

## ADDING in angular:

```
add-course.component.html X
app > add-course > add-course.component.html > div.container > div.row > div.col-6 > form > div.row.mb-3
1  <div class="container">
2    <h1>Add New Course</h1>
3
4    <div class="row">
5      <div class="col-6">
6        <form #form="ngForm" (ngSubmit)="addCourse()" novalidate>
7          <div class="row mb-3">
8            <label for="name" class="col-sm-2 col-form-label">Name</label>
9            <input style="border-color: #darkgray;" type="text" class="form-control" id="name" name="name" [(ngModel)]="addCourseAtt.name" required #nameField
10          </div>
11
12          <div class="row mb-3">
13            <label for="duration" class="col-sm-2 col-form-label">Duration</label>
14            <input style="border-color: #darkgray;" type="text" class="form-control" id="duration" name="duration" [(ngModel)]="addCourseAtt.duration" require
15          </div>
16
17          <div class="row mb-3">
18            <label for="description" class="col-sm-2 col-form-label">Description</label>
19            <input style="border-color: #darkgray;" type="text" class="form-control" id="description" name="description" [(ngModel)]="addCourseAtt.description
20          </div>
21
22          <button type="submit" class="btn-add" [disabled]="form.invalid">Add</button>
23          <button type="button" class="btn-cancel" (click)="cancel()">Cancel</button>
24        </form>
25      </div>
26    </div>
27
28
29
30
31  </div>
```

TS add-course.component.spec.ts 6 X

app > add-course > TS add-course.component.spec.ts > ...

```
1  import { ComponentFixture, TestBed } from '@angular/core/testing';
2
3  import { AddCourseComponent } from './add-course.component';
4
5  describe('AddCourseComponent', () => {
6    let component: AddCourseComponent;
7    let fixture: ComponentFixture<AddCourseComponent>;
8
9    beforeEach(async () => {
10      await TestBed.configureTestingModule({
11        declarations: [ AddCourseComponent ]
12      })
13        .compileComponents();
14    });
15
16    beforeEach(() => {
17      fixture = TestBed.createComponent(AddCourseComponent);
18      component = fixture.componentInstance;
19      fixture.detectChanges();
20    });
21
22    it('should create', () => {
23      expect(component).toBeTruthy();
24    });
25  });
26
```

TS add-course.component.ts 2 X

app > add-course > TS add-course.component.ts > ...

```
1  import { Component, OnInit } from '@angular/core';
2  import { Router } from '@angular/router';
3  import { DataService } from '../services/data.service';
4  import { Course } from '../shared/course';
5
6  @Component({
7    selector: 'app-add-course',
8    templateUrl: './add-course.component.html',
9    styleUrls: ['./add-course.component.scss']
10 })
11 export class AddCourseComponent implements OnInit {
12
13   addCourseAtt: Course = {
14     courseId: 0,
15     name: '',
16     duration: '',
17     description: '',
18   };
19
20   constructor(private dataService: DataService, private router: Router ) { }
21
22   ngOnInit(): void {
23   }
24
25   addCourse(){
26     this.dataService.addCourse(this.addCourseAtt).subscribe({
27       next: (course) => {
28         this.router.navigate(['courses'])
29         console.log(course)
30       }
31     });
32   }
33
34
35
36   cancel(){
37     this.router.navigate(["courses"]);
38   }
39 }
```

Displaying

```
courses.component.html X
app > course > courses.component.html > div.card-container
1
2 <h1>Course Listing</h1>
3
4 <div class="card-container">
5   <div class="card" *ngFor="let course of courses">
6     <div class="card-header">
7       <h2>{{ course.name }}</h2>
8       <p>Duration: {{course.duration}}</p>
9     </div>
10    <div class="card-body">
11      <p>Description: {{course.description}}</p>
12    </div>
13    <div class="card-footer">
14      <button class="btn-edit" [routerLink]="['/editCourses', course.courseId]">Edit</button>
15      <button class="btn-delete" (click)="deleteCourse(course.courseId)">Delete</button>
16    </div>
17  </div>
18 </div>
19
20
21
22
```

```
TS courses.component.ts 2 X
app > course > TS courses.component.ts > ...
12 export class CoursesComponent implements OnInit {
13
14     courses: Course[] = []
15
16     constructor(private dataService: DataService, private router: Router) { }
17
18     ngOnInit(): void {
19         this.GetCourses()
20
21         console.log(this.courses)
22     }
23
24     GetCourses()
25     {
26         this.dataService.GetCourses().subscribe(result => {
27             let courseList: any[] = result
28             courseList.forEach((element) => {
29                 this.courses.push(element)
30             });
31             this.courses.reverse();
32         })
33     }
34
35     deleteCourse(id: number){
36         this.dataService.deleteCourse(id).subscribe({
37             next: (response) => {
38                 alert("Deleted");
39                 // this.GetCourses();
40                 window.location.reload();
41             }
42         });
43     }
44
45     cancel(){
46         this.router.navigate(["courses"]);
47     };
48 }
49
```

Editing

edit-course.component.html X

> > edit-course > edit-course.component.html > div.container

```
1  <div class="container">
2    <h2>Edit Course</h2>
3
4    <div class="row">
5      <div class="col-6">
6        <form #form="ngForm" (ngSubmit)="updateCourse()">
7
8          <div class="row mb-3">
9            <label for="name" class="col-sm-2 col-form-label">Name</label>
10
11            <input style="border-color: #darkgray;" type="text" class="form-control" id="name" name="name" [(ngModel)] = "courseAtt
12
13          </div>
14
15          <div class="row mb-3">
16            <label for="duration" class="col-sm-2 col-form-label">Duration</label>
17
18            <input style="border-color: #darkgray;" type="text" class="form-control" id="duration" name="duration" [(ngModel)] = "c
19
20          </div>
21
22          <div class="row mb-3">
23            <label for="description" class="col-sm-2 col-form-label">Description</label>
24
25            <input style="border-color: #darkgray;" type="text" class="form-control" id="description" name="description" [(ngModel)
26
27          </div>
28
29          <button type="submit" class="btn-save">Save</button>
30          <button type="button" class="btn-cancel" (click)="cancel()">Cancel</button>
31        </form>
32      </div>
33    </div>
34  </div>
```

<> edit-course.component.html

TS edit-course.component.spec.ts 6 X

app > edit-course > TS edit-course.component.spec.ts > ...

```
1  import { ComponentFixture, TestBed } from '@angular/core/testing';
2  ⚡
3  import { EditCourseComponent } from './edit-course.component';
4
5  describe('EditCourseComponent', () => {
6    let component: EditCourseComponent;
7    let fixture: ComponentFixture<EditCourseComponent>;
8
9    beforeEach(async () => {
10     await TestBed.configureTestingModule({
11       declarations: [ EditCourseComponent ]
12     })
13     .compileComponents();
14   });
15
16   beforeEach(() => {
17     fixture = TestBed.createComponent(EditCourseComponent);
18     component = fixture.componentInstance;
19     fixture.detectChanges();
20   });
21
22   it('should create', () => {
23     expect(component).toBeTruthy();
24   });
25 });
26
```

```

TS edit-course.component.ts 2 X
app > edit-course > TS edit-course.component.ts > ...
1 import { Component, OnInit } from '@angular/core';
2 import { ActivatedRoute, Router } from '@angular/router';
3 import { DataService } from '../services/data.service';
4 import { Course } from '../shared/course';
5
6 @Component({
7   selector: 'app-edit-course',
8   templateUrl: './edit-course.component.html',
9   styleUrls: ['./edit-course.component.scss']
10 })
11 export class EditCourseComponent implements OnInit {
12
13   courseAtt: Course = {
14     courseId: 0,
15     name: '',
16     duration: '',
17     description: ''
18   };
19
20
21   constructor(private route: ActivatedRoute, private dataService: DataService, private router: Router ) { }
22
23   ngOnInit(): void {
24     this.route.paramMap.subscribe({
25       next: (params) => {
26         const courseId = params.get('courseId');
27
28         //Call the API
29         if(courseId){
30           this.dataService.getCourseId(courseId).subscribe({
31             next: (response) => {
32               this.courseAtt = response;
33             }
34           });
35         }
36       }
37     })
38   }
39 }

```

## NavigationBar



```
app.component.html X
app > app.component.html > nav.navbar.navbar-expand-lg.navbar-dark.bg-dark
1 <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
2   <div class="container-fluid">
3     <a class="navbar-brand" href="#">INF 354 Assignment 1</a>
4     <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav"
5       <span class="navbar-toggler-icon"></span>
6     </button>
7     <div class="collapse navbar-collapse" id="navbarNav">
8       <ul class="navbar-nav">
9         <li class="nav-item">
10          <h5><a class="nav-link active" aria-current="page" routerLink="courses">Courses</a></h5>
11        </li>
12        <li class="nav-item">
13          <h5><a class="nav-link" routerLink="addCourses">Add Course</a></h5>
14        </li>
15      </ul>
16    </div>
17  </div>
18 </nav>
19 </nav>
20
21 <router-outlet></router-outlet>
22
```