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

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

import { MatDialog } from '@angular/material/dialog';

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

import { map } from 'rxjs/operators';

import { ImageProcessingService } from '../image-processing.service';

import { MovieTicket } from '../movieticket.model';

import { ShowMoviePosterDialogComponent } from '../show-movie-poster-dialog/show-movie-poster-dialog.component';

import { TicketService } from '../ticket.service';

@Component({

selector: 'app-show-all-tickets',

templateUrl: './show-all-tickets.component.html',

styleUrls: ['./show-all-tickets.component.css']

})

export class ShowAllTicketsComponent implements OnInit {

// displayedColumns: string[] =['Id', 'Movie Title','Movie Genre','Main actors','Director','Movie Original Language','Actual Ticket Price','Discounted Ticket Price'];

ticketDetails = [];

constructor(

private ticketService:TicketService,

public posterDialog: MatDialog,

private imageProcessing: ImageProcessingService,

private router:Router) { }

ngOnInit(): void {

this.getAllTickets();

}

getAllTickets()

{

this.ticketService.getAllTickets()

.pipe(

map((X: MovieTicket[], i) => X.map((ticket:MovieTicket) => this.imageProcessing.createImages(ticket)))

)

.subscribe(data=>{

this.ticketDetails = data;

});

}

public deleteTicket(ticketId){

console.log(ticketId);

this.ticketService.deleteTicket(ticketId).subscribe((resp)=>{

console.log(resp);

this.getAllTickets();

},

(error:HttpErrorResponse)=>{

console.log(error);

})

}

showPoster(movieticket:MovieTicket){

console.log(movieticket);

this.posterDialog.open(ShowMoviePosterDialogComponent, {

data: {

images: movieticket.moviePoster

},

height: '500px',

width: '800px'

} );

}

editTicket(ticketId){

console.log(ticketId);

this.router.navigate(['/addNewTicket',{ticketId:ticketId}]); //now this ticketId will be seen in the url of addNewTicket component page but for fetching the relative data from backend we need a service --> resolver

}

// public getAllTickets(){

// this.ticketService.getAllTickets().subscribe(

// (resp :MovieTicket[])=>{

// console.log(resp);

// this.ticketDetails = resp;

// },(error:HttpErrorResponse) =>{

// console.log(error);

// }

// );

// }

}