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

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

import { Observable, of } 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'

})

//Resolve will take input of what type of value is to be resolved

export class TicketResolverService implements Resolve<MovieTicket>{

constructor(

private ticketService:TicketService,

private imageProcessingService:ImageProcessingService

) { }

resolve(route: ActivatedRouteSnapshot, state: RouterStateSnapshot): Observable <MovieTicket> {

//throw new Error('Method not implemented');

const id = route.paramMap.get("ticketId");//pass that value to paramMap that is added to the addNewTicket Url

if(id){

//then fetch data from backend

return this.ticketService.GetTicketById(id).pipe(

map(mt => this.imageProcessingService.createImages(mt))

);

//for processing the stored imges use imageprocessingservice method createImage

//.map takes the values and convert it into desired format

//if id is present-->fetch data from backend -->use imageprocessingservice to onvert image bytes into real image

//now configure this resolver in approuting module.

}else{

//return empty product observabel

return of(this.getTicketDetails()); //of function has to be imported from .rxjs

}

}

getTicketDetails(){

return{

movieName:"",

genre:"",

director:"",

actors:"",

language:"",

description:"",

actualPrice:0,

discountedPrice:0,

moviePoster: []

}

}

}