

Shopping Cart Exercise

This exercise should be discussed in the class. In this exercise you will practice how to implement a simplified shopping cart by using JavaScript. You will create a shopping cart for an online purchase order system of books. The stock information is provided by an external XML file **books.xml**.

You can find all the required files for this exercise on Moodle.

You can find a solution of this exercise on the course Github repository.

The outcome of this exercise should look similar to the following:

Add Books to the Shopping Cart

Item	Title	Author	Price	Add
0	Empire Burlesque	Bob Dylan	10.90	Add
1	Hide your heart	Bonnie Tyler	19.90	Add
2	Greatest Hits	Dolly Parton	29.90	Add
3	Still got the blues	Gary Moore	50.20	Add
4	Eros	Eros Ramazzotti	35.90	Add
5	One night only	Bee Gees	10.90	Add
6	Sylvias Mother	Dr.Hook	18.10	Add
7	Maggie May	Rod Stewart	28.50	Add
8	Romanza	Andrea Bocelli	12.80	Add
9	When a man loves a woman	Percy Sledge	38.70	Add

Item	Title	Price	Quantity
Total Cost: 0.00			
Clear Cart		Check Out	

Get ready

1. Create or open a website project in your preferred IDE.
2. Create a new HTML file **ShoppingCart.html** and attach the given CSS stylesheet **shopping-cart.css** to the page.
3. Create page elements on **ShoppingCart.html** by using the pre-defined styles. Here is the source code the comments explains the code:

```

<body>
  <h1>Add Books to the Shopping Cart</h1>
  <!-- Shopping cart -->
  <div id="txtContainer">
    <!-- Book Area -->
    <div class="display products" id="txtBookInfo">Books</div>
    <!-- Shopping Cart Area-->
    <div class="display">
      <table>
        <tr>
          <td colspan="2">
            <div id="txtCart">Cart</div>
          </td>
        </tr>
        <tr>
          <td class="right">
            <input type="button" value="Clear Cart" onclick="clearCart();" />
          </td>
          <td class="left">
            <input type="button" value="Check Out" onclick="checkOut();" />
          </td>
        </tr>
      </table>
    </div>
  </div>
</body>

```

4. The outcome of this part should look similar to the following:



:

Load stock information from books.xml

1. Create a new JavaScript file **shopping-cart.js** in your server folder **Shopping Cart Exercise**, and attach it to **ShoppingCart.html**.
2. Inside file **shopping-cart.js**, define function **loadXMLFile(xmlFile)** which takes an XML file as a parameter and converts it into an XML Document object as the return object.

```
function loadXMLFile(xmlFile) {
    xmlhttp = new XMLHttpRequest();

    //Load the XML file and return an XML object
    xmlhttp.open("GET", xmlFile, false);
    xmlhttp.send();
    return (xmlhttp.responseXML);
}
```

3. Inside file **shopping-cart.js**, define two global variables to hold the shopping cart later:

```
let nextItem = 0; //The index of next item to be added
let currentCart = new Array(); //This array contains the current items in the cart
```

4. Inside file **shopping-cart.js**, define function **displayBooks()** which loads the XML file **books.xml** and save the data into an XML Document object **xmlDoc**. It then constructs an HTML table that displays the key information for all the books within **<div id="txtBookInfo">**. The comments explain the code.

```
function displayBooks() {
    //Load the XML file that contains the information for all the books
    xmlDoc = loadXMLFile("books.xml");

    //Construct a table that contains the key information for all the books
    txt = "<table border='0px' id='tblBook' cellpadding='3px' cellspacing='0px'>";
    txt += "<tr><th>Item</th><th>Title</th><th>Author</th><th>Price</th><th>Add</th></tr>";

    //Retrieve all the BOOK elements from the XML file
    books = xmlDoc.documentElement.getElementsByTagName("BOOK");
    for (let i = 0; i < books.length; i++) {
        txt += "<tr>";

        txt += "<td align='left'>" + i + "</td>";
        bookChildren = books[i].getElementsByTagName("TITLE");
        txt += "<td align='left' id='title' " + i + "'>" + bookChildren[0].firstChild.
            nodeValue + "</td>";

        bookChildren = books[i].getElementsByTagName("AUTHOR");
        txt += "<td align='left'>" + bookChildren[0].firstChild.nodeValue + "</td>";

        bookChildren = books[i].getElementsByTagName("PRICE");
        txt += "<td align='left' id='price' " + i + "'>" + bookChildren[0].firstChild.
            nodeValue + "</td>";

        txt += "<td align='right'><img src='book-icon.jpg' /></td>";
        txt += "<td align='right'><button type='button' id='bntAddToCart'
            onclick='addToCart(" + i + ")'>Add to Cart</button>" + "</td>";

        txt = txt + "</tr>";
    }
    txt += "</table>";

    //Display book table in <div id="txtBookInfo">
    document.getElementById("txtBookInfo").innerHTML = txt;
}
```

5. Inside file **shopping-cart.js**, call function **displayBooks()** inside a **window.onload** event function.

```

window.onload = function () {
    displayBooks();
}

```

6. This part of the outcome should be similar to the following, when a book icon is clicked, it will call function **addToCart(i)** with the item number ,i.e. **i**, as a parameter.

Add Books to the Shopping Cart

Item	Title	Author	Price	Add
0	Empire Burlesque	Bob Dylan	10.90	 <input type="button" value="Add to Cart"/>
1	Hide your heart	Bonnie Tyler	19.90	 <input type="button" value="Add to Cart"/>
2	Greatest Hits	Dolly Parton	29.90	 <input type="button" value="Add to Cart"/>
3	Still got the blues	Gary Moore	50.20	 <input type="button" value="Add to Cart"/>
4	Eros	Eros Ramazzotti	35.90	 <input type="button" value="Add to Cart"/>
5	One night only	Bee Gees	10.90	 <input type="button" value="Add to Cart"/>
6	Sylvias Mother	Dr.Hook	18.10	 <input type="button" value="Add to Cart"/>
7	Maggie May	Rod Stewart	28.50	 <input type="button" value="Add to Cart"/>
8	Romanza	Andrea Bocelli	12.80	 <input type="button" value="Add to Cart"/>
9	When a man loves a woman	Percy Sledge	28.70	 <input type="button" value="Add to Cart"/>

Cart

Build the shopping cart

1. Inside file **shopping-cart.js**, define function **displayCart()** which constructs the shopping cart table from the global array **currentCart** and displays it within **<div id="txtCart">**. The array will hold all the selected items within the shopping cart.

```
function displayCart() {
    //Construct shopping cart table heading
    txt = "<table id='tblCart' border='5px' cellpadding='1px' cellspacing='0px' align='center' style='margin:0px;border:#666 solid;'>";
    txt += "<tr><th>Item</th><th>Title</th><th>Price</th><th>Quantity</th></tr>";

    totalCost = 0;

    //Construct shopping cart table body
    for (i = 0; i < nextItem; i++) {
        txt += "<tr>";

        txt += "<td align='left'>" + currentCart[i][0] + "</td>";
        txt += "<td align='left'>" + currentCart[i][1] + "</td>";
        txt += "<td align='left'>" + currentCart[i][2] + "</td>";
        txt += "<td align='right'>" + currentCart[i][3] + "</td>";

        totalCost += currentCart[i][2] * currentCart[i][3];

        txt = txt + "</tr>";
    }

    //Display the total cost
    txt += "<td align='center' colspan='4' style='color:#900;'> Total Cost: " +
    parseFloat(totalCost).toFixed(2) + "</td>";

    //Close shopping cart table
    txt += "</table>";




    //Display shopping cart table in <div id="txtCart">
    document.getElementById("txtCart").innerHTML = txt;
}
```

2. Inside file **shopping-cart.js**, update the **window.onload()** function to also call the **displayCart()** function:

```
window.onload = function () {
    displayBooks();
    displayCart();
}
```

3. The outcome of this part should look similar to the following:

Add Books to the Shopping Cart

Item	Title	Author	Price	Add
0	Empire Burlesque	Bob Dylan	10.90	 <input type="button" value="Add to Cart"/>
1	Hide your heart	Bonnie Tyler	19.90	 <input type="button" value="Add to Cart"/>
2	Greatest Hits	Dolly Parton	29.90	 <input type="button" value="Add to Cart"/>
3	Still got the blues	Gary Moore	50.20	 <input type="button" value="Add to Cart"/>
4	Eros	Eros Ramazzotti	35.90	 <input type="button" value="Add to Cart"/>
5	One night only	Bee Gees	10.90	 <input type="button" value="Add to Cart"/>
6	Sylvias Mother	Dr.Hook	18.10	 <input type="button" value="Add to Cart"/>
7	Maggie May	Rod Stewart	28.50	 <input type="button" value="Add to Cart"/>
8	Romanza	Andrea Bocelli	12.80	 <input type="button" value="Add to Cart"/>
9	When a man loves a woman	Percy Sledge	28.70	 <input type="button" value="Add to Cart"/>

Item	Title	Price	Quantity
Total Cost: 0.00			

- Inside file **shopping-cart.js**, define function **addToCart(selectedItem)** which is called when a book icon is clicked. It checks if the selected item has been added once, if YES then its quantity is increased by one, otherwise, it adds a new item to the shopping cart.

```
function addToCart(selectedItem) {
    let addedIndex = -1;

    //Check if the selected item has been added once
    for (i = 0; i < nextItem; i++) {
        if (currentCart[i][0] == selectedItem) {
            //The selected item has been added once, record the index
            addedIndex = i;
            break; //stop looping
        }
    }

    //If the selected item has not been added, add a new item to the shopping
    //cart, otherwise, simply increase the quantity for the item by one.
    if (addedIndex == -1) {
        //Create a new item in the shopping cart
        currentCart[nextItem] = new Array();

        //Put the item details into the shopping cart
        currentCart[nextItem][0] = selectedItem;
        currentCart[nextItem][1] = document.getElementById("title" + selectedItem).
            innerHTML;
        currentCart[nextItem][2] = document.getElementById("price" + selectedItem).
            innerHTML;
        currentCart[nextItem][3] = 1; //set the quantity

        //Get ready to add next item
        nextItem += 1;
    }
    else {
        currentCart[addedIndex][3] += 1;
    }
    //Refresh the shopping cart
    displayCart();
}
```

5. Try to click on some of the book icons, the outcome should be similar to the following:

Add Books to the Shopping Cart

Item	Title	Author	Price	Add
0	Empire Burlesque	Bob Dylan	10.90	 <input type="button" value="Add to Cart"/>
1	Hide your heart	Bonnie Tyler	19.90	 <input type="button" value="Add to Cart"/>
2	Greatest Hits	Dolly Parton	29.90	 <input type="button" value="Add to Cart"/>
3	Still got the blues	Gary Moore	50.20	 <input type="button" value="Add to Cart"/>
4	Eros	Eros Ramazzotti	35.90	 <input type="button" value="Add to Cart"/>
5	One night only	Bee Gees	10.90	 <input type="button" value="Add to Cart"/>
6	Sylvias Mother	Dr.Hook	18.10	 <input type="button" value="Add to Cart"/>
7	Maggie May	Rod Stewart	28.50	 <input type="button" value="Add to Cart"/>
8	Romanza	Andrea Bocelli	12.80	 <input type="button" value="Add to Cart"/>
9	When a man loves a woman	Percy Sledge	28.70	 <input type="button" value="Add to Cart"/>

Item	Title	Price	Quantity
0	Empire Burlesque	10.90	2
1	Hide your heart	19.90	1
2	Greatest Hits	29.90	1
Total Cost: 71.60			

Handle the two buttons

1. Inside file **shopping-cart.js**, define function **clearCart()** which is called when the **Clear Cart** button is clicked. It clears the current item array and refreshes the shopping cart:

```
function clearCart() {
    //Reinitialise the current item array
    nextItem = 0;
    currentCart = new Array();

    //Refresh the shopping cart
    displayCart();
}
```

2. Inside file **shopping-cart.js**, define function **checkOut()** which is called when the button **Check Out** is clicked. It saves the selected items in the shopping cart into an XML document object and clears the shopping cart. It also calls the given PHP file **ProcessingOrder.php** to save the selected items in an external XML file **item.xml** (when run in an environment with PHP available).

```
function checkOut() {  
  
    //Construct XML string  
    txt = "<ITEMS>";  
  
    //Loop through the shopping cart  
    for (i = 0; i < nextItem; i++) {  
        txt = txt + "<ITEM>";  
        txt = txt + "<TITLE>" + currentCart[i][1] + "</TITLE>";  
        txt = txt + "<PRICE>" + currentCart[i][2] + "</PRICE>";  
        txt = txt + "<QUTANTY>" + currentCart[i][3] + "</QUTANTY>";  
        txt = txt + "</ITEM>";  
    }  
    txt = txt + "</ITEMS>";  
  
    //Save the XML data into an XML object, which could be used by another pice of  
    JavaScript code  
  
    parser = new DOMParser();  
    xmlDoc = parser.parseFromString(txt, "text/xml");  
  
    //XML HTTP object  
    xmlhttp = new XMLHttpRequest();  
  
    //Send the XML string to server  
    let url = "ProcesingOrder.php?XMLStr=" + txt;  
    xmlhttp.open("GET", url, false);  
    xmlhttp.send(null);  
  
    //Clear cart  
    clearCart();  
}
```

Compare your completed exercise with the solution on the course Github repository.