



Front-End Essentials

Exam


Document Code	25e-BM/HR/HDCV/FSOFT
Version	1.1
Effective Date	20/11/2012

RECORD OF CHANGES

No	Effective Date	Change Description	Reason	Reviewer	Approver
1	25/Jun/2018	Create a new Exam	Create new	DuongTQ	VinhNV

Contents

Problem: Checkout Cart Application	4
--	---

		CODE: TYPE: LOC: DURATION:	FEE.Exams.Opt2 Long N/A 150 MINUTES
---	---	-------------------------------------	--

Working tools and delivery requirements

- **Working tools:** Visual Studio Code
- **Delivery:** Source code in a compressed archive (zip extension is a must).

Technologies

The product illustrates:

- Front-end Essentials
- Practice to use JavaScript to calculate value

Technical Requirements

- Follow the standard naming and coding convention.
- Add appropriate comments for each class, method, attribute, algorithms, ...
- Create Project named <YOUR ACCOUNT>_fee.exams.opt2 and put every file inside









Example if your account is AnhNV then you must create a project called: **AnhNV_fee.exams.opt2**

Problem: Checkout Cart Application

Given an array of products:

```
var products = [  
  {  
    img: "./assets/t-shirt.png",  
    description: "RRR Logo Minimal Yellow T-Shirt",  
    quantity: 1,  
    price: 30,  
    id: 1,  
  },  
  {  
    img: "./assets/glasses.png",  
    description: "Porgy Backstage Reading Glasses | CADDIS Readers",  
    quantity: 4,  
    price: 5,  
    id: 2,  
  },  
  {  
    img: "./assets/hat.png",  
    description:  
      "The Classic Fedora - Genuine Fine Weave Panama Hat By Pachacuti",  
    quantity: 3,  
    price: 10,  
    id: 3,  
  },  
  {  
    img: "./assets/hoodie.png",  
    description: "High Street Hiphop Printing Sm Hoodie Fg E467 - T2ng",  
    quantity: 2,  
    price: 20,  
    id: 4,  
  },  
];
```

Your task is to build a Checkout Cart Application that looks like below (correct spacing and icons are not required)

My Cart						
#	Product	Description	Quantity	Price	Total	
1		RRR Logo Minimal Yellow T-Shirt	- <input type="text" value="1"/> +	\$30	\$30	
2		Porgy Backstage Reading Glasses CADDIS Readers	- <input type="text" value="4"/> +	\$5	\$20	
3		The Classic Fedora - Genuine Fine Weave Panama Hat By Pachacuti	- <input type="text" value="3"/> +	\$10	\$30	
4		High Street Hiphop Printing Sm Hoodie Fg E467 - T2ng	- <input type="text" value="2"/> +	\$20	\$40	
Total Price:					\$120	

Specification:

1. Must use plain HTML, CSS and JavaScript without utilizing any frameworks or libraries. However, you are allowed to use icons and fonts.
2. The application should correctly render the following information from the provided array: Product Order Number, Image, Description, Quantity, Price, Total, and Total Price using the below formula:

```
Total = Quantity * Price
Total Price = Sum of all Total Values
```

3. Users should be able to change the quantity of a product by clicking the "-" or "+" button. When the quantity is 1, the "-" button should be disabled to prevent invalid quantity values.
4. Users should also be able to change the quantity of a product by manually entering a value in the quantity input box. The Product Total and Total Price should be updated when the user presses the "Enter" key or when they blur (unfocus) the input box.
Preventing invalid values (such as zero, negative, or non-numeric) in the quantity input box is a bonus.
5. User can Delete a Product by clicking the "Trash" icon. Make sure to properly render the order number of each product after a deletion.