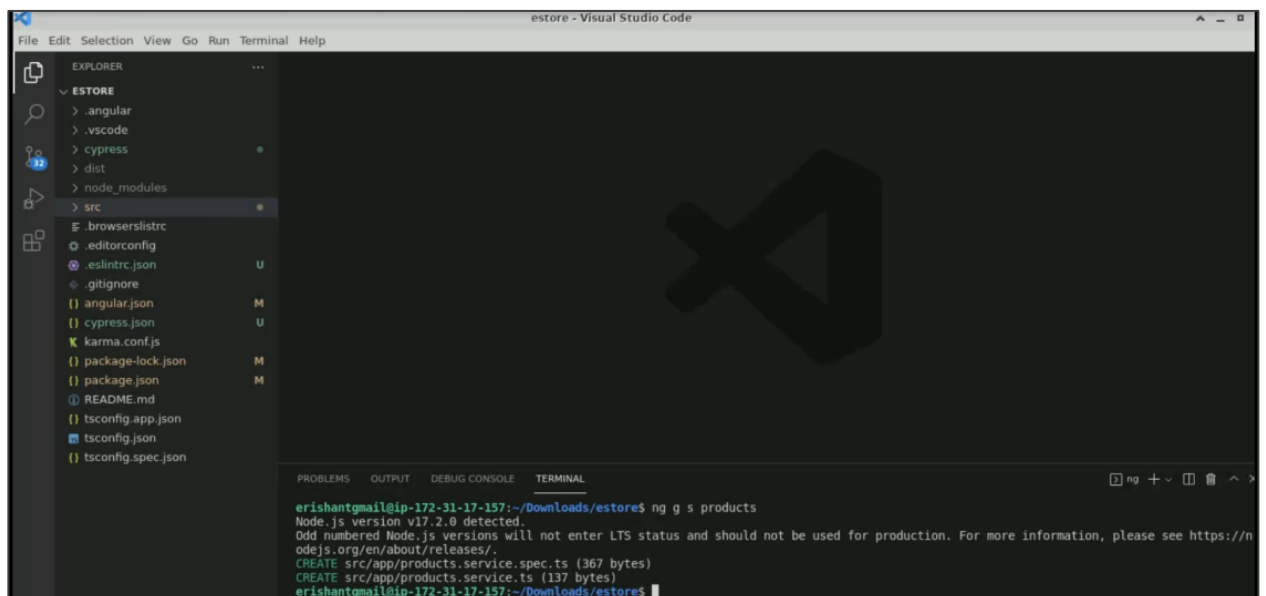
Prerequisites: None

Steps to be followed:

1. Create services for data in the application

Step 1: Create services for data in the application

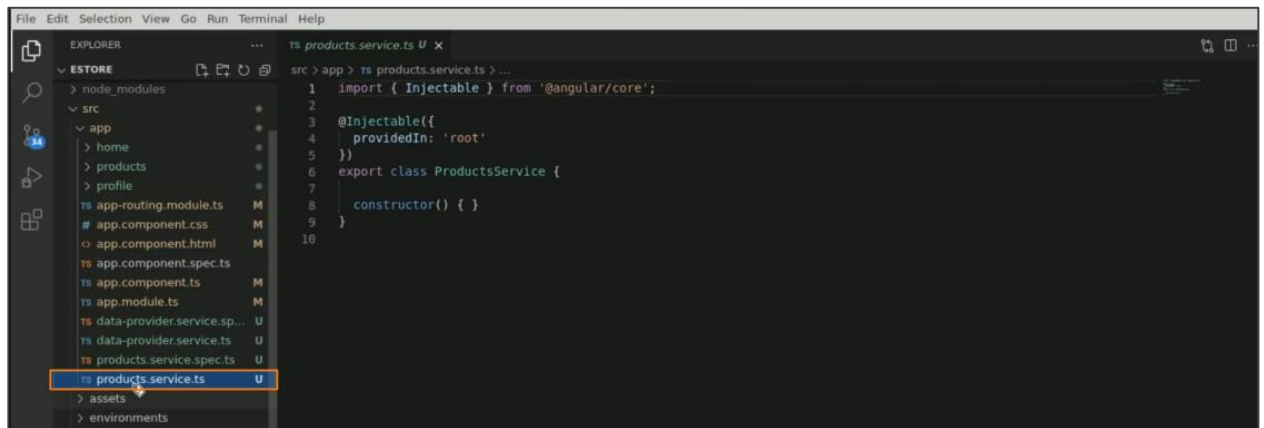
- 1.1 In the Visual Studio Code terminal, generate a service by typing the command **ng g s products**



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left displays the file structure of a project named 'estore'. The 'src' directory is expanded, showing various files including 'browserlistrc', 'editorconfig', 'eslinttrc.json', '.gitignore', 'angular.json', 'cypress.json', 'karma.conf.js', 'package-lock.json', 'package.json', 'README.md', 'tsconfig.app.json', 'tsconfig.json', and 'tsconfig.spec.json'. The Terminal panel at the bottom shows the command 'ng g s products' being executed. The output indicates that Node.js version v17.2.0 was detected and that the command successfully created two files: 'src/app/products.service.spec.ts' (367 bytes) and 'src/app/products.service.ts' (137 bytes).

```
erishantgmail@ip-172-31-17-157:~/Downloads/estore$ ng g s products
Node.js version v17.2.0 detected.
Odd numbered Node.js versions will not enter LTS status and should not be used for production. For more information, please see https://nodejs.org/en/about/releases/.
CREATE src/app/products.service.spec.ts (367 bytes)
CREATE src/app/products.service.ts (137 bytes)
erishantgmail@ip-172-31-17-157:~/Downloads/estore$
```

1.2 A product service **products.service.ts** has been generated in the source



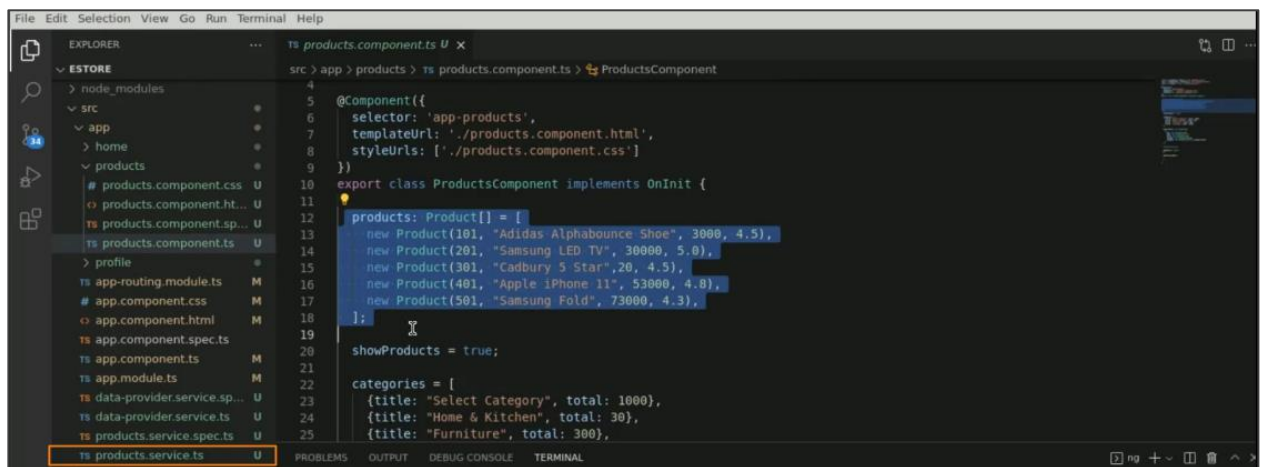
```

File Edit Selection View Go Run Terminal Help
EXPLORER
  > node_modules
  > src
    > app
      > home
      > products
      > profile
      > products.service.ts
    > app-routing.module.ts
    > app.component.css
    > app.component.html
    > app.component.spec.ts
    > app.component.ts
    > app.module.ts
    > data-provider.service.spec.ts
    > data-provider.service.ts
    > products.service.spec.ts
    > products.service.ts
  > assets
  > environments

src > app > ts products.service.ts
1 import { Injectable } from '@angular/core';
2
3 @Injectable({
4   providedIn: 'root'
5 })
6 export class ProductsService {
7
8   constructor() {}
9
10 }

```

1.3 Now, navigate to the **products.component.ts** file, get the array of products, and paste it into the **products.service.ts** file



```

File Edit Selection View Go Run Terminal Help
EXPLORER
  > node_modules
  > src
    > app
      > home
      > products
      > products.component.css
      > products.component.html
      > products.component.spec.ts
      > products.component.ts
    > app-routing.module.ts
    > app.component.css
    > app.component.html
    > app.component.spec.ts
    > app.component.ts
    > app.module.ts
    > data-provider.service.spec.ts
    > data-provider.service.ts
    > products.service.spec.ts
    > products.service.ts
  > assets
  > environments

src > app > products > ts products.component.ts
4
5 @Component({
6   selector: 'app-products',
7   templateUrl: './products.component.html',
8   styleUrls: ['./products.component.css']
9 })
10 export class ProductsComponent implements OnInit {
11
12   products: Product[] = [
13     new Product(101, 'Adidas Alphabounce Shoe', 3000, 4.5),
14     new Product(201, 'Samsung LED TV', 30000, 5.0),
15     new Product(301, 'Cadbury 5 Star', 20, 4.5),
16     new Product(401, 'Apple iPhone 11', 53000, 4.8),
17     new Product(501, 'Samsung Fold', 73000, 4.3),
18   ];
19
20   showProducts = true;
21
22   categories = [
23     {title: 'Select Category', total: 1000},
24     {title: 'Home & Kitchen', total: 30},
25     {title: 'Furniture', total: 300},
26   ];
27 }

```

1.4 In the same service file, add one more function named **getProducts()**

```

1 import { Injectable } from '@angular/core';
2 import { Product } from 'src/modal/product';
3
4 @Injectable({
5   providedIn: 'root'
6 })
7 export class ProductsService {
8
9   products: Product[] = [
10     new Product(101, "Adidas Alphabounce Shoe", 3000, 4.5),
11     new Product(201, "Samsung LED TV", 30000, 5.0),
12     new Product(301, "Cadbury 5 Star", 20, 4.5),
13     new Product(401, "Apple iPhone 11", 53000, 4.8),
14     new Product(501, "Samsung Fold", 73000, 4.3),
15   ];
16
17   getProducts(){
18     return this.products;
19   }
20
21   constructor() { }
22 }

```

1.5 Now, add a service injection command in the **products.component.ts** file

```

1 import { Product } from 'src/modal/product';
2 import { FormGroup, FormControl } from '@angular/forms';
3
4 @Component({
5   selector: 'app-products',
6   templateUrl: './products.component.html',
7   styleUrls: ['./products.component.css']
8 })
9 export class ProductsComponent {
10   title: string;
11   total: number;
12   productForm: FormGroup;
13
14   constructor(private productService: ProductsService) {
15     this.title = 'Furniture', total: 300;
16     this.title = 'Clothing', total: 1230;
17   }
18
19   ngOnInit() {
20     this.productForm = new FormGroup({
21       pid: new FormControl(0),
22       name: new FormControl(''),
23       price: new FormControl(0),
24       ratings: new FormControl(0.0),
25       category: new FormControl(this.categories[0])
26     });
27   }
28 }

```

1.6 Copy this **ProductsService** to the **products.services.ts** file

```

1 import { Injectable } from '@angular/core';
2 import { Product } from 'src/modal/product';
3
4 @Injectable({
5   providedIn: 'root'
6 })
7 export class ProductsService {
8
9   products: Product[] = [
10     new Product(101, "Adidas Alphabounce Shoe", 3000, 4.5),
11     new Product(201, "Samsung LED TV", 30000, 5.0),
12     new Product(301, "Cadbury 5 Star", 20, 4.5),
13     new Product(401, "Apple iPhone 11", 53000, 4.8),
14     new Product(501, "Samsung Fold", 73000, 4.3),
15   ];
16
17   getProducts(){
18     return this.products;
19   }
20
21   constructor() { }
22 }

```

1.7 Now, add this in the **products.component.ts** file to fetch all the products from the **products.service.ts** file:

```

src > app > products > ts products.component.ts > ProductsComponent > constructor
25 {title: 'Home & Kitchen', total: 30},
26 {title: 'Furniture', total: 300},
27 {title: 'Clothing', total: 1230},
28 ];
29
30 productForm = new FormGroup({
31   pid: new FormControl(0),
32   name: new FormControl(''),
33   price: new FormControl(0),
34   ratings: new FormControl(0.0),
35   category: new FormControl(this.categories[0])
36 });
37
38 constructor(private productService: ProductService) {}

```

1.8 Now, add one more product in the **products.component.ts** file by typing the below statement:

new Product(101, "Hydrwate Water Bottle", 5000, 5.0),

```

src > app > products > ts products.service.ts > ProductService > products
1 import { Injectable } from '@angular/core';
2 import { Product } from 'src/modal/product';
3
4 @Injectable({
5   providedIn: 'root'
6 })
7 export class ProductService {
8
9   products: Product[] = [
10     new Product(101, "Hydrwate Water Bottle", 5000, 5.0),
11     new Product(101, "Adidas Alphabounce Shoe", 3000, 4.5),
12     new Product(201, "Samsung LED TV", 30000, 5.0),
13     new Product(301, "Cadbury 5 Star", 20, 4.5),
14     new Product(401, "Apple iPhone 11", 53000, 4.8),
15     new Product(501, "Samsung Fold", 73000, 4.3),
16   ];

```

1.9 Execute the code in the terminal by typing the command **ng serve -o**

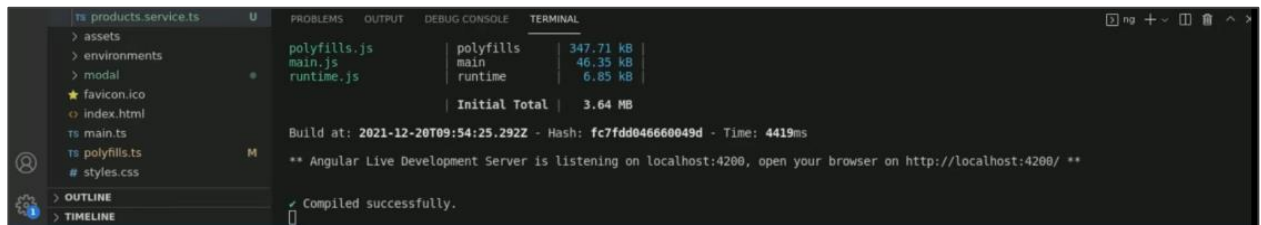
```

ts products.service.ts U PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
assets
environments
modal
favicon.ico
index.html
main.ts
polyfills.ts
styles.css

erishantgmail@ip-172-31-17-157:~/Downloads/estore$ ng s products
Node.js version v17.2.0 detected.
Odd numbered Node.js versions will not enter LTS status and should not be used for production. For more information, please see https://nodejs.org/en/about/releases/.
CREATE src/app/products.service.spec.ts (367 bytes)
CREATE src/app/products.service.ts (137 bytes)
erishantgmail@ip-172-31-17-157:~/Downloads/estore$ ng serve -o

```

1.10 You can see that the code is compiled successfully.



1.11 Now, you will be directed to an eStore home page.



1.12 Click on **Products**, and you will be able to view the list of all the products as shown here:



By following the above steps, you have successfully created services for the data in the application.