import { CartItem } from './../models/cart-item';

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

import { BehaviorSubject, Subject } from 'rxjs';

@Injectable({

providedIn: 'root'

})

export class CartService {

cartItems: CartItem[] = [];

totalPrice: Subject<number> = new BehaviorSubject<number>(0);

totalQuantity: Subject<number> = new BehaviorSubject<number>(0);

constructor() { }

addToCart(theCartItem: CartItem){

// check if we already have the item in our cart.

let alreadyExistInCart: boolean = false;

let existingCartItem: CartItem = undefined;

if(this.cartItems.length > 0){

//find the item in the cart based on item id

existingCartItem = this.cartItems.find(tempCartItem => tempCartItem.id === theCartItem.id);

// check if we found it

alreadyExistInCart = (existingCartItem != undefined);

}

if(alreadyExistInCart){

// increment the quantity

existingCartItem.quantity++;

}else{

// just add the item in the array

this.cartItems.push(theCartItem);

}

// compute the totals

this.computeCartTotals();

}

computeCartTotals() {

let totalPriceValue: number =0;

let totalQuantityValue: number =0;

for(let tempCartItem of this.cartItems){

totalPriceValue += tempCartItem.quantity \* tempCartItem.unitPrice;

totalQuantityValue += tempCartItem.quantity;

}

// publish the new values....all subscribers will recieve the new data

this.totalPrice.next(totalPriceValue);

this.totalQuantity.next(totalQuantityValue);

// log the cart data just for debugging purpose

this.logCartData( totalPriceValue , totalQuantityValue);

}

logCartData(totalPriceValue: number, totalQuantityValue: number) {

console.log('-----------the cart content--------')

for(let tempCartItem of this.cartItems){

let subTotalPrice = tempCartItem.unitPrice \* tempCartItem.quantity;

console.log(`Name: ${tempCartItem.name}, quantity: ${tempCartItem.quantity}, price: ${tempCartItem.unitPrice}, subTotalPrice: ${subTotalPrice} `);

console.log(`totalPrice: ${totalPriceValue}, totalQuantity: ${totalQuantityValue}`);

}

}

decrementQuantity(theCartItem: CartItem) {

theCartItem.quantity--;

if(theCartItem.quantity == 0){

this.removeCartItem(theCartItem);

}else{

this.computeCartTotals();

}

}

removeCartItem(theCartItem: CartItem) {

// get the item index in the array

const itemIndex = this.cartItems.findIndex(tempCartItem => tempCartItem.id == theCartItem.id);

// if found, remove the item from the aray at the given index

if(itemIndex > -1){

this.cartItems.splice(itemIndex, 1);

this.computeCartTotals();

}

}

}