Week 7 Tasks for Submission

This document contains a number of tasks for you to attempt.
There are three types of task:

- Tasks labelled as "no submission required, not part of portfolio"
  - They are for you to attempt and practice.
  - They provided assistance for you to develop solutions to other tasks that are part of your portfolio
- Tasks labelled as " BASIC Submission Task"
  - These tasks are part of your portfolio geared towards all students
  - The solutions to these tasks must be uploaded for marking by your tutor
- Tasks labelled as "ADVANCED Submission Task"
  - These tasks are part of your portfolio geared mainly towards students aiming for more than a pass grade. Of course students aiming for a pass grade may attempt and submit these tasks.

The solutions to these tasks must be uploaded for marking by your tutor

JavaScript Tasks

Task 1. (no submission required, not part of portfolio)

Create a HTML file named w7t1.html based on template_upd.html

- Ensure the footer section contains your name.
- Inside the header section: add a **h1** element with the value "Week 7"
- Inside the article section: add form tags.
- Within the form tags add a paragraph with a **label** tag containing **Customer Name:**
- Add an input element within the above label tag.
  - The **type** attribute of the element must be **text**
  - The **id** of the element must be **custname**
- Add a button element.
  - The **type** attribute of the element must have the value **button**
  - The **id** attribute of the element must have the value **process.**
  - The **text** on the button must be **Process Name**
- Add the following paragraph tag:
  ** **

Create a JavaScript file named w7t1.js based on template.js. Link this file to w7t1.html.

Within init():

- Use document.getElementByID("…").onclick with the relevant button id to assign call to the function **start()**

Create a function called **start().**
This function must:

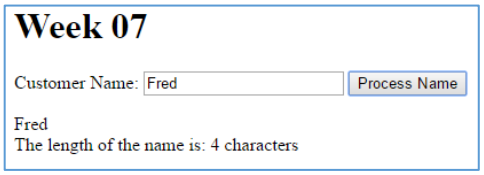
- Create a local variable **vName**

- Get the input from the textbox with id **custname** and assign it to variable vName
- Display the value of the vName in the paragraph with id **output**

Task 2. (no submission required, not part of portfolio)

Same as above, plus

- When you click the button, the HTML id **output** should display:
  - Display the value of the vName
  - Display the length of vName

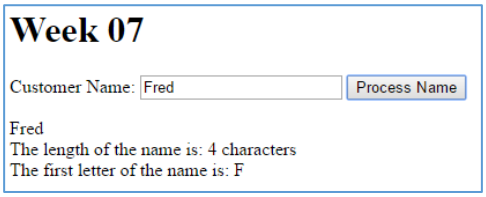


Task 3. (no submission required, not part of portfolio)

Create a function processString() that takes one parameter **str**.
This function extracts the first character from str using charAt(0) and returns the string "The first letter of the name is: X" where X will be replaced by the first character.

Within start():
Same as above, plus

- When you click the button, the HTML id **output** should also display:
  - The string showing the first letter of vName returned by processString()

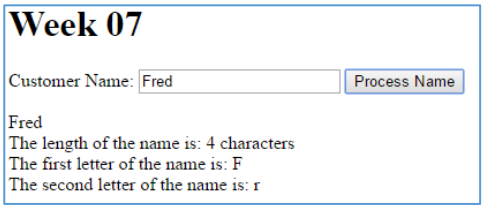


Task 4. (no submission required, not part of portfolio)

Modify processString() so that it returns two first characters of the parameter str as shown on the screenshot.

Within start():
Same as above, plus

- When you click the button, the HTML id **output** should also display:
  - The second letter of vName



Task 5. (no submission required, not part of portfolio)

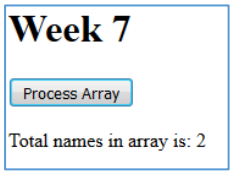
Create a HTML file named w7t6.html based on template_upd.html

- Ensure the footer section contains your name.
- Inside the header section: add a **h1** element with the value "Week 7"
- Add a button element.
  - The **type** attribute of the element must have the value **button**
  - The **id** attribute of the element must have the value **process**
  - The **text** on the button must be **Process Array**
- Add the following paragraph tag:
  ** **

Create a JavaScript file named w7t6.js based on template.js. Link it to the w7t6.html
When a user clicks the button, call the function process().

Within **process()**:

- Create an array named arrNameList with the following names: John, Jenny
- Call the function **processArray()** passing arrNameList as a parameter. This function does not return any values so you do not need assignment statement.

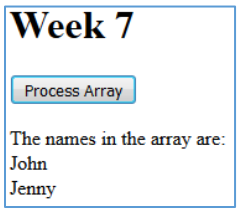


Create a function called **processArray()** which takes one parameter **pArray**.
This function does not return any values. This function must:

- Display the number of elements in pArray in the paragraph with id **output** as on the screenshot (use the relevant property of the array)

Task 6. (no submission required, not part of portfolio)

Same as above, except



- When the user clicks the button, the paragraph with id **output** must display:
  - All the names in the array (see the screenshot)

Task 7. (Pass Submission Task)

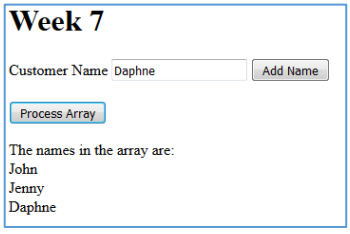
Create a HTML file named **w7P.html** based on template_upd.html

- Make sure that the <footer> section and the meta tag with the name author contain **your student name and ID.**
- Inside the <header> section add a **h1** element with the value "Week 07 Pass Submission"
- Inside the <article> section create a form containing a label showing Customer Name and a textbox with id **custname**.
- Add a button element.
  - The **type** attribute of the element must have the value **button**
  - The **id** attribute of the element must have the value **add**
  - The **text** on the button must be **Add Name**
- Add a button element.
  - The **type** attribute of the element must have the value **button**
  - The **id** attribute of the element must have the value **process**.
  - The **text** on the button must be **Process Name**
- Add the following paragraph tag:
  ** **

Create the JavaScript file **w7P.js** based on **template.js** and link to the html file.

Create a **global** array arrNameList.
Create an addName() function.

- This function adds the name entered into the **custname** input box to the end of arrNameList.

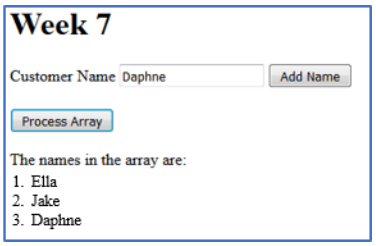


Call this function when the user clicks the button with the id **add**.

Create a function called **displayArray()**.

- This function must list all the names in the array, one per line preceded by position number (see the screenshot).

Call this function when the user clicks the button with the id **process**.



Test your code.

Place the **Screen Captures** of the **web page** in your browser into **W07P.DOCX**
Copy and Paste the **JavaScript code** from your code editor into **W07P.DOCX**
Copy and Paste the **HTML code** from your web page into **W07P.DOCX**

Task 8. (Credit Submission Task)

Create a copy of HTML file **w7P.html** and rename it **w7C.html**.

- Make sure that the <footer> section and the meta tag with the name author contain **your student name and ID.**
- Inside the <header> section change the **h1** element to display "Week 07 Credit Submission 1"

Make a copy of the JavaScript file **w7P.js** and rename it as **w7C.js**

Part 1.
Modify addName() so that before adding a name to the array list it will do the following:

- Call a function named **findName()** to see if the name already exists in the array.
  - Use the **while loop** to check if the name is in the array (note, if the name is found the loop should stop).
  - If the name already exists then return **false**, otherwise return **true**

- If findName() returns **false**, do not add the name to the array, but display "Cannot add xxxx: Name already exists" where xxxx is the custname value

Note, using *break* instruction to exit the loop is a bad programming practice and is not acceptable in this unit.
Test your code with a name duplicate and non-duplicate names.

Part 2.
Modify **w7C.html** as per specs below:
Inside the form tags, after textbox accepting Customer name, add a label showing Year of Birth and a textbox with id **yob**.

Modify **w7C.jc** as per specs below:

- Create another **global** array arrYOB.
- Modify the addName() function that when it adds a name to arrNameList, it also adds the year from the second textbox to arrYOB.
  At this point we recommend that you test that the function works by using Console.log() or alert()

- Modify displayArray() so that it displays year birth next to the corresponding name. The output must formatted as a table with headings Name and Year of Birth. Use <th> tags for headings and <td> tags for cells with data.

Place the **Screen Captures** of the **web page** in your browser into **W07C.DOCX**
Copy and Paste the **JavaScript code** from your code editor into **W07C.DOCX**
Copy and Paste the **HTML code** from your web page into **W07C.DOCX**