favorites.ts :

import { Imovie } from './movie';

export interface Ifavorites {

    id: number,

    user\_id: number,

    movies: Imovie

}

movie.ts :

export interface Imovie {

    id: number,

    name: string,

    boxOffice: number,

    isActive: boolean,

    date: Date,

    genre: string,

    hasTeaser: boolean

}

**componenets**

header.ts :

import { Component } from '@angular/core';

@Component({

    selector: 'app-header',

    templateUrl: './header.component.html'

})

export class HeaderComponent {

}

header.html :

<nav class="navbar navbar-expand-lg navbar-dark bg-primary">

    <div class="container-fluid">

        <a class="navbar-brand" href="#">Movie Cruiser</a>

        <button class="navbar-toggler" type="button" data-bs-toggle="collapse"

            data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false"

            aria-label="Toggle navigation">

            <span class="navbar-toggler-icon"></span>

        </button>

        <div class="collapse navbar-collapse justify-content-end" id="navbarSupportedContent">

            <ul class="navbar-nav ml-auto mb-2 mb-lg-0">

                <li class="nav-item"><a class="nav-link active" aria-current="page" href="movie-list">Movies</a>

                </li>

                <li class="nav-item"><a class="nav-link active" aria-current="page" href="login">Login</a></li>

                <li class="nav-item"><a class="nav-link active" aria-current="page" href="signup">Signup</a></li>

            </ul>

        </div>

    </div>

</nav>

movie-list.ts :

import { Component, OnInit } from '@angular/core';

import { Imovie } from '../movie';

import { MovieListService } from '../movie-list.service';

@Component({

  selector: 'app-movie-list',

  templateUrl: './movie-list.component.html',

  styleUrls: ['./movie-list.component.css']

})

export class MovieListComponent implements OnInit {

  public movies:Imovie[];

  constructor(private movieService: MovieListService) { }

  ngOnInit(): void {

    this.movieService.getMovieList().subscribe(data => {

      this.movies = data;

      console.log(data);

    });

  }

}

movie-list.html :

<app-header></app-header>

<body>

    <!------------End of header------------>

    <div>

        <h2 class="display-5">Movies</h2>

    </div>

    <div class="container">

        <table class="table">

            <tr>

                <th id="title">Title</th>

                <th id="box-office">Box Office</th>

                <th id="genre">Genre</th>

                <th id="has-teaser">Has Teaser</th>

            </tr>

            <tr \*ngFor="let movie of movies">

                <td><label for="title">{{movie.name}}</label></td>

                <td><label for="box-office">{{movie.boxOffice}}</label></td>

                <td><label for="genre">{{movie.genre}}</label></td>

                <td><label for="has-teaser">{{movie.hasTeaser}}</label></td>

            </tr>

        </table>

    </div>

</body>

movie-list-customer.ts :

import { Component, OnInit } from '@angular/core';

import { Imovie } from '../movie';

import { MovieListService } from '../movie-list.service';

@Component({

  selector: 'app-movie-list-customer',

  templateUrl: './movie-list-customer.component.html',

  styleUrls: ['./movie-list-customer.component.css']

})

export class MovieListCustomerComponent implements OnInit {

  message:any;

  public movies:Imovie;

  constructor(private movieService: MovieListService) { }

  add(movie) {

    this.movieService.addToFavorite(movie).subscribe(

      data => {

        this.message = data;

        console.log(this.message);

      }

    )

  }

  ngOnInit(): void {

    this.movieService.getMovieList().subscribe(data => {

      this.movies = data;

      console.log(data);

    });

  }

}

movie-list-customer.html :

<nav class="navbar navbar-expand-lg navbar-dark bg-primary">

    <div class="container-fluid">

        <a class="navbar-brand" href="#">Movie Cruiser</a>

        <button class="navbar-toggler" type="button"

            data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"

            aria-controls="navbarSupportedContent" aria-expanded="false"

            aria-label="Toggle navigation">

            <span class="navbar-toggler-icon"></span>

        </button>

        <div class="collapse navbar-collapse justify-content-end"

            id="navbarSupportedContent">

            <ul class="navbar-nav ml-auto mb-2 mb-lg-0">

                <li class="nav-item"><a class="nav-link active"

                    aria-current="page" href="movie-list-customer">Movies</a></li>

                <li class="nav-item"><a class="nav-link active"

                    href="favorites">Favorites</a></li>

                <li class="nav-item"></li>

                <li class="nav-item"><a class="nav-link active" href="#"><i class="bi bi-person-circle" style="color:white; padding:1vh"></i></a></li>

                <li class="nav-item"><a class="nav-link active"

                    href="movie-list">Logout</a></li>

            </ul>

        </div>

    </div>

</nav>

<div>

    <h2 class="display-5">Movies</h2>

</div>

<div class="container">

    <table class="table">

        <tr>

            <th id="title">Title</th>

            <th id="box-office">Box Office</th>

            <th id="genre">Genre</th>

            <th id="has-teaser">Has Teaser</th>

            <th id="action">Action</th>

        </tr>

        <tr \*ngFor="let movie of movies">

            <td><label for="title">{{movie.name}}</label></td>

            <td><label for="box-office">{{movie.boxOffice}}</label></td>

            <td><label for="genre">{{movie.genre}}</label></td>

            <td><label for="has-teaser">{{movie.hasTeaser}}</label></td>

            <td><label for="action"><button

                (click)="add(movie)" class="btn btn-info">Add

                    to Favorite</button></label></td>

        </tr>

    </table>

</div>

movie-list-admin.ts :

import { Component, OnInit } from '@angular/core';

import { Router } from '@angular/router';

import { Imovie } from '../movie';

import { MovieListService } from '../movie-list.service';

@Component({

  selector: 'app-movie-list-admin',

  templateUrl: './movie-list-admin.component.html',

  styleUrls: ['./movie-list-admin.component.css']

})

export class MovieListAdminComponent implements OnInit {

  public movies:Imovie;

  constructor(private movieService: MovieListService, private router: Router) { }

  ngOnInit(): void {

    this.movieService.getMovieList().subscribe(data => {

      this.movies = data;

      console.log(data);

    });

  }

}

movie-list-admin.html :

<nav class="navbar navbar-expand-lg navbar-dark bg-primary">

    <div class="container-fluid">

        <a class="navbar-brand" href="#">Movie Cruiser</a>

        <button class="navbar-toggler" type="button"

            data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"

            aria-controls="navbarSupportedContent" aria-expanded="false"

            aria-label="Toggle navigation">

            <span class="navbar-toggler-icon"></span>

        </button>

        <div class="collapse navbar-collapse justify-content-end"

            id="navbarSupportedContent">

            <ul class="navbar-nav ml-auto mb-2 mb-lg-0">

                <li class="nav-item"><a class="nav-link active"

                    aria-current="page" href="movie-list-customer">Movies</a></li>

                <li class="nav-item"><a class="nav-link active" href="#"><i

                        class="bi bi-person-circle" style="color: white; padding: 1vh"></i>

                        Admin</a></li>

                <li class="nav-item"><a class="nav-link active"

                    aria-current="page" href="movie-list">Logout</a></li>

            </ul>

        </div>

    </div>

</nav>

<div>

    <h2 class="display-5">Movies</h2>

</div>

<div class="container">

    <table class="table">

        <tr>

            <th id="title">Title</th>

            <th id="box-office">Box Office</th>

            <th id="genre">Genre</th>

            <th id="has-teaser">Has Teaser</th>

            <th id="action">Action</th>

        </tr>

        <tr \*ngFor="let movie of movies">

            <td><label for="title">{{movie.name}}</label></td>

            <td><label for="box-office">{{movie.boxOffice}}</label></td>

            <td><label for="genre">{{movie.genre}}</label></td>

            <td><label for="has-teaser">{{movie.hasTeaser}}</label></td>

            <td><label for="action"><a [routerLink]="['/edit', movie.id]"

                    class="btn btn-warning">Edit</a></label></td>

        </tr>

    </table>

</div>

login.ts :

import { Component, OnInit } from '@angular/core';

import { Router } from "@angular/router";

@Component({

  selector: 'app-login',

  templateUrl: './login.component.html',

  styleUrls: ['./login.component.css']

})

export class LoginComponent implements OnInit {

  username = "";

  password = "";

  submit() {

    this.router.navigateByUrl('movie-list-customer');

  }

  constructor(private router: Router) { }

  ngOnInit(): void {

  }

}

login.html :

<app-header></app-header>

    <div class="container-fluid">

        <p class="display-5">Login</p>

        <div class="container-fluid">

            <form name="login-submit" id="form" (ngSubmit)="submit()">

                <div id="login-error">Invalid Username/Password!</div>

                <div class="mb-3">

                    <label for="username" id="username" class="form-label">Username</label>

                    <input type="text" id="username" name="username" [(ngModel)]="username" placeholder="Enter Username" class="form-control">

                </div>

                <div class="mb-3">

                    <label for="pass-login" class="form-label">Password</label> <input type="password" id="pass-login"

                        name="password" [(ngModel)]="password" placeholder="Enter Username" class="form-control">

                </div>

                <div class="mb-3">

                    <button type="submit" class="btn btn-primary mb-3">Login</button>

                    <p>

                        New here? <a href="signup">Sign up</a>

                    </p>

                    <p id="admin-login">

                        Login as <a href="admin-login">Admin?</a>

                    </p>

                </div>

            </form>

        </div>

favorites.ts :

import { Component, OnInit } from '@angular/core';

import { Ifavorites } from '../favorites';

import { FavoritesServiceService } from '../favorites-service.service';

@Component({

  selector: 'app-favorites',

  templateUrl: './favorites.component.html',

  styleUrls: ['./favorites.component.css']

})

export class FavoritesComponent implements OnInit {

  public favorites: Ifavorites;

  constructor(private favoritesService: FavoritesServiceService) { }

  delete(id: number) {

    console.log(id);

    this.favoritesService.deleteFavorite(id).subscribe(

      data => {

        // this.favorites = data;

        this.onInIt();

        console.log(data);

      }

    )

  }

  onInIt() {

    this.favoritesService.getFavoritesList().subscribe(

      data => {

        this.favorites = data;

        console.log(data);

      }

    );

  }

  ngOnInit(): void {

    this.onInIt();

  }

}

favorites.html :

<nav class="navbar navbar-expand-lg navbar-dark bg-primary">

    <div class="container-fluid">

        <a class="navbar-brand" href="#">Movie Cruiser</a>

        <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"

            aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">

            <span class="navbar-toggler-icon"></span>

        </button>

        <div class="collapse navbar-collapse justify-content-end" id="navbarSupportedContent">

            <ul class="navbar-nav ml-auto mb-2 mb-lg-0">

                <li class="nav-item"><a class="nav-link active" aria-current="page"

                        href="movie-list-customer">Movies</a></li>

                <li class="nav-item"><a class="nav-link active" href="favorites">Favorites</a></li>

                <li class="nav-item"></li>

                <li class="nav-item"><a class="nav-link active" href="#"><i class="bi bi-person-circle"

                            style="color:white; padding:1vh"></i></a></li>

                <li class="nav-item"><a class="nav-link active" href="movie-list">Logout</a></li>

            </ul>

        </div>

    </div>

</nav>

<div class="main-heading">

    <p>Movies</p>

</div>

<div class="container">

    <table class="table" id="favorite-table">

        <tr>

            <th id="title">Title</th>

            <th id="box-office">Box Office</th>

            <th id="genre">Genre</th>

            <th id="action"></th>

        </tr>

        <tr \*ngFor="let fav of favorites">

            <td><label for="title">{{ fav.movies.name }}</label></td>

            <td><label for="genre">{{ fav.movies.genre }}</label></td>

            <td><label for="action"><button (click)="delete(fav.id)"

                        class="btn btn-danger">Delete</button></label></td>

        </tr>

    </table>

</div>

edit.ts :

import { Component, OnInit } from '@angular/core';

import { ActivatedRoute } from '@angular/router';

import { Imovie } from '../movie';

import { MovieListService } from '../movie-list.service';

@Component({

  selector: 'app-edit',

  templateUrl: './edit.component.html',

  styleUrls: ['./edit.component.css']

})

export class EditComponent implements OnInit {

  public movie: any;

  constructor(private route: ActivatedRoute, private movieService: MovieListService) { }

  edit(movie) {

    this.movieService.editMovie(movie).subscribe(

      data => {

        console.log(data);

      }

    )

  }

  ngOnInit(): void {

    const id = this.route.snapshot.paramMap.get('id');

    this.movieService.getMovieById(parseInt(id)).subscribe(

      data => {

        this.movie = data;

        console.log(this.movie);

      }

    )

  }

}

edit.html :

<nav class="navbar navbar-expand-lg navbar-dark bg-primary">

    <div class="container-fluid">

        <a class="navbar-brand" href="#">Movie Cruiser</a>

        <button class="navbar-toggler" type="button"

            data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"

            aria-controls="navbarSupportedContent" aria-expanded="false"

            aria-label="Toggle navigation">

            <span class="navbar-toggler-icon"></span>

        </button>

        <div class="collapse navbar-collapse justify-content-end"

            id="navbarSupportedContent">

            <ul class="navbar-nav ml-auto mb-2 mb-lg-0">

                <li class="nav-item"><a class="nav-link active"

                    aria-current="page" href="movie-list-customer">Movies</a></li>

                <li class="nav-item"><a class="nav-link active" href="#"><i

                        class="bi bi-person-circle" style="color: white; padding: 1vh"></i>

                        Admin</a></li>

                <li class="nav-item"><a class="nav-link active"

                    aria-current="page" href="movie-list">Logout</a></li>

            </ul>

        </div>

    </div>

</nav>

<div>

    <p class="display-5">Edit Movie</p>

    <form class="edit-movie-container">

        <div class="mb-3">

            <label for="title" class="form-label">Title</label> <input type="text" name="title"

                id="title" class="form-control" [(ngModel)]="movie.name"> <label for="id" class="form-label">ID</label>

            <input type="text" name="id" id="title" class="form-control" [(ngModel)]="movie.id">

        </div>

        <div class="mb-3">

            <div class="second-row-elements">

                <label for="gross" class="form-label">Gross ($)</label> <input type="text" id="gross"

                    name="gross" class="form-control" [(ngModel)]="movie.boxoffice">

            </div>

            <div class="second-row-elements">

                <label for="active" class="form-label">Active</label>

                <div id="active">

                    <label for="yes" class="form-label">Yes</label> <input type="radio" id="yes" >

                    <label for="no" class="form-label">No</label> <input type="radio" id="no" >

                </div>

            </div>

            <div class="second-row-elements">

                <label for="dol" class="form-label">Date of Launch</label> <input type="date"

                    id="dol" name="dol" class="form-control" [(ngModel)]="movie.date">

            </div>

            <div class="second-row-elements">

                <label for="genre" class="form-label">Genre</label> <select name="genre" id="genre">

                    <option [(ngModel)]="movie.genre"></option>

                    <option value="Science Fiction">Science Fiction</option>

                    <option value="Romance">Romance</option>

                    <option value="Comedy">Comedy</option>

                    <option value="Adventure">Adventure</option>

                    <option value="Thriller">Thriller</option>

                </select>

            </div>

        </div>

        <div class="mb-3">

            <div>

                <label for="has-teaser" class="form-label">Has Teaser</label> <input type="checkbox"

                    id="has-teaser" >

            </div>

        </div>

        <div class="mb-3">

            <button type="submit" id="save" (click)="edit(movie)" class="btn btn-primary">Save</button>

        </div>

    </form>

</div>

movie-service.ts :

import { HttpClient } from '@angular/common/http';

import { Injectable } from '@angular/core';

import { Observable } from 'rxjs';

import { Imovie } from './movie';

@Injectable({

  providedIn: 'root'

})

export class MovieListService {

  getMovieList(): Observable<any> {

    return this.http.get<any>('http://localhost:8080/movie-list').pipe();

  }

  addToFavorite(movie) {

    const addUrl = 'http://localhost:8080/add-favorite';

    return this.http.post(addUrl, movie, {responseType: 'text' as 'json'});

  }

  getMovieById(id) {

    const url = 'http://localhost:8080/edit/' + id

    return this.http.get(url);

  }

  editMovie(movie) {

    const url = 'http://localhost:8080/edit-success';

    return this.http.put(url, movie, {responseType: 'text' as 'json'})

  }

  constructor(private http: HttpClient) { }

}

favorite-service.ts :

import { HttpClient } from '@angular/common/http';

import { Injectable } from '@angular/core';

import { Observable } from 'rxjs';

@Injectable({

  providedIn: 'root'

})

export class FavoritesServiceService {

  getFavoritesList(): Observable<any> {

    return this.http.get<any>('http://localhost:8080/favorites').pipe();

  }

  deleteFavorite(id) {

    const deleteUrl = 'http://localhost:8080/favorites-delete/' + id;

    return this.http.delete(deleteUrl);

  }

  constructor(private http: HttpClient) { }

}

app-routing-module :

import { NgModule } from '@angular/core';

import { RouterModule, Routes } from '@angular/router';

import { EditComponent } from './edit/edit.component';

import { FavoritesComponent } from './favorites/favorites.component';

import { LoginComponent } from './login/login.component';

import { MovieListAdminComponent } from './movie-list-admin/movie-list-admin.component';

import { MovieListCustomerComponent } from './movie-list-customer/movie-list-customer.component';

import { MovieListComponent } from './movie-list/movie-list.component';

const routes: Routes = [

  {path: 'movie-list', component: MovieListComponent},

  {path: 'login', component: LoginComponent},

  {path: 'movie-list-customer', component: MovieListCustomerComponent},

  {path: 'favorites', component: FavoritesComponent},

  {path: 'movie-list-admin', component: MovieListAdminComponent},

  {path: 'edit/:id', component: EditComponent}

];

@NgModule({

  imports: [RouterModule.forRoot(routes)],

  exports: [RouterModule]

})

export class AppRoutingModule { }

app-component.html :

<router-outlet></router-outlet>

<!-- <app-movie-list></app-movie-list> -->

<footer>

    <p class="h6">Copyright @ 2020</p>

</footer>

app-module.ts :

import { NgModule } from '@angular/core';

import { BrowserModule } from '@angular/platform-browser';

import { HttpClientModule } from '@angular/common/http';

import { AppRoutingModule } from './app-routing.module';

import { AppComponent } from './app.component';

import { HeaderComponent } from './header/header.component';

import { RecipesComponent } from './recipes/recipes.component';

import { RecipeListComponent } from './recipes/recipe-list/recipe-list.component';

import { RecipeDetailComponent } from './recipes/recipe-detail/recipe-detail.component';

import { RecipeItemComponent } from './recipes/recipe-list/recipe-item/recipe-item.component';

import { ShoppingListComponent } from './shopping-list/shopping-list.component';

import { ShoppingEditComponent } from './shopping-list/shopping-edit/shopping-edit.component';

import { MovieListComponent } from './movie-list/movie-list.component';

import { LoginComponent } from './login/login.component';

import { FormsModule } from '@angular/forms';

import { MovieListCustomerComponent } from './movie-list-customer/movie-list-customer.component';

import { FavoritesComponent } from './favorites/favorites.component';

import { MovieListAdminComponent } from './movie-list-admin/movie-list-admin.component';

import { EditComponent } from './edit/edit.component';

@NgModule({

  declarations: [

    AppComponent,

    HeaderComponent,

    RecipesComponent,

    RecipeListComponent,

    RecipeDetailComponent,

    RecipeItemComponent,

    ShoppingListComponent,

    ShoppingEditComponent,

    MovieListComponent,

    LoginComponent,

    MovieListCustomerComponent,

    FavoritesComponent,

    MovieListAdminComponent,

    EditComponent

  ],

  imports: [

    BrowserModule,

    AppRoutingModule,

    HttpClientModule,

    FormsModule

  ],

  providers: [],

  bootstrap: [AppComponent]

})

export class AppModule { }

angular.json :

{

  "$schema": "./node\_modules/@angular/cli/lib/config/schema.json",

  "version": 1,

  "newProjectRoot": "projects",

  "projects": {

    "SampleProject": {

      "projectType": "application",

      "schematics": {},

      "root": "",

      "sourceRoot": "src",

      "prefix": "app",

      "architect": {

        "build": {

          "builder": "@angular-devkit/build-angular:browser",

          "options": {

            "outputPath": "dist/SampleProject",

            "index": "src/index.html",

            "main": "src/main.ts",

            "polyfills": "src/polyfills.ts",

            "tsConfig": "tsconfig.app.json",

            "aot": true,

            "assets": [

              "src/favicon.ico",

              "src/assets"

            ],

            "styles": [

              "node\_modules/bootstrap/dist/css/bootstrap.min.css",

              "src/styles.css"

            ],

            "scripts": []

          },

          "configurations": {

            "production": {

              "fileReplacements": [

                {

                  "replace": "src/environments/environment.ts",

                  "with": "src/environments/environment.prod.ts"

                }

              ],

              "optimization": true,

              "outputHashing": "all",

              "sourceMap": false,

              "namedChunks": false,

              "extractLicenses": true,

              "vendorChunk": false,

              "buildOptimizer": true,

              "budgets": [

                {

                  "type": "initial",

                  "maximumWarning": "2mb",

                  "maximumError": "5mb"

                },

                {

                  "type": "anyComponentStyle",

                  "maximumWarning": "6kb",

                  "maximumError": "10kb"

                }

              ]

            }

          }

        },

        "serve": {

          "builder": "@angular-devkit/build-angular:dev-server",

          "options": {

            "browserTarget": "SampleProject:build"

          },

          "configurations": {

            "production": {

              "browserTarget": "SampleProject:build:production"

            }

          }

        },

        "extract-i18n": {

          "builder": "@angular-devkit/build-angular:extract-i18n",

          "options": {

            "browserTarget": "SampleProject:build"

          }

        },

        "test": {

          "builder": "@angular-devkit/build-angular:karma",

          "options": {

            "main": "src/test.ts",

            "polyfills": "src/polyfills.ts",

            "tsConfig": "tsconfig.spec.json",

            "karmaConfig": "karma.conf.js",

            "assets": [

              "src/favicon.ico",

              "src/assets"

            ],

            "styles": [

              "src/styles.css"

            ],

            "scripts": []

          }

        },

        "lint": {

          "builder": "@angular-devkit/build-angular:tslint",

          "options": {

            "tsConfig": [

              "tsconfig.app.json",

              "tsconfig.spec.json",

              "e2e/tsconfig.json"

            ],

            "exclude": [

              "\*\*/node\_modules/\*\*"

            ]

          }

        },

        "e2e": {

          "builder": "@angular-devkit/build-angular:protractor",

          "options": {

            "protractorConfig": "e2e/protractor.conf.js",

            "devServerTarget": "SampleProject:serve"

          },

          "configurations": {

            "production": {

              "devServerTarget": "SampleProject:serve:production"

            }

          }

        }

      }

    }

  },

  "defaultProject": "SampleProject"

}