

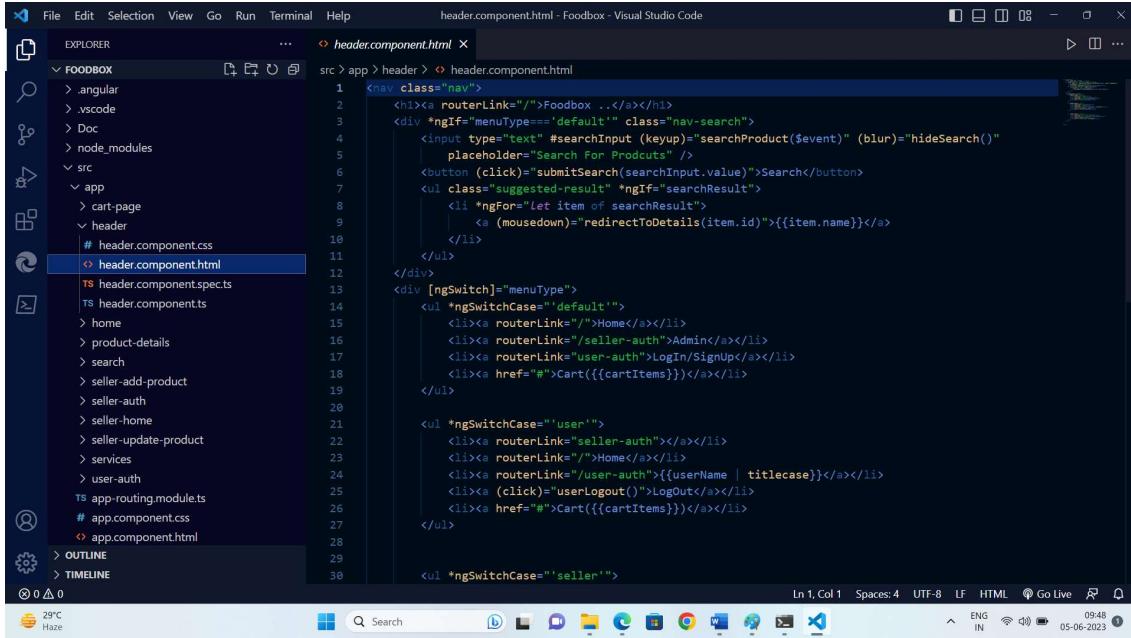
SIMPLILERAN

Full Stack Java Developer (FSD Learner's Academy Project)

Screenshots

Developer: Aman kumar
Email: amankverma5@gmail.com

Foodbox..



The screenshot shows the Visual Studio Code interface with the file `header.component.html` open in the editor. The code is an Angular template for a navigation bar. It includes a search bar, a dropdown menu for different user types (default, seller, user), and links for home, admin, log in, sign up, and logout. The code uses ngIf, ngFor, and ngSwitch directives.

```
<nav class="nav">
  <h1><a routerLink="/">Foodbox ..</a></h1>
  <div *ngIf="menuType==='default'" class="nav-search">
    <input type="text" #searchInput (keyup)="searchProduct($event)" (blur)="hideSearch()" placeholder="Search For Products" />
    <button (click)="submitSearch(searchInput.value)">Search</button>
    <ul class="suggested-result" *ngIf="searchResult">
      <li *ngFor="let item of searchResult">
        <a (mousedown)="redirectToDetails(item.id)">{{item.name}}</a>
      </li>
    </ul>
  </div>
  <div [ngSwitch]="menuType">
    <ul *ngSwitchCase="''default''">
      <li><a routerLink="/">Home</a></li>
      <li><a routerLink="/seller-auth">Admin</a></li>
      <li><a routerLink="/user-auth">LogIn/SignUp</a></li>
      <li><a href="#">Cart({{cartItems}})</a></li>
    </ul>
    <ul *ngSwitchCase="''user''">
      <li><a routerLink="seller-auth">/</a></li>
      <li><a routerLink="/">Home</a></li>
      <li><a routerLink="/user-auth">{{userName | titlecase}}</a></li>
      <li><a (click)="userLogout()"/>LogOut</a></li>
      <li><a href="#">Cart({{cartItems}})</a></li>
    </ul>
    <ul *ngSwitchCase="''seller''">
    </ul>
  </div>

```

File Edit Selection View Go Run Terminal Help header.component.ts - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > header > TS header.components
1 import { Component, OnInit } from '@angular/core';
2 import { Router } from '@angular/router';
3 import { ProductService } from '../services/product.service';
4 import { product } from '../data-type';
5 @Component({
6   selector: 'app-header',
7   templateUrl: './header.component.html',
8   styleUrls: ['./header.component.css']
9 })
10 export class HeaderComponent implements OnInit {
11   menuType: string = 'default';
12   sellerName: string = '';
13   searchResult: undefined | product[];
14   userName: string = "";
15   cartItems: 0;
16   constructor(private route: Router, private product: ProductService) { }
17
18   ngOnInit(): void {
19     this.route.events.subscribe((val: any) => {
20       if (val.url) {
21         if (localStorage.getItem('seller') && val.url.includes('seller')) {
22           console.warn('in seller area')
23           this.menuType = "seller";
24           if (localStorage.getItem('seller')) {
25             let sellerStore = localStorage.getItem('seller');
26             let sellerData = sellerStore && JSON.parse(sellerStore)[0];
27             this.sellerName = sellerData.name;
28           } else if (localStorage.getItem('user')) {
29             let userStore = localStorage.getItem('user');
30             let userData = userStore && JSON.parse(userData);
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF () TypeScript Go Live ⚡ 05:49 05-06-2023

SENSEX Market Brief

File Edit Selection View Go Run Terminal Help home.component.html - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > home > TS home.component.html > ngb-carousel
1 <ngb-carousel *ngIf=" popularProducts">
2   <ng-template *ngFor="let item of popularProducts" ngbSlide>
3     <div class="picsum-img-wrapper">
4       <a routerLink="details/{{item.id}}> 
6     </a>
7   </div>
8   <div class="carousel-caption">
9     <h3>{{item.name}}</h3>
10    <p>{{item.description}}</p>
11  </div>
12 </ng-template>
13 </ngb-carousel>
14
15 <div class="product-list-wrapper">
16   <h1>Fresh Products</h1>
17   <div class="row">
18
19     <div class="col-sm-3" *ngFor="let item of trendyProducts">
20       <div class="product-item">
21         
22
23         <div class="product-details">
24           <h6 class="name-head">{{item.name}}</h6>
25           <div class="product-price">
26             <h6>Price: {{item.price}}</h6>
27           </div>
28         </div>
29       </div>
```

Ln 1, Col 1 Tab Size 4 UTF-8 CRLF HTML Go Live ⚡ 05:49 05-06-2023

SENSEX Market Brief

File Edit Selection View Go Run Terminal Help home.component.ts - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > home > TS home.component.ts < ...
1 import { Component, OnInit } from '@angular/core';
2 import { ProductService } from '../services/product.service';
3 import { product } from '../data-type';
4
5 @Component({
6   selector: 'app-home',
7   templateUrl: './home.component.html',
8   styleUrls: ['./home.component.css']
9 })
10 export class HomeComponent implements OnInit {
11   popularProducts: undefined | product[]
12   trendyProducts: undefined | product[]
13   constructor(private product: ProductService) { }
14
15   ngOnInit(): void {
16     this.product.popularProducts().subscribe((data) => {
17       console.warn(data);
18       this.popularProducts = data;
19     });
20     this.product.trendyProducts().subscribe((data) => {
21       this.trendyProducts = data;
22     })
23   }
24 }
25
```

Ln 1, Col 1 Spaces:2 UTF-8 LF {} TypeScript Go Live ⚡ 05:49 05-06-2023

SENSEX Market Brief

File Edit Selection View Go Run Terminal Help product-details.component.html - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > product-details > TS product-details.component.html > div.row.product-details
1 <div class="row product-details" *ngIf="productData">
2   <div class="col-sm-6">
3     
4   </div>
5   <div class="col-sm-6">
6     <div class="details">
7       <h1>{{productData.name}}</h1>
8       <h3>{{productData.price}}</h3>
9       <div>
10         <h5 [style.backgroundColor]= "productData.color" class="product-color"> </h5>
11       </div>
12       <h6>Category : {{productData.category}}</h6>
13       <h6>Description : {{productData.description}}</h6>
14       <button class="form-button" (click)="AddToCart()">Add to Cart</button>
15       <button class="form-button">Order Now</button>
16       <div class="quantity-group">
17         <button (click)= "handleQuantity('min')"></button>
18         <input type="text" value="{{productQuantity}}"/>
19         <button (click)= "handleQuantity('plus')">+</button>
20       </div>
21     </div>
22   </div>
23 </div>
24 </div>
25
```

Ln 1, Col 1 Spaces:4 UTF-8 LF HTML Go Live ⚡ 05:49 05-06-2023

SENSEX Market Brief

File Edit Selection View Go Run Terminal Help

product-details.component.ts - Foodbox - Visual Studio Code

```
src > app > product-details > TS product-details.component.ts ...
1 import { Component, OnInit } from '@angular/core';
2 import { ActivatedRoute } from '@angular/router';
3 import { ProductService } from '../services/product.service';
4 import { cart, product } from '../data-type';
5
6 @Component({
7   selector: 'app-product-details',
8   templateUrl: './product-details.component.html',
9   styleUrls: ['./product-details.component.css']
10})
11 export class ProductDetailsComponent implements OnInit {
12   productData: undefined | product;
13   productQuantity: number = 1;
14   constructor(private activeRoute: ActivatedRoute, private product: ProductService) { }
15
16   ngOnInit(): void {
17     let productId = this.activeRoute.snapshot.paramMap.get('productId');
18     console.warn(productId);
19     productId && this.product.getProduct(productId).subscribe((result) => {
20       console.warn(result);
21       this.productData = result;
22     })
23   }
24
25   handleQuantity(val: string) {
26     if (this.productQuantity < 20 && val === 'plus') {
27       this.productQuantity += 1
28     } else if (this.productQuantity > 1 && val === 'min') {
29       this.productQuantity -= 1
30     }
31   }
32   AddToCart(){
33     if(this.productData){
34       this.productData.quantity=this.productQuantity;
35       if(localStorage.getItem('user')){
36         this.product.localAddToCart(this.productData);
37       }else{
38
39         let user = localStorage.getItem('user');
40         let userId = user && JSON.parse(user).id
41
42         let cartData = {
43           ...this.productData,
44           userId,
45           productId:this.productData.id,
46         }
47         delete cartData.id;
48
49         this.product.addToCart(cartData).subscribe((result)=>{
50           if(result){
51             alert('product is added in cart')
52           }
53         })
54       }
55     }
56   }
57 }
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF {} TypeScript Go Live ⚡ 05-06-2023

SENSEX Market Brief

File Edit Selection View Go Run Terminal Help

product-details.component.ts - Foodbox - Visual Studio Code

```
src > app > product-details > TS product-details.component.ts ...
28   } else if (this.productQuantity > 1 && val === 'min') {
29     this.productQuantity -= 1
30   }
31   AddToCart(){
32     if(this.productData){
33       this.productData.quantity=this.productQuantity;
34       if(localStorage.getItem('user')){
35         this.product.localAddToCart(this.productData);
36       }else{
37
38         let user = localStorage.getItem('user');
39         let userId = user && JSON.parse(user).id
40
41         let cartData = {
42           ...this.productData,
43           userId,
44           productId:this.productData.id,
45         }
46         delete cartData.id;
47
48         this.product.addToCart(cartData).subscribe((result)=>{
49           if(result){
50             alert('product is added in cart')
51           }
52         })
53       }
54     }
55   }
56 }
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF {} TypeScript Go Live ⚡ 05-06-2023

SENSEX Market Brief

File Edit Selection View Go Run Terminal Help search.component.html - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > search > search.component.html > div.row.search-item
```

Go to component!

```
1 <div class="row search-item" *ngFor="let item of SearchResult">
  2   <div class="col-sm-3">
  3     <a routerLink="/details/{{item.id}}">
  4       
  5     </a>
  6   </div>
  7   <div class="col-sm-8 details">
  8     <h6> <a routerLink="/details/{{item.id}}"/>{{item.name}}</a></h6>
  9     <p><span>{{item.price}}</span></p>
 10    <p><span>{{item.color}}</span><span>{{item.category}}</span></p>
 11    <p>{{item.description}}</p>
 12  </div>
 13</div>
```

Ln 1, Col 1 Spaces: 4 UTF-8 LF HTML Go Live ⚙ ENG IN 09:50 05-06-2023

SENSEX Market Brief

File Edit Selection View Go Run Terminal Help search.component.ts - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > search > search.component.ts ...
```

```
1 import { Component, OnInit } from '@angular/core';
2 import { ActivatedRoute } from '@angular/router';
3 import { product } from '../data-type';
4 import { ProductService } from '../services/product.service';
5
6 @Component({
7   selector: 'app-search',
8   templateUrl: './search.component.html',
9   styleUrls: ['./search.component.css']
10 })
11 export class SearchComponent implements OnInit {
12   SearchResult: undefined | product[];
13   constructor(private activeRoute: ActivatedRoute, private product: ProductService) { }
14   ngOnInit(): void {
15     let query = this.activeRoute.snapshot.paramMap.get('query');
16     console.warn(query);
17     query && this.product.searchProducts(query).subscribe((result) => {
18       this.SearchResult = result
19     })
20   }
21 }
22 }
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF TypeScript Go Live ⚙ ENG IN 09:50 05-06-2023

SENSEX Market Brief

The screenshot shows the Visual Studio Code interface with the file `seller-add-product.component.html` open. The code defines a form for adding a product, including fields for name, price, color, category, and description, along with a file input for image and a submit button.

```
<div class="add-product">
  <h1>Add New Products </h1>
  <p class="p_message">{addProductMessage}</p>
  <form #addProduct="ngForm" (ngSubmit)="submit(addProduct.value)" class="common-form">
    <input class="form-input" type="text" ngModel name="name" placeholder="Enter Product Name">
    <input class="form-input" type="text" ngModel name="price" placeholder="Enter Product Price">
    <input class="form-input" type="text" ngModel name="color" placeholder="Enter Product Color">
    <input class="form-input" type="text" ngModel name="category" placeholder="Enter Product Category">
    <textarea class="form-input" type="text" ngModel name="description" placeholder="Enter Product Description"></textarea>
    <input class="form-input" type="text" ngModel name="image" placeholder="Enter Product Image URL">
    <button class="form-button">Add Product</button>
  </form>
</div>
```

The screenshot shows the Visual Studio Code interface with the file `seller-add-product.component.ts` open. The code defines a component that implements `OnInit`. It uses Angular services like `ProductService` and `data-type` to handle product addition and display success messages.

```
import { Component, OnInit } from '@angular/core';
import { ProductService } from '../services/product.service';
import { product } from '../data-type';

@Component({
  selector: 'app-seller-add-product',
  templateUrl: './seller-add-product.component.html',
  styleUrls: ['./seller-add-product.component.css']
})
export class SellerAddProductComponent implements OnInit {
  addProductMessage: string | undefined;
  constructor(private product: ProductService) { }
  ngOnInit(): void {
  }

  submit(data: product) {
    this.product.addProduct(data).subscribe((result) => {
      console.warn(result);
      if (result) {
        this.addProductMessage = "Product is Successfully Added"
      }
      setTimeout(() => (this.addProductMessage = undefined), 3000);
    })
  }
}
```

File Edit Selection View Go Run Terminal Help seller-auth.component.html - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > seller-auth > seller-auth.component.html > div.seller-auth
  Go to component
  1 <div class="seller-auth">
  2   <div *ngIf="!showLogin" class="signup">
  3     <h1> Admin Sign-Up </h1>
  4     <form #sellerSignUp="ngForm" (ngSubmit)="signUp(sellerSignUp.value)" class="common-form">
  5       <input class="form-input" type="text" ngModel name="name" placeholder="Enter Name">
  6       <input class="form-input" type="password" ngModel name="password" placeholder="Enter Password">
  7       <input class="form-input" type="email" ngModel name="email" placeholder="Enter E-mail">
  8       <button class="form-button">SignUp</button>
  9     <div class="auth-toggle-link" (click)="openlogin()">Already Have Account ?<div class="link"><u> Click Here </u></div>
  10    </form>
  11  </div>
  12
  13
  14  <!--  LogIn  -->
  15
  16  <div *ngIf="showLogin" class="Login">
  17    <h1> Admin Log-In </h1>
  18    <div class="authError" *ngIf="authError">{{authError}}</div>
  19    <form #sellerLogin="ngForm" (ngSubmit)="login(sellerLogin.value)" class="common-form">
  20      <input class="form-input" type="email" ngModel name="email" placeholder="Enter E-mail">
  21      <input class="form-input" type="password" ngModel name="password" placeholder="Enter Password">
  22      <button class="form-button">Login</button>
  23    </form>
  24  </div>
  25
  26
  27
  28 </div>
```

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF HTML ⚡ Go Live ⌂ ENG IN 09:51 05-06-2023

29°C Haze

File Edit Selection View Go Run Terminal Help seller-auth.component.ts - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > seller-auth > TS seller-auth.components.ts > ...
  Go to component
  1 import { Component, OnInit } from '@angular/core';
  2 import { SellerService } from '../services/seller.service';
  3 import { Router } from '@angular/router';
  4 import { Signup } from '../data-type';
  5
  6 @Component({
  7   selector: 'app-seller-auth',
  8   templateUrl: './seller-auth.component.html',
  9   styleUrls: ['./seller-auth.component.css']
 10 })
 11 export class SellerAuthComponent implements OnInit {
 12   constructor(private seller: SellerService, private router: Router) { }
 13   showLogin = false;
 14   authError: string = '';
 15
 16   ngOnInit(): void {
 17     this.seller.reloadSeller()
 18   }
 19   signUp(data: Signup): void {
 20     console.warn(data);
 21     this.seller.userSignUp(data)
 22   }
 23
 24   login(data: Signup): void {
 25     this.authError="";
 26     //console.warn(data);
 27     this.seller.userLogin(data)
 28     this.seller.isLoggedIn.subscribe((isError) => {
 29       if (isError) {
 30         this.authError = "Email or Password is not Correct";
 31       }
 32     })
 33   }
 34 }
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF {} TypeScript ⚡ Go Live ⌂ ENG IN 09:51 05-06-2023

29°C Haze

File Edit Selection View Go Run Terminal Help seller-home.component.html - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > seller-home > seller-home.component.html > div.product-list
```

```
1 <div class="product-list">
2
3   <h1>All Products</h1>
4   <p class="p_message">{{productMessage}}</p>
5   <table>
6     <thead>
7       <tr>
8         <th>Image</th>
9         <th>Name</th>
10        <th>Price</th>
11        <th>Color</th>
12        <th>Category</th>
13        <th>Description</th>
14        <th>Action</th>
15      </tr>
16    </thead>
17    <tbody>
18      <tr *ngFor="let item of productList">
19        <td></td>
20        <td>{{item.name}}</td>
21        <td>{{item.price}}</td>
22        <td>{{item.color}}</td>
23        <td>{{item.category}}</td>
24        <td>{{item.description}}</td>
25        <td>
26          <fa-icon (click)="deleteProduct(item.id)" [icon]="icon"></fa-icon>
27          <a class="adjust-update-link" routerLink="/seller-update-product/{{item.id}}"><fa-icon [icon]="editIcon"></fa-icon> Update</a>
28        </td>
29      </tr>
29    </tbody>
```

Ln 1, Col 1 Spaces: 4 UTF-8 LF HTML ⚡ Go Live ⚡ 05-06-2023

29°C Haze

File Edit Selection View Go Run Terminal Help seller-home.component.ts - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > seller-home > seller-home.component.ts > ...
```

```
1 import { Component, OnInit } from '@angular/core';
2 import { ProductService } from '../services/product.service';
3 import { product } from '../data-type';
4 import { faTrash, faEdit } from '@fortawesome/free-solid-svg-icons'
5
6 @Component({
7   selector: 'app-seller-home',
8   templateUrl: './seller-home.component.html',
9   styleUrls: ['./seller-home.component.css']
10 })
11 export class Seller HomeComponent implements OnInit {
12   productList: undefined | product[];
13   productMessage: undefined | string;
14   icon=faTrash;
15   editIcon=faEdit;
16
17   constructor(private product: ProductService) { }
18
19   ngOnInit(): void {
20     this.list();
21   }
22
23
24
25
26
27
28
29
30 }
```

```
deleteProduct(id: number) {
  console.warn("test id", id)
  this.product.deleteProduct(id).subscribe((result) => {
    if (result) {
      this.productMessage = "Product is Removed";
    }
  })
}
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF {} TypeScript ⚡ Go Live ⚡ 05-06-2023

29°C Haze

File Edit Selection View Go Run Terminal Help seller-update-product.component.html - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > seller-add-product > seller-add-product.component.css
src > app > seller-add-product > seller-add-product.component.html
src > app > seller-add-product > seller-add-product.component.spec.ts
src > app > seller-auth > seller-auth.component.css
src > app > seller-auth > seller-auth.component.html
src > app > seller-auth > seller-auth.component.spec.ts
src > app > seller-home > seller-home.component.css
src > app > seller-home > seller-home.component.html
src > app > seller-home > seller-home.component.spec.ts
src > app > seller-update-product > seller-update-product.component.css
src > app > seller-update-product > seller-update-product.component.html
src > app > seller-update-product > seller-update-product.component.spec.ts
src > app > services > user-auth > user-auth.module.ts
src > app > app-routing.module.ts
src > app > component.html
```

OUTLINE TIMELINE

Ln 1, Col 1 Spaces: 4 UTF-8 LF HTML Go Live

29°C Haze ENG IN 09:51 05-06-2023

Go to component

```
<div class="update-product">
  <h1>Update Products </h1>
  <p class="p_message">{{productMessage}}</p>
  <form #addProduct="ngForm" (ngSubmit)="submit(addProduct.value)" class="common-form">
    <input class="form-input" type="text" ngModel="{{productData?.name}}" name="name"
      placeholder="Enter Product Name">

    <input class="form-input" type="text" ngModel="{{productData?.price}}" name="price"
      placeholder="Enter Product Price">

    <input class="form-input" type="text" ngModel="{{productData?.color}}" name="color"
      placeholder="Enter Product Color">

    <input class="form-input" type="text" ngModel="{{productData?.category}}" name="category"
      placeholder="Enter Product Category">

    <textarea class="form-input" type="text" ngModel="{{productData?.description}}" name="description"
      placeholder="Enter Product Description"></textarea>

    <input class="form-input" type="text" ngModel="{{productData?.image}}" name="image"
      placeholder="Enter Product Image URL">

    <button class="form-button">Update Product</button>
  </form>
</div>
```

File Edit Selection View Go Run Terminal Help seller-update-product.component.ts - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > seller-add-product > TS seller-add-product.component.ts > ...
src > app > seller-add-product > seller-add-product.component.css
src > app > seller-add-product > seller-add-product.component.html
src > app > seller-add-product > seller-add-product.component.spec.ts
src > app > seller-auth > seller-auth.component.css
src > app > seller-auth > seller-auth.component.html
src > app > seller-auth > seller-auth.component.spec.ts
src > app > seller-home > seller-home.component.css
src > app > seller-home > seller-home.component.html
src > app > seller-home > seller-home.component.spec.ts
src > app > seller-update-product > seller-update-product.component.css
src > app > seller-update-product > seller-update-product.component.html
src > app > seller-update-product > seller-update-product.component.spec.ts
src > app > services > user-auth > user-auth.module.ts
src > app > app-routing.module.ts
src > app > component.html
```

OUTLINE TIMELINE

Ln 1, Col 1 Spaces: 2 UTF-8 LF TypeScript Go Live

29°C Haze ENG IN 09:51 05-06-2023

```
import { Component, OnInit } from '@angular/core';
import { ActivatedRoute } from '@angular/router';
import { ProductService } from '../services/product.service';
import { product } from '../data-type';

@Component({
  selector: 'app-seller-update-product',
  templateUrl: './seller-update-product.component.html',
  styleUrls: ['./seller-update-product.component.css']
})
export class SellerUpdateProductComponent implements OnInit {
  productId: undefined | product;
  productMessage: undefined | string;
  constructor(private route: ActivatedRoute, private product: ProductService) { }

  ngOnInit(): void {
    let productId = this.route.snapshot.paramMap.get('id')
    console.warn(productId)
    productId && this.product.getProduct(productId).subscribe((data) => {
      console.warn(data);
      this.productData = data
    })
  }
  submit(data: any) {
    console.warn(data);
    if (this.productData) {
      data.id = this.productData.id;
    }
    this.product.updateProduct(data).subscribe((result) => {
      if (result) {
        console.log(result);
      }
    })
  }
}
```

product.service.ts - Foodbox - Visual Studio Code

```
src > app > services > TS product.service.ts > ...  
1 import { HttpClient } from '@angular/common/http';  
2 import { EventEmitter, Injectable } from '@angular/core';  
3 import { cart, product } from '../data-type';  
4  
5 @Injectable({  
  providedIn: 'root'  
})  
6  
export class ProductService {  
  cartData = new EventEmitter<product[] | []>()  
  constructor(private http: HttpClient) {}  
  addProduct(data: product) {  
    return this.http.post('http://localhost:3000/products', data)  
  }  
  productList() {  
    return this.http.get<product[]>('http://localhost:3000/products');  
  }  
  deleteProduct(id: number) {  
    return this.http.delete(`http://localhost:3000/products/${id}`)  
  }  
  getProduct(id: string) {  
    return this.http.get<product>(`http://localhost:3000/products/${id}`)  
  }  
  updateProduct(product: product) {  
    console.warn(product);  
    return this.http.put<product>(`http://localhost:3000/products/${product.id}`, product)  
  }  
  popularProducts() {  
    return this.http.get<product[]>('http://localhost:3000/products?_limit=3');  
  }  
}
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF {} TypeScript Go Live ⚡ 05/06/2023

product.service.ts - Foodbox - Visual Studio Code

```
src > app > services > TS product.service.ts > ...  
28 }  
29 popularProducts() {  
30   return this.http.get<product[]>('http://localhost:3000/products?_limit=3');  
31 }  
32 trendyProducts() {  
33   return this.http.get<product[]>('http://localhost:3000/products?_limit=12');  
34 }  
35 searchProducts(query: string) {  
36   return this.http.get<product[]>(`http://localhost:3000/products?q=${query}`)  
37 }  
38 localAddToCart(data: product){  
39   let cartData = [];  
40   let localCart = localStorage.getItem('localCart');  
41   if(localCart){  
42     localStorage.setItem('localCart', JSON.stringify([data]));  
43   }  
44   else{  
45     cartData=JSON.parse(localCart);  
46     cartData.push(data)  
47     localStorage.setItem('localCart', JSON.stringify(cartData));  
48   }  
49   this.cartData.emit(cartData);  
50 }  
51 addToCart(cartData: cart){  
52   return this.http.post('http://localhost:3000/cart', cartData);  
53 }  
54 }  
55 }
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF {} TypeScript Go Live ⚡ 05/06/2023

File Edit Selection View Go Run Terminal Help

SELLER SERVICES

```
src > app > services > TS seller.service.ts ...
```

```
1 import { EventEmitter, Injectable } from '@angular/core';
2 import { HttpClient } from '@angular/common/http';
3 import { Signup, login } from './data-type';
4 import { BehaviorSubject } from 'rxjs';
5 import { Route, Router } from '@angular/router';
6 @Injectable({
7   providedIn: 'root'
8 })
9 export class SellerService {
10   isSellerLoggedIn = new BehaviorSubject<boolean>(false);
11   isLoggedInError = new EventEmitter<boolean>(false);
12
13   constructor(private http: HttpClient, private router: Router) { }
14   userSignUp(data: Signup) {
15     this.http
16       .post('http://localhost:3000/seller', data, { observe: 'response' })
17       .subscribe((result) => {
18         this.isSellerLoggedIn.next(true);
19         localStorage.setItem('seller', JSON.stringify(result.body));
20         this.router.navigate(['seller-home']);
21       });
22   }
23
24   reloadSeller() {
25     if (localStorage.getItem('seller')) {
26       this.isSellerLoggedIn.next(true);
27       this.router.navigate(['seller-home']);
28     }
29   }
30 }
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF {} TypeScript Go Live 09:52 05-06-2023

SL - AFG SL won by 132 r...

File Edit Selection View Go Run Terminal Help

SELLER SERVICES

```
src > app > services > TS seller.service.ts ...
```

```
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
```

```
userLogin(data: login) {
  console.warn(data);
  this.http.get(`http://localhost:3000/seller?email=${data.email}&password=${data.password}`, { observe: 'response' })
    .subscribe((result: any) => {
      console.warn(result);
      if(result && result.body && result.body.length){
        console.warn("user logged in")
        localStorage.setItem('seller', JSON.stringify(result.body));
        this.router.navigate(['seller-home']);
      }else{
        console.warn("login failed");
        this.isLoggedInError.emit(true);
      }
    })
}
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF {} TypeScript Go Live 09:52 05-06-2023

SL - AFG SL won by 132 r...

The screenshot shows the Visual Studio Code interface with the following details:

- File Path:** user.service.ts - Foodbox - Visual Studio Code
- Editor Content:** The code for the UserService class. It includes imports for Injectable, HttpClient, Router, and Signup/Login types. The class has methods for userSignUp and userLogin, which make POST requests to "http://localhost:3000/users". The userLogin method also handles the response and stores user data in localStorage.
- Explorer:** Shows the project structure under the FOODBOX folder, including files like seller-auth.component.css, seller-home.component.css, and various service and component files.
- Bottom Bar:** Includes tabs for TypeScript, Go Live, and other development tools.

This screenshot is identical to the one above, showing the same code for the UserService and the same project structure in the Explorer view. The bottom bar also shows the same development tools.

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure under the "FOODBOX" folder:
 - seller-home.component.spec.ts
 - seller-home.component.ts
 - seller-update-product
 - # seller-update-product.component.css
 - seller-update-product.component.html
 - seller-update-product.component.spec.ts
 - seller-update-product.component.ts
 - services
 - product.service.spec.ts
 - product.service.ts
 - seller.service.spec.ts
 - seller.service.ts
 - user.service.spec.ts
 - user.service.ts
 - user-auth
 - # user-auth.component.css
 - user-auth.component.html
 - user-auth.component.spec.ts
 - user-auth.component.ts
 - app-routing.module.ts
 - # app.component.css
 - app.component.html
 - app.component.spec.ts
 - ann-component.ts
- Editor (Center):** Displays the content of "user-auth.component.html". The code includes two sections: "User Sign-Up" and "User Log-In", each with input fields for name, email, and password, and buttons for "Sign Up" and "Log In".

```
<div *ngIf="showLogin" class="user-login">
  <h3>User Log-In</h3>
  <form #userLogin="ngForm" class="common-form" (ngSubmit)="login(userLogin.value)">
    <input class="form-input" type="text" placeholder="Enter User E-mail" name="email" ngModel>
    <input class="form-input" type="password" placeholder="Enter User Password" name="password" ngModel>
    <button class="form-button" type="submit">Log In</button>
    <p>Don't Have Account? <a (click)="openSignUp()">Click Here</a></p>
  </form>
</div>

<div *ngIf="!showLogin" class="user-signup">
  <h3>User Sign-Up</h3>
  <form #userSignUp="ngForm" class="common-form" (ngSubmit)="signUp(userSignUp.value)">
    <input class="form-input" type="text" placeholder="Enter User Name" name="name" ngModel>
    <input class="form-input" type="text" placeholder="Enter User E-mail" name="email" ngModel>
    <input class="form-input" type="password" placeholder="Enter User Password" name="password" ngModel>
    <button class="form-button" type="submit">Sign Up</button>
    <p>Already Have Account? <a (click)="openLogin()">Click Here</a></p>
  </form>
</div>
```
- Bottom Status Bar:** Shows file navigation (Ln 1, Col 1), workspace settings (Spaces: 4, UTF-8, LF, HTML), and a "Go Live" button.
- Bottom Icons:** Includes icons for search, file operations, and system status.



```
File Edit Selection View Go Run Terminal Help user-auth.component.ts - Foodbox - Visual Studio Code

EXPLORER ... TS user-auth.component.ts x
src > app > user-auth > TS user-auth.components.ts ...
1 import { Component, OnInit } from '@angular/core';
2 import { SignUp, cart, login, product } from './data-type';
3 import { UserService } from '../services/user.service';
4 import { ProductService } from '../services/product.service';
5
6 @Component({
7   selector: 'app-user-auth',
8   templateUrl: './user-auth.component.html',
9   styleUrls: ['./user-auth.component.css']
10 })
11 export class UserAuthComponent implements OnInit {
12   showLogin: boolean = true;
13   authError: string = '';
14   constructor(private user: UserService, private product: ProductService) {
15   }
16
17   ngOnInit(): void {
18
19     this.user.userAuthReload();
20
21   }
22   signUp(data: SignUp) {
23
24     this.user.userSignUp(data);
25   }
26   login(data: login) {
27
28     this.user.userLogin(data);
29     this.user.invalidUserAuth.subscribe((result: any) => {
30       if (result) {
31         console.warn("remove else");
32       }
33     });
34   }
35 }

SL - AFG SL won by 132 ...
Search ... LF {} TypeScript Go Live 09:53 05-06-2023
```

```
File Edit Selection View Go Run Terminal Help user-auth.component.ts - Foodbox - Visual Studio Code
EXPLORER ...
FOODBOX ...
TS seller-home.component.spec.ts
TS seller-home.component.ts
seller-update-product ...
# seller-update-product.component.css
seller-update-product.component.html
TS seller-update-product.component.spec...
TS seller-update-product.component.ts
seller-update-product.component.ts
services ...
TS product.service.spec.ts
TS product.service.ts
TS seller.service.spec.ts
TS seller.service.ts
TS user.service.spec.ts
TS user.service.ts
user-auth ...
# user-auth.component.css
user-auth.component.html
TS user-auth.component.spec.ts
TS user-auth.component.ts
TS app-routing.module.ts
# app.component.css
app.component.html
TS app.component.spec.ts
TS ann-component.ts
> OUTLINE
> TIMELINE
TS user-auth.component.ts
src > app > user-auth > TS user-auth.component.ts ...
35      })
36    }
37  }
38  openSignUp() {
39    this.showLogin = false;
40  }
41  openLogin() {
42    this.showLogin = true;
43  }
44  localCartToRemoteCart() {
45    let data = localStorage.getItem('localCart');
46    if (data) {
47      let cartDataList: product = JSON.parse(data)
48      let user = localStorage.getItem('user');
49      let userId = user && JSON.parse(user).id;
50
51      cartDataList['forEach']((product: product) => {
52        let cartData: cart = {
53          ...product,
54          productId: product.id,
55          userId
56        };
57        delete cartData.id;
58
59        setTimeout(() => {
60          this.product.addToCart(cartData).subscribe((result) => {
61            if (result) {
62              console.warn("item store in db");
63            }
64          });
65        }, 1000);
66      });
67    }
68  }
69
```

File Edit Selection View Go Run Terminal Help app-routing.module.ts - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > ts app-routing.module.ts > ...
```

```
32     canActivate: [AuthGuard]
33   },
34   {
35     component: SellerUpdateProductComponent,
36     path: 'seller-update-product/:id',
37     canActivate: [AuthGuard]
38   },
39   {
40     component: SearchComponent,
41     path: 'search/:query'
42   },
43   {
44     component: ProductDetailsComponent,
45     path: 'details/:productId'
46   },
47   {
48     component: UserAuthComponent,
49     path: 'user-auth'
50   };
51
52 @NgModule({
53   imports: [RouterModule.forRoot(routes)],
54   exports: [RouterModule]
55 })
56 export class AppRoutingModule { }
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF {} TypeScript Go Live ⚡ 05:53 05-06-2023

UAE - WI Wi won by 7 wi...

File Edit Selection View Go Run Terminal Help app.component.spec.ts - Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > ts app.component.spec.ts > ...
```

```
1 import { TestBed } from '@angular/core/testing';
2 import { RouterTestingModule } from '@angular/router/testing';
3 import { AppComponent } from './app.component';
4
5 describe('AppComponent', () => {
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       imports: [
9         RouterTestingModule
10      ],
11      declarations: [
12        AppComponent
13      ],
14    }).compileComponents();
15  });
16
17 it('should create the app', () => {
18   const fixture = TestBed.createComponent(AppComponent);
19   const app = fixture.componentInstance;
20   expect(app).toBeTruthy();
21 });
22
23 it('should have as title \'ecom-project\'', () => {
24   const fixture = TestBed.createComponent(AppComponent);
25   const app = fixture.componentInstance;
26   expect(app.title).toEqual('ecom-project');
27 });
28
29 it('should render title', () => {
30   const fixture = TestBed.createComponent(AppComponent);
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF {} TypeScript Go Live ⚡ 09:54 05-06-2023

UAE - WI Wi won by 7 wi...

File Edit Selection View Go Run Terminal Help app.component.ts · Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > TS app.components.ts > ...
1 import { Component } from '@angular/core';
2 import { SellerService } from './services/seller.service';
3
4 @Component({
5   selector: 'app-root',
6   templateUrl: './app.component.html',
7   styleUrls: ['./app.component.css']
8 })
9 export class AppComponent {
10   title = 'ecom-project';
11 }
12
13
```

UAE - WI
Wi won by 7 wi...

Ln 1, Col 1 Spaces: 2 UTF-8 LF {} TypeScript Go Live ⚡ ENG IN 09:54 05-06-2023

File Edit Selection View Go Run Terminal Help app.module.ts · Foodbox - Visual Studio Code

EXPLORER FOODBOX

```
src > app > TS app.modules.ts > ...
1 import { NgModule } from '@angular/core';
2 import { BrowserModule } from '@angular/platform-browser';
3
4 import { AppRoutingModule } from './app-routing.module';
5 import { AppComponent } from './app.component';
6 import { HeaderComponent } from './header/header.component';
7 import { HomeComponent } from './home/home.component';
8 import { SellerAuthComponent } from './seller-auth/seller-auth.component';
9 import { FormsModule } from '@angular/forms';
10 import { HttpClientModule } from '@angular/common/http';
11 import { SellerHomeComponent } from './seller-home/seller-home.component';
12 import { SellerAddProductComponent } from './seller-add-product/seller-add-product.component';
13 import { FontAwesomeModule } from '@fortawesome/angular-fontawesome';
14 import { SellerUpdateProductComponent } from './seller-update-product/seller-update-product.component';
15 import { NgbModule } from '@ng-bootstrap/ng-bootstrap';
16 import { SearchComponent } from './search/search.component';
17 import { ProductDetailsComponent } from './product-details/product-details.component';
18 import { UserAuthComponent } from './user-auth/user-auth.component';
19 import { CartPageComponent } from './cart-page/cart-page.component';
20
21 @NgModule({
22   declarations: [
23     AppComponent,
24     HeaderComponent,
25     HomeComponent,
26     SellerAuthComponent,
27     SellerHomeComponent,
28     SellerAddProductComponent,
29     SellerUpdateProductComponent,
30     SearchComponent,
31   ],
32   imports: [
33     BrowserModule,
34     FormsModule,
35     HttpClientModule,
36     NgbModule,
37     AppRoutingModule,
38     FontAwesomeModule,
39   ],
40   providers: [
41     SellerService,
42     UserAuthGuard,
43   ],
44   bootstrap: [AppComponent]
45 })
46 export class AppModule {}
```

29°C Haze

Ln 1, Col 1 Spaces: 2 UTF-8 LF {} TypeScript Go Live ⚡ ENG IN 09:54 05-06-2023

The screenshot shows the Visual Studio Code interface with the file `auth.guard.ts` open in the editor. The code implements an `AuthGuard` class that checks if a user is logged in from local storage before activating a route.

```
import { Injectable } from '@angular/core';
import { ActivatedRouteSnapshot, CanActivate, RouterStateSnapshot, UrlTree } from '@angular/router';
import { Observable } from 'rxjs';
import { SellerService } from './services/seller.service';

@Injectable({
  providedIn: 'root'
})
export class AuthGuard implements CanActivate {
  constructor(private sellerService: SellerService) { }

  canActivate(
    route: ActivatedRouteSnapshot,
    state: RouterStateSnapshot): Observable<boolean | UrlTree | Promise<boolean | UrlTree> | boolean {
    if (localStorage.getItem('seller')) {
      return true;
    }

    return this.sellerService.isSellerLoggedIn();
  }
}
```

The screenshot shows the Visual Studio Code interface with the file `data-type.ts` open in the editor. It defines several TypeScript interfaces for user authentication and product data.

```
export interface SignUp{
  name:string,
  password:string,
  email:string
}

export interface login{
  email:string,
  password:string
}

export interface product{
  [x: string]: any
  name:string,
  price:number,
  category:string,
  color:string,
  description:string,
  image:string,
  id:number,
  quantity:undefined | number
}

export interface cart{
  name:string,
  price:number,
  category:string,
  color:string,
  description:string,
  image:string,
  id:number | undefined,
  quantity:undefined | number,
}
```

File Edit Selection View Go Run Terminal Help index.html - Foodbox - Visual Studio Code

EXPLORER FOODBOX

- # user-auth.component.css
- user-auth.component.html
- TS user-auth.component.spec.ts
- TS user-auth.component.ts
- TS app-routing.module.ts
- # app.component.css
- app.component.html
- TS app.component.spec.ts
- TS app.component.ts
- TS app.module.ts
- TS auth.guard.specs
- TS auth.guard.ts
- TS data-type.ts
- > assets
- ★ favicon.ico
- index.html
- TS main.ts
- # styles.css
- editorconfig
- .gitignore
- {.httrc}
- {angular.json}
- {db.json}
- {package-lock.json}

> OUTLINE > TIMELINE

Ln 1, Col 1 Spaces: 2 UTF-8 LF HTML ⚡ Go Live ⚡ 05-06-2023

29°C Haze

File Edit Selection View Go Run Terminal Help styles.css - Foodbox - Visual Studio Code

EXPLORER FOODBOX

- # user-auth.components.css
- user-auth.component.html
- TS user-auth.component.spec.ts
- TS user-auth.component.ts
- TS app-routing.module.ts
- # app.component.css
- app.component.html
- TS app.component.spec.ts
- TS app.component.ts
- TS app.module.ts
- TS auth.guard.specs
- TS auth.guard.ts
- TS data-type.ts
- > assets
- ★ favicon.ico
- index.html
- TS main.ts
- # styles.css
- editorconfig
- .gitignore
- {.httrc}
- {angular.json}
- {db.json}
- {package-lock.json}

> OUTLINE > TIMELINE

Ln 1, Col 1 Spaces: 4 UTF-8 LF CSS ⚡ Go Live ⚡ 05-06-2023

29°C Haze