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

import { ActivatedRouteSnapshot, Resolve, RouterStateSnapshot } from '@angular/router';

import { Observable } from 'rxjs';

import { map } from 'rxjs/operators';

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

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

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

@Injectable({

providedIn: 'root'

})

export class BuyTicketResolverService implements Resolve<MovieTicket[]> {

constructor(

private ticketService:TicketService,

private imageProcessing:ImageProcessingService

) { }

resolve(route: ActivatedRouteSnapshot, state: RouterStateSnapshot):MovieTicket[]| Observable <MovieTicket[]>|Promise<MovieTicket[]>{

const id = route.paramMap.get("id");

const isSingleTicketCheckout = route.paramMap.get("isSingleTicketCheckout");

return this.ticketService.getTicketDetails(isSingleTicketCheckout, id)

.pipe(

map(

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

)

);

}

}