JavaScript Assignment #3

Grading Note: Grading will be based on whether your page does what it is supposed to do, but also that it is written properly with correct HTML 5, CSS, and JavaScript. If any of your assignment parts isn’t working right but you are turning it in anyway, be sure to indicate that it doesn’t work properly. I will discover this anyway, but you might save me a lot of time and keep me in a better grading mood!

**Part 1: What’s your name? Using a Products Form**(Like A2P2 except with NO alerts, NO prompts, and NO document.writes!)

You are given some HTML to start you off on the right path. User fills in the form information (their name at least) and then click the “Process Order” button. All you have to do is produce the person’s name in a sentence at the bottom of the page as shown in the screen shot below. Produce the result by writing a “onclick” event for the “process order” button.

Your starting point for this assignment part is the given HTML below that includes a “form” and a “table”. Copy/paste the code from here into your own editor to begin the process of coding your solution. Fill in the areas of the code that are currently bold as needed to transform this generic page into one for your own company, products, prices, etc. just like you did in A2P2 through A2P4. For example, you don’t sell widgets and thingys, so change that. You can also copy/paste your styles from A2 and plug them into the “head” section here to make your page look nicer.

**=========== A3P1 Starter Code Begins Below ============  
========** Fill in the bolded areas with your own code **=======**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<title>A3P1 by **YOUR NAME HERE**</title>

</head>

<body>

<h1>A3P1 **STORE NAME GOES HERE**</h1>

<p>**STORE INTRODUCTION GOES HERE**</p>

<p>This months specials:</p>

<form>

<table border="1" cellspacing="0" cellpadding="6" summary="Table of product images, prices & descriptions">

<tbody>

<tr>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td>Please enter your name:<br><input type="text" id="txtName" name="txtName">  
 </td>

</tr>

<tr>

<td>**PRODUCT IMAGE #1 GOES HERE**</td>

<td><p style="font-weight: bold">**$1 Widgets**</p>

<p>**PRODUCT #1 DESCRIPTION GOES HERE** </p></td>

<td>Quantity:<br><input type="text" id="txtQ1" name="txtQ1"></td>

</tr>

<tr>

<td>**PRODUCT IMAGE #2 GOES HERE**</td>

<td><p style="font-weight: bold">**$2 Thingys**</p>

<p>**PRODUCT #2 DESCRIPTION GOES HERE** </p></td>

<td>Quantity:<br><input type="text" id="txtQ2" name="txtQ2"></td>

</tr>

</tbody>

</table>

<p>  
<input type="button" value="Process Order" id="btnProcess" onClick="**EVENT CODE HERE**">  
</p>

</form>

<div id="receipt">&nbsp;</div>

</body>

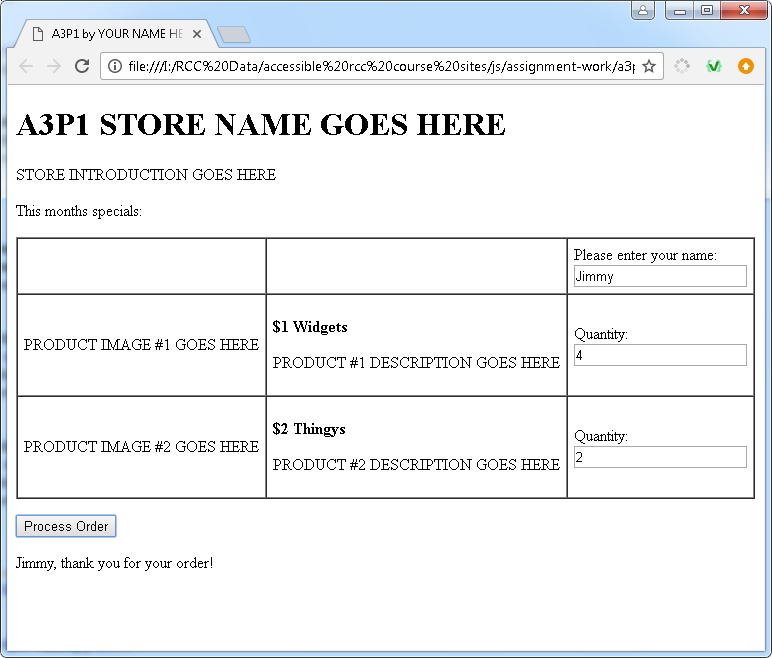
</html>

**========= A3P1 Starter Code Ends Above =============**

When the “Process Order” button is clicked, your code should react by looking into the “name” textbox and getting the name entered by the user from there and displaying it within the already existing but empty “receipt” div. Your code should be written right there within the “onclick” event area of the “button” (HTML "input" tag with type="button"). In your "onclick" event code, you will need to work with the existing receipt div (the "div" tag below the "Process Order" button with the "id" of "receipt"). In order to work with that receipt div, you will need to use the "document.getElementById" method. That way, you can use that div's "innerHTML" property to set the display of the receipt information there instead of using document.write to display it. So the receipt info will appear below the “Process Order” button as shown at the bottom of the screen shot below. The receipt text should match that shown in the screen shot below.

Note that **NO document.writes are allowed to be used to produce this result!**

For this part, do not worry about the information that may have been entered into the quantity text boxes for your two products, just ignore that for now. A later part of this assignment will do something further with those quantities!



**What to turn in:**

1. Into the given assignment 3 “turn it in” Word file, paste a screen shot of your products page in the browser. It should look similar to the one above except for your own product information and images and your name on the browser’s tab (from the HTML “title” tag). The entire page should be visible including the h1, the table with 3 filled in textbox values, the “process order” button, and the summary line that includes the person’s name below the button as shown. Paste your screen shot into the assignment 3 part 1 screen shot area of the document.

2. Into the given assignment 3 “turn it in” Word file, paste a copy of your web page’s code into the part 1 code section of the document. It should contain ALL of your page’s code. Do not paste a screenshot of code, paste the actual text of the code. You can format the font, spacing, or size of the code after pasting into Word to improve its appearance and readability, just be sure not to actually change any of the code itself.

**Part 2: Make it Functional!**   
(Display the person’s name and what they ordered. And do it with a function!)

Also, all product prices should be set as constants in the "head" section. Any displaying of prices or calculating based on prices should use those constants. For example, prices shown within the form should be done with document.write using those constants.

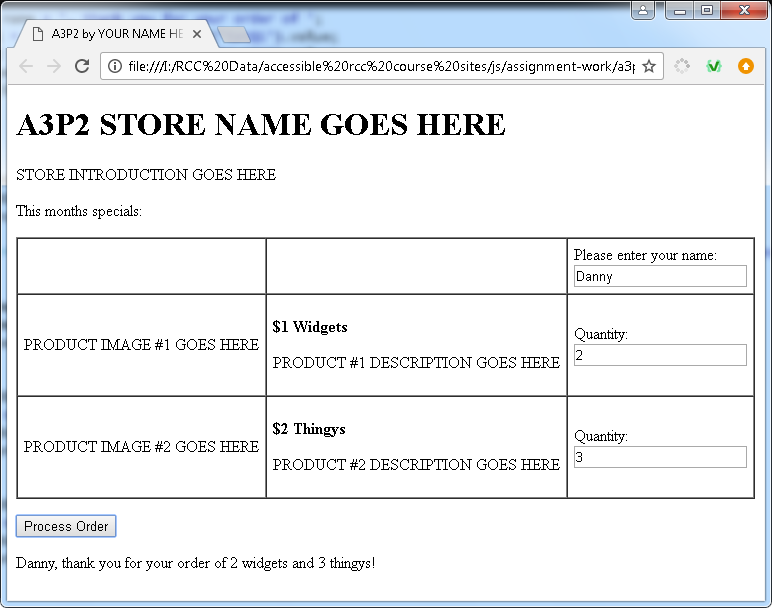
Begin by making a copy of your A3P1 file and call it A3P2. Then you’ll need to create 2 price constants which are declared and given their price values (no decimals) in the “head” area and then use those variables every time a price is needed for display or calculation purposes. In your document.write for the prices displaying in the form area, use document.write to produce as little as possible – only the price itself – no HTML tags or other text characters that can be done outside of the script itself.

Also for this part 2 version, your output at the bottom of the page (the receipt) needs to report the order in sentence format and include the quantities entered by the user for both of your two products. To keep things simpler, you can assume the textbox input contains valid data. See the sample screenshot below to see an example of what your result should look like. But we are not going to put any of this code into the “onclick” event of the “button” itself. Instead, the code from part 1 that processed the text from within the name textbox and the new part 2 code that processes the text from within the 2 quantity textboxes must be located within a function called “processOrder” in the <head> section of the page. In fact, here is what your “button” code **must** look like:

**<p>  
<input type="button" value="Process Order" id="btnProcess"** **onClick="processOrder();">  
</p>**

Note that you will be asked to use an “addEventListener” in the next assignment part but for this part you should use the above “onClick” event on the button itself as shown above.

So copy that “html” paragraph (and “input” tag) above and use it instead of the old “input tag and paragraph” you had for part 1. You need to create a script in the <head> area and define the “processOrder” function there. Inside that function will be where the rest of your work for this part goes. Note that your old part 1 paragraph should have a few JavaScript statements that are still useful and should help you get started on your new function in the <head> because the person’s name still needs to be a part of the receipt. Getting quantities from the other 2 textboxes is not very different from getting the name – you’ll need to use “document.getElementById” to get access to those quantity values.



**What to turn in:**

1. Into the given assignment 3 “turn it in” Word file, paste a screen shot of your products page in the browser after entering input into all 3 text boxes and clicking the “Process Order” button. Paste it into the part 2 screen shot area of the document. Make sure your entire page is showing in the browser. It should look similar to the screen shot shown above except with your own products and name.

2. Into the given assignment 3 “turn it in” Word file, paste a copy of your web page’s code into the part 2 code section of the document. It should contain ALL of your page’s code. Do not paste a screenshot of code, paste the actual text of the code. You can format the font, spacing, or size of the code after pasting into Word to improve its appearance and readability, just be sure not to actually change any of the code itself.

**Part 3: You Receive A Receipt!**  
What to do: Process the order by calculating subtotals, total, and showing detailed receipt.   
How to code it: Receipt is done using “innerHtml”. The event (and function call) is no longer triggered from within the button – it is now triggered by an “addEventListener” in a script at the bottom in the page that activates your “processOrder” function which is still in the “head”:

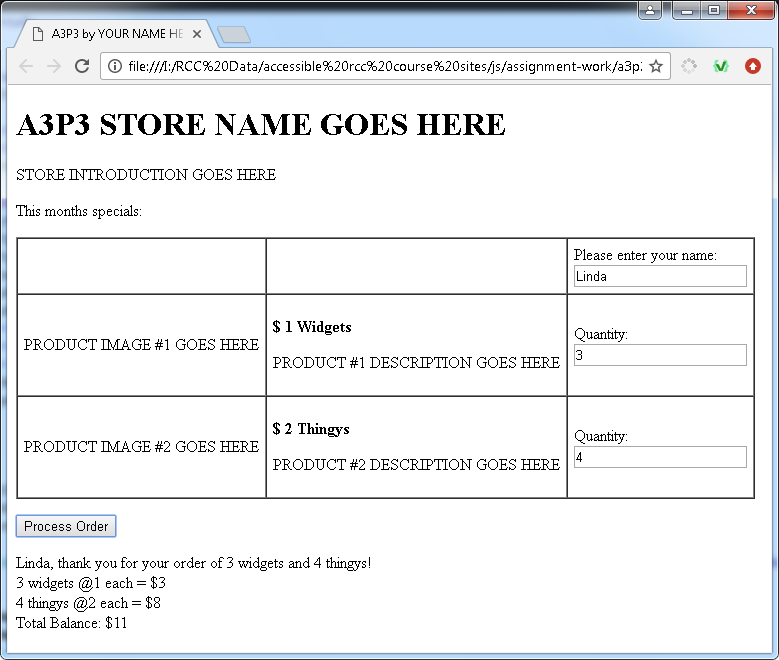
Begin by copying your A3P2 page calling it A3P3. Next, locate your “input” tag for your button and remove the “event” related code from it, leaving the rest of the “input” tag (and surrounding paragraph tags) intact. Remove this code:

**onClick="processOrder();"**

Once that is removed, add the “event listener” code to a script at the bottom of your page (in a new script just above the closing “body” tag). The evet listener code should look just like the code shown below

**document.getElementById("btnProcess").addEventListener("click", processOrder);**

Modify your “process order” function code (in the “head”) using the screen shot below as your guide, with your own products and prices. (Note that your prices should still be controlled by the pricing variables in the “head” from part 1.)

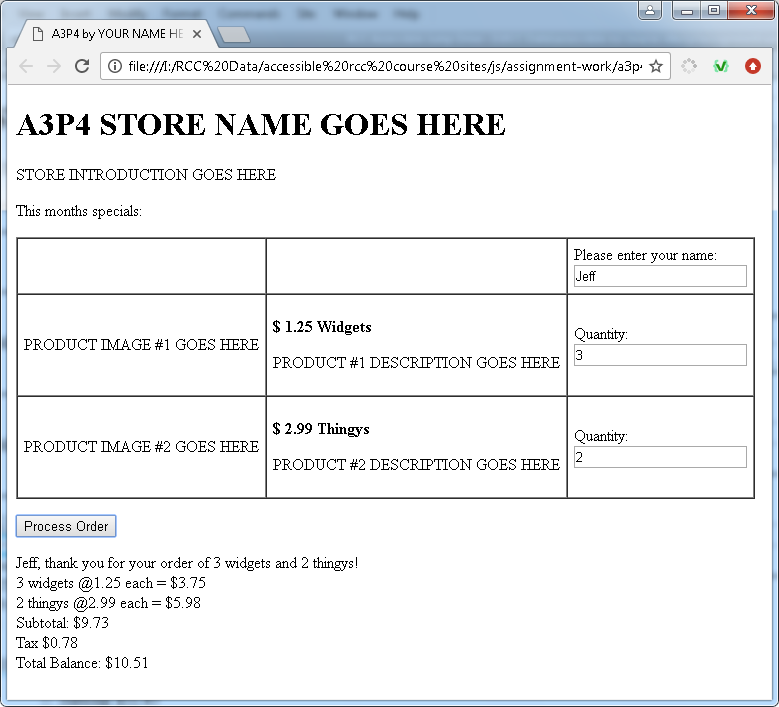


**What to turn in:**

1. Into the given assignment 3 “turn it in” Word file, paste a screen shot of your products page in the browser after entering info into the 3 text boxes and clicking the “Process Order” button. Paste it into the part 3 screen shot area of the document. Make sure all of your page is showing in the browser, especially the bottom part. It should look similar to the screen shot shown above (but should look a lot nicer with your products there instead of the generic products above.)

2. Into the given assignment 3 “turn it in” Word file, paste a copy of your web page’s code into the part 3 code section of the document. It should contain ALL of your page’s code. Do not paste a screenshot of code, paste the actual text of the code. You can format the font, spacing, or size of the code after pasting into Word to improve its appearance and readability, just be sure not to actually change any of the code itself.

**Part 4: Change Your Prices and Pay Your Taxes!**  
What to do: Change your prices to include dollars and cents, add an 8% tax charge to the receipt, show a subtotal, tax, and grand total. Show 2 decimal places on ALL money.   
How to code it: Change prices by changing your 2 global price constants in the “head”. Use “toFixed” to display all money to 2 decimal places everywhere it is displayed on the page. Receipt is still done using “innerHtml”, prices shown within the form still done with document.write (no change from part 2).



**What to turn in:**

1. Into the given assignment 3 “turn it in” Word file, paste a screen shot of your products page in the browser after entering info into the 3 text boxes and clicking the “Process Order” button. Paste it into the part 4 screen shot area of the document. Make sure all of your page is showing in the browser, especially the bottom part. It should look similar to the screen shot shown above (but should look a lot nicer with your products there instead of the generic products above.)

2. Into the given assignment 3 “turn it in” Word file, paste a copy of your web page’s code into the part 4 code section of the document. It should contain ALL of your page’s code. Do not paste a screenshot of code, paste the actual text of the code. You can format the font, spacing, or size of the code after pasting into Word to improve its appearance and readability, just be sure not to actually change any of the code itself.