Web Technology Lab

Lab4

Kruti

21011102025

index.html

<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<title>Homes</title>

<base href="/">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link rel="icon" type="image/x-icon" href="favicon.ico">

<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Be+Vietnam+Pro:ital,wght@0,400;0,700;1,400;1 ,700&display=swap" rel="stylesheet">

</head>

<body>

<app-root></app-root>

</body>

</html>

Main.ts

|  |
| --- |
| /\*   * Protractor support is deprecated in Angular. * Protractor is used in this example for compatibility with Angular documentation tools.   \*/ import { bootstrapApplication,provideProtractorTestingSupport } from '@angular/platformbrowser'; import { AppComponent } from './app/app.component'; import { provideRouter } from '@angular/router'; import routeConfig from './app/routes';  bootstrapApplication(AppComponent,  { providers: [ provideProtractorTestingSupport(), provideRouter(routeConfig)  ]  }  ).catch(err => console.error(err)); |

Styles.css

|  |
| --- |
| /\* Updated styles.css \*/    /\* Add a background color and adjust font styles \*/ body { font-family: 'Be Vietnam Pro', sans-serif; background-color: #f2f2f2; color: #333;  }    /\* Update the primary color \*/  :root {  --primary-color: #e74c3c; /\* Change to a different color \*/  --secondary-color: #3498db;  --accent-color: #f7dc6f;  --shadow-color: #bdc3c7;  }    /\* Modify button styles \*/ button.primary { padding: 12px 20px; border: none; background: var(--primary-color); color: white; border-radius: 5px; cursor: pointer; transition: background 0.3s ease;  } button.primary:hover { background: #c0392b;  }    /\* Adjust input styles \*/ input[type="text"], input[type="email"] { border: none; padding: 12px; border-radius: 5px; margin-right: 8px; display: inline-block; width: 50%; background-color: #f5f5f5;  }    /\* Update header styles \*/ header { display: flex; justify-content: space-between; |
| align-items: center; height: 80px; padding: var(--content-padding); background-color: var(--secondary-color); color: white;  }    /\* Update listing styles \*/  .listing { background: white; border-radius: 10px; box-shadow: 0px 2px 5px var(--shadow-color); margin: 10px; padding: 20px;  }    .listing-heading { font-size: 24px; margin-bottom: 10px; color: var(--primary-color);  }    .listing-photo { height: 200px; width: 100%; object-fit: cover; border-radius: 10px;  }    .listing-location { color: var(--secondary-color); font-size: 14px;  }    .listing a { text-decoration: none; color: var(--primary-color);  }    .listing a:hover { text-decoration: underline;  } |

app.component.ts

import { Component } from '@angular/core'; import { HomeComponent } from './home/home.component'; import { RouterModule } from '@angular/router';

|  |
| --- |
| @Component({ selector: 'app-root', standalone: true, imports: [ HomeComponent,  RouterModule,  ], template: `  <main>  <a [routerLink]="['/']">  <header class="brand-name">  <img class="brand-logo" src="/assets/logo.svg" alt="logo" aria-hidden="true">  </header>  </a>  <section class="content">  <router-outlet></router-outlet>  </section>  </main> `, styleUrls: ['./app.component.css'],  }) export class AppComponent { title = 'homes';  } |

app.component.css

|  |
| --- |
| :host {  --content-padding: 10px;  } header { display: block; height: 60px; padding: var(--content-padding); box-shadow: 0px 5px 25px var(--shadow-color);  }  .content { padding: var(--content-padding);  } |

Housing.service.ts

|  |
| --- |
| import { Injectable } from '@angular/core'; import { HousingLocation } from './housinglocation';    @Injectable({ providedIn: 'root' |

}) export class HousingService {

url = 'http://localhost:3000/locations';

|  |
| --- |
| async getAllHousingLocations(): Promise<HousingLocation[]> { const data = await fetch(this.url); return await data.json() ?? [];  } async getHousingLocationById(id: number): Promise<HousingLocation | undefined> { const data = await fetch(`${this.url}/${id}`); return await data.json() ?? {};  } submitApplication(firstName: string, lastName: string, email: string) { console.log(firstName, lastName, email);  }  } |

Housinglocation.ts

export interface HousingLocation { id: number; name: string; city: string; state: string; photo: string; availableUnits: number; wifi: boolean; laundry: boolean; }

Routes.ts

|  |
| --- |
| import { Routes } from '@angular/router'; import { HomeComponent } from './home/home.component'; import { DetailsComponent } from './details/details.component';  const routeConfig: Routes = [  { path: '', component: HomeComponent, title: 'Home page'  }, { path: 'details/:id', |

component: DetailsComponent, title: 'Home details'

}

];

export default routeConfig;

housing-location.component.ts

|  |
| --- |
| .listing { background: var(--accent-color); border-radius: 30px; padding-bottom: 30px;  }  .listing-heading { color: var(--primary-color); padding: 10px 20px 0 20px;  }  .listing-photo { height: 250px; |

|  |
| --- |
| import { Component, Input } from '@angular/core'; import { CommonModule } from '@angular/common'; import { HousingLocation } from '../housinglocation'; import { RouterLink, RouterOutlet} from '@angular/router';  @Component({ selector: 'app-housing-location', standalone: true, imports: [CommonModule, RouterLink, RouterOutlet], template: `  <section class="listing">  <img class="listing-photo" [src]="housingLocation.photo" alt="Exterior photo of {{housingLocation.name}}">  <h2 class="listing-heading">{{ housingLocation.name }}</h2>  <p class="listing-location">{{ housingLocation.city}}, {{housingLocation.state }}</p>  <a [routerLink]="['/details', housingLocation.id]">Learn More</a>  </section>  `, styleUrls: ['./housing-location.component.css']  })  export class HousingLocationComponent { @Input() housingLocation!: HousingLocation;  } |

Housing-location.component.css

width: 100%; object-fit: cover; border-radius: 30px 30px 0 0; } .listing-location { padding: 10px 20px 20px 20px;

|  |
| --- |
| }  .listing-location::before { content: url("/assets/location-pin.svg") / "";  } section.listing a { padding-left: 20px; text-decoration: none; color: var(--primary-color);  } section.listing a::after { content: "\203A"; margin-left: 5px;  } |

[Home.component.ts](http://home.component.ts/)

|  |
| --- |
| import { Component, inject } from '@angular/core'; import { CommonModule } from '@angular/common'; import { HousingLocationComponent } from '../housing-location/housing-location.component'; import { HousingLocation } from '../housinglocation'; import { HousingService } from '../housing.service';    @Component({ selector: 'app-home', standalone: true, imports: [ CommonModule,  HousingLocationComponent  ], template: ` <section>  <form>  <input type="text" placeholder="Filter by city" #filter>  <button class="primary" type="button"  (click)="filterResults(filter.value)">Search</button>  </form>  </section>  <section class="results">  <app-housing-location  \*ngFor="let housingLocation of filteredLocationList"  [housingLocation]="housingLocation">  </app-housing-location>  </section> |

|  |
| --- |
| `, styleUrls: ['./home.component.css']    })  export class HomeComponent { housingLocationList: HousingLocation[] = []; housingService: HousingService = inject(HousingService); filteredLocationList: HousingLocation[] = [];  constructor() { this.housingService.getAllHousingLocations().then((housingLocationList: HousingLocation[]) => { this.housingLocationList = housingLocationList; this.filteredLocationList = housingLocationList;  });  } filterResults(text: string) { if (!text) { this.filteredLocationList = this.housingLocationList;  } this.filteredLocationList = this.housingLocationList.filter( housingLocation => housingLocation?.city.toLowerCase().includes(text.toLowerCase()) );  }  } |

[Home.component.css](http://home.component.css/)

|  |
| --- |
| /\* Updated home.component.css \*/    .results { display: grid; gap: 20px; /\* Adjust spacing between items \*/ grid-template-columns: repeat(auto-fill, minmax(300px, 1fr)); /\* Adjust min width and column layout \*/ margin-top: 30px;  }    /\* Update input style for better visibility \*/ input[type="text"] { border: none; padding: 8px; border-radius: 5px; margin-right: 8px; display: inline-block; |
| width: 40%; background-color: #f5f5f5; border: 1px solid #ccc;  }    /\* Update button style \*/ button { padding: 10px 20px; border: none; background: var(--primary-color); color: white; border-radius: 5px; cursor: pointer; transition: background 0.3s ease;  } button:hover { background: #c0392b;  }      @media (min-width: 500px) and (max-width: 768px) {  .results { grid-template-columns: repeat(2, 1fr);  } input[type="text"] { width: 70%;  }  }    @media (max-width: 499px) {  .results { grid-template-columns: 1fr;  }  } |

Details.component.ts

|  |
| --- |
| import { Component, inject } from '@angular/core'; import { CommonModule } from '@angular/common'; import { ActivatedRoute } from '@angular/router'; import { HousingService } from '../housing.service'; import { HousingLocation } from '../housinglocation'; import { FormControl, FormGroup, ReactiveFormsModule } from '@angular/forms';    @Component({ selector: 'app-details', standalone: true, imports: [ CommonModule, |

|  |
| --- |
| ReactiveFormsModule  ], template: `  <article>  <img class="listing-photo" [src]="housingLocation?.photo" alt="Exterior photo of {{housingLocation?.name}}"/>  <section class="listing-description">  <h2 class="listing-heading">{{housingLocation?.name}}</h2>  <p class="listing-location">{{housingLocation?.city}}, {{housingLocation?.state}}</p> </section>  <section class="listing-features">  <h2 class="section-heading">About this housing location</h2> <ul>  <li>Units available: {{housingLocation?.availableUnits}}</li>  <li>Does this location have wifi: {{housingLocation?.wifi}}</li>  <li>Does this location have laundry: {{housingLocation?.laundry}}</li>  </ul>  </section>  <section class="listing-apply">  <h2 class="section-heading">Apply now to live here</h2>  <form [formGroup]="applyForm" (submit)="submitApplication()">  <label for="first-name">First Name</label>  <input id="first-name" type="text" formControlName="firstName">  <label for="last-name">Last Name</label>  <input id="last-name" type="text" formControlName="lastName">  <label for="email">Email</label>  <input id="email" type="email" formControlName="email">  <button type="submit" class="primary">Apply now</button>  </form>  </section>  </article>  `, styleUrls: ['./details.component.css']  })  export class DetailsComponent {  route: ActivatedRoute = inject(ActivatedRoute); housingService = inject(HousingService); housingLocation: HousingLocation | undefined;  applyForm = new FormGroup({ firstName: new FormControl(''), lastName: new FormControl(''), email: new FormControl('')  }); constructor() { |
| const housingLocationId = parseInt(this.route.snapshot.params['id'], 10); this.housingService.getHousingLocationById(housingLocationId).then(housingLocation => { this.housingLocation = housingLocation;  });  } submitApplication() { this.housingService.submitApplication( this.applyForm.value.firstName ?? '', this.applyForm.value.lastName ?? '', this.applyForm.value.email ?? ''  );  }    } |

Details.component.css

|  |
| --- |
| .listing-photo { height: 600px; width: 50%; object-fit: cover; border-radius: 30px; float: right;  }    .listing-heading { font-size: 48pt; font-weight: bold; margin-bottom: 15px;  }    .listing-location::before { content: url('/assets/location-pin.svg') / '';  }    .listing-location { font-size: 24pt; margin-bottom: 15px;  }    .listing-features > .section-heading { color: var(--secondary-color); font-size: 24pt; margin-bottom: 15px;  }    .listing-features { margin-bottom: 20px; |
| }    .listing-features li { font-size: 14pt;  } li { list-style-type: none;  }    .listing-apply .section-heading { font-size: 18pt; margin-bottom: 15px;  } label, input { display: block;  } label { color: var(--secondary-color); font-weight: bold; text-transform: uppercase; font-size: 12pt;  } input { font-size: 16pt; margin-bottom: 15px; padding: 10px; width: 400px; border-top: none; border-right: none; border-left: none; border-bottom: solid .3px;  }  @media (max-width: 1024px) {  .listing-photo { width: 100%; height: 400px;  }  } |

Db.json

{

"locations": [

{

"id": 0,

"name": "Acme Fresh Start Housing",

"city": "Chicago",

"state": "IL",

|  |
| --- |
| "photo": "https://angular.io/assets/images/tutorials/faa/bernard-hermantCLKGGwIBTaY-unsplash.jpg",  "availableUnits": 4,  "wifi": true,  "laundry": true  },  {  "id": 1,  "name": "A113 Transitional Housing",  "city": "Santa Monica",  "state": "CA",  "photo": "https://angular.io/assets/images/tutorials/faa/brandon-griggswR11KBaB86U-unsplash.jpg", "availableUnits": 0,  "wifi": false,  "laundry": true  },  {  "id": 2,  "name": "Warm Beds Housing Support",  "city": "Juneau",  "state": "AK",  "photo": "https://angular.io/assets/images/tutorials/faa/i-do-nothing-but-lovelAyXdl1-Wmc-unsplash.jpg", "availableUnits": 1,  "wifi": false,  "laundry": false  },  {  "id": 3,  "name": "Homesteady Housing",  "city": "Chicago",  "state": "IL",  "photo": "https://angular.io/assets/images/tutorials/faa/ian-macdonald-  W8z6aiwfi1E-unsplash.jpg",  "availableUnits": 1,  "wifi": true,  "laundry": false  },  {  "id": 4,  "name": "Happy Homes Group",  "city": "Gary",  "state": "IN",  "photo": "https://angular.io/assets/images/tutorials/faa/krzysztof-hepner-  978RAXoXnH4-unsplash.jpg",  "availableUnits": 1,  "wifi": true,  "laundry": false  },  { |

|  |
| --- |
| "id": 5,  "name": "Hopeful Apartment Group",  "city": "Oakland",  "state": "CA",  "photo": "https://angular.io/assets/images/tutorials/faa/r-architectureJvQ0Q5IkeMM-unsplash.jpg",  "availableUnits": 2,  "wifi": true,  "laundry": true  },  {  "id": 6,  "name": "Seriously Safe Towns",  "city": "Oakland",  "state": "CA",  "photo": "https://angular.io/assets/images/tutorials/faa/phil-hearing-  IYfp2Ixe9nM-unsplash.jpg",  "availableUnits": 5,  "wifi": true,  "laundry": true  },  {  "id": 7,  "name": "Hopeful Housing Solutions",  "city": "Oakland",  "state": "CA",  "photo": "https://angular.io/assets/images/tutorials/faa/r-architectureGGupkreKwxA-unsplash.jpg",  "availableUnits": 2,  "wifi": true,  "laundry": true  },  {  "id": 8,  "name": "Seriously Safe Towns",  "city": "Oakland",  "state": "CA",  "photo": "https://angular.io/assets/images/tutorials/faa/saru-robert-9rP3mxf8qWIunsplash.jpg",  "availableUnits": 10,  "wifi": false,  "laundry": false  },  {  "id": 9,  "name": "Capital Safe Towns",  "city": "Portland",  "state": "OR",  "photo": "https://angular.io/assets/images/tutorials/faa/webaliser-\_TPTXZd9mOounsplash.jpg",  "availableUnits": 6, |

"wifi": true,

"laundry": true

}

]

}

Output Screenshot:



