App Module

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

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

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

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

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

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

import { ProductListComponent } from './products/product-list.component';

import { ConvertToSpacesPipe } from './shared/convert-to-spaces.pipe';

import { StarComponent } from './shared/star.component';

import { ProductDetailComponent } from './products/product-detail.component';

import { WelcomeComponent } from './home/welcome.component';

//import { WelcomeComponent } from app/home/welcome.component';

//import { ProductDetailComponent } from './products/product-detail.component';

@NgModule({

declarations: [

AppComponent,

ProductListComponent,

ConvertToSpacesPipe,

StarComponent,

ProductDetailComponent,

WelcomeComponent,

],

imports: [

BrowserModule,

FormsModule,

HttpClientModule,

RouterModule.forRoot([

{path: 'products', component: ProductListComponent},

{path: 'products/:id', component: ProductDetailComponent},

{path: 'welcome', component: WelcomeComponent},

{path: '', redirectTo: 'welcome', pathMatch: 'full'},

{path: '\*\*', redirectTo: 'welcome', pathMatch: 'full'}

])

// P

],

providers: [],

bootstrap: [AppComponent]

})

export class AppModule { }

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

import { ProductService } from './products/product.service';

@Component({

selector: 'pm-root',

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

styleUrls: ['./app.component.css'],

providers: [ ProductService ]

})

export class AppComponent {

title: string = 'Acme Product Management';

}

APP COMPONENT HTML

<!--The whole content below can be removed with the new code.-->

<div style="text-align:center">

<h1>

Welcome to {{title}}!!

</h1>

... Starter Files ...

<pm-products></pm-products>

</div>

PRODUCT LIST HTML

<div class="panel panel-primary">

<div class ="panel-heading">

{{pageTitle}}

</div>

<div class="panel-body">

<div class ="row">

<div class="col-md-2"> Filter By </div>

<div class="col-md-4">

<input type="text"

[(ngModel)]= 'listFilter' />

</div>

</div>

<div class ="row">

<div class="col-md-6">

<h3> Filtered By : </h3>

</div>

</div>

<div class="table-responsive">

<table class="table"

\*ngIf='products && products.length'>

<thead>

<tr>

<th>

<button class="btn btn-primary"

(click)='toggleImage()'>

{{showImage ? 'Hide' : 'Show' }} Image

</button>

</th>

<th>Product</th>

<th>Code</th>

<th>Available</th>

<th>Price</th>

<th>5 Star Rating</th>

</tr>

</thead>

<tbody>

<tr \*ngFor="let product of filteredProducts">

<td>

<img \*ngIf='showImage'

[src]='product.imageUrl'

[title] ='product.productName'

[style.width.px] = 'imageWidth'

[style.margin.px] = 'imageMargin'>

</td>

<td>{{product.productName}}</td>

<td>{{product.productCode | lowercase | convertTo: '-'}}</td>

<td>{{product.releaseDate}}</td>

<td>{{product.price | currency :'USD':true:'1.2-2'}}</td>

<td><pm-star [rating]='product.starRating'

(ratingClicked)='onRatingClicked($event)'></pm-star></td>

</tr>

</tbody>

</table>

</div>

</div>

</div>

PRODUCT LIST TS

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

import { IProduct } from './product';

import { ProductService } from './product.service';

@Component({

selector: 'pm-products',

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

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

})

export class ProductListComponent implements OnInit {

pageTitle: string = 'Product List';

imageWidth: number = 50;

imageMargin: number = 2;

showImage: boolean = false;

\_listFilter: string;

errorMessage: string;

filteredProducts: IProduct[];

products: IProduct[] = [];

private \_productService;

constructor(productService: ProductService) {

this.\_productService = productService;

// moved to ngOnit

// this.filteredProducts = this.products;

// this.\_listFilter = 'cart';

}

onRatingClicked(message) {

this.pageTitle = ' Product List ' + message;

}

get listFilter(): string {

return this.\_listFilter;

}

set listFilter(value: string) {

this.\_listFilter = value;

this.filteredProducts = this.listFilter ? this.performFilter(this.listFilter) : this.products;

}

// any can be replaced with interface

performFilter(filterBy: string): IProduct[] {

filterBy = filterBy.toLocaleLowerCase();

return this.products.filter((product: IProduct) =>

product.productName.toLocaleLowerCase().indexOf(filterBy) !== -1);

}

toggleImage() {

this.showImage = !this.showImage;

}

ngOnInit() {

// console.log('Hey there');

this.products = this.\_productService.getProducts().

subscribe(products => {this.products = products;

this.filteredProducts = this.products;

},

error => this.errorMessage = <any>error);

// since the constructor is exexcuted first and before this , product list will be empty

// so we will move filtered productsin ngOnit as well

this.filteredProducts = this.products;

// since no data was coming as explained in notebook , we will have to move this line to subscribe()

}

}

PRODUCT SERVICE

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

import { IProduct } from './product';

import { Observable } from 'rxjs/Observable';

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

//import 'rxjs/add/operator/throw';

import 'rxjs/add/operator/catch';

import 'rxjs/add/operator/do';

@Injectable()

export class ProductService {

private \_productUrl = './api/products/products.json';

constructor(private \_http: HttpClient) {

}

getProducts(): Observable<IProduct[]> {

return this.\_http.get<IProduct[]>(this.\_productUrl)

.do(data => console.log('ALl:' + JSON.stringify(data)))

.catch(this.handleError);

}

private handleError(err: HttpErrorResponse) {

console.log(err.message);

return Observable.throw(err.message);

}

}

PRODUCT

export interface IProduct {

productId: number;

productName: string;

productCode: string;

releaseDate: string;

price: number;

description: string;

starRating: number;

imageUrl: string;

}

// export class Product implements IProduct {

// constructor(public productId: number,

// public productName: string,

// public productCode: string,

// public releaseDate: string,

// public price: number,

// public description: string,

// public starRating: number,

// public imageUrl: string,

// ) {

// }

// }