

SOEN 287: Web Programming – Winter 2024
Assignment 2

Due Date:	By 11:55pm Mar 29 (Friday), 2024
Evaluation:	7% of final mark
Late Submission:	none accepted
Type:	Individual Assignment
Purpose:	The purpose of this assignment is to have you practice JavaScript and use JavaScript to operate on DOM model
CEAB Attributes:	This assignments is primarily evaluating your use of JavaScript and DOM models hence <i>use of engineering tools</i> as well as <i>design</i> .

Common Requirements

- All the code should pass the HTML validator (<https://validator.w3.org/nu/#file>) without any error or warning messages.
- All the code should have no error or warning message in the browser console after execution.

Exercise 1 : Use JavaScript to Change Content of HTML Tag

Figure 1 is the original HTML page. Figure 2 shows the page after clicking the button. The line above the button is changed. Requirements:

- Define a function (e.g. changeit()) as the event handler in the head of the html
- Register the event handler to the button (e.g. <button ... onclick="changeit()" ...>
- Click the button, the content of a <p> is changed to “Paragraph changed”

Use JavaScript to change HTML content

The original content

[Try it](#)

Figure 1: the original html page

Use JavaScript to change HTML content

Paragraph changed.

[Try it](#)

Figure 2: the changed html page

Exercise 2: Print JavaScript Array

In this exercise, you can put javascript code in <script> inside <body> tag. First, use const to define an array, e.g. const arryStr = const fruits = ["Banana", "Orange", "Apple", "Mango"]; You can mix the other datatype in the array. Then implement the follow functions.

- A. Print all the elements of the array using `toString()` and insert the output string inside a `<p>` tag.
- B. Print all the elements of the array using `join()` and separate the elements using “&”. Insert the output string inside a `<p>` tag.
- C. Print the array in the format of index: value. For example:
0: Banana
1: Orange
2: Apple
3: Mango
You can use `Array.foreach()` or your own implementation

The results are demoed in Figure 3.

JavaScript Array

Print the elements of an array using `toString()`

Banana,Orange,Apple,Mango

Print the elements of an array using `join()`

Banana&Orange&Apple&Mango

Print the elements of an array in the format of index: value

0: Banana
1: Orange
2: Apple
3: Mango

Figure 3: Print the elements in an array of Strings

Exercise 3: JavaScript functions

All your JavaScript functions must be declared in an external .js file which is linked to an HTML file. Each function's name must be as specified below. To demonstrate the functionality of each method, you must make function calls in the document body and insert the results in a html tag. Please use const for array declaration and use let for normal variables. Avoid using global variables.

A. **Function:** addNumbers

Parameter(s): Array of numbers

Each element in the array must be added and the summation (answer) must be returned. You can either implement it using loop or call array.reduce().

B. **Function:** findMaxNumber

Parameter(s): None (Hint: Make use of the arguments array)

From the arguments array, find the number element that is the largest and return it. You can either implement it using loop or call Math.max().

C. **Function:** addOnlyNumbers

Parameter(s): Array of mixed data type

Convert all other datatypes into number using parseFloat(). Please notice parseFloat("3 birds") = 3. So addOnlyNumbers([4, 5, "3.0 birds", true, "birds2"])=12.

D. **Function:** getDigits

Parameter(s): A String

Scan the string and find all the digits (0-9), concatenate them into a string in the order that they are found and return the string of numbers. Use patterns.

E. **Function:** reverseString

Parameter(s): A String

Reverse the entire string (character by character) and return the resulting string. For example, reverseString("See you later") returns "!retal uoy eeS"

F. **Function:** getCurrentDate

Parameter(s): None

Retrieve the current date in the format similar to: *Sunday, Feb 19, 2023* and return it. Use Date.

Exercise 4: Calculate your order

Please reference to the demo implementation below. First, design an online order page, with several items and their prices (you can give any items and prices). Allow the visitor to enter the quantity of each item. After the visitor presses “Place order” button, the total cost is displayed on the same page beneath the form.

All JavaScript code must be external. If any fields are left blank or do not contain an integer (must be an integer), an alert box should display an appropriate error message upon form submission.
Hint: use pattern matching.

Order Books Online

Book	Quantity	Price
Basic Web Programming	<input type="text"/>	\$19.99
Intro to PHP	<input type="text"/>	\$86.00
Advanced JQuery	<input type="text"/>	\$55.00

Figure 4: Before the “Place Order” button is pressed

Order Books Online

Book	Quantity	Price
Basic Web Programming	<input type="text" value="1"/>	\$19.99
Intro to PHP	<input type="text" value="3"/>	\$86.00
Advanced JQuery	<input type="text" value="4"/>	\$55.00

Basic Web Programming (Quantity = 1): \$19.99
Intro to PHP (Quantity = 3): \$258
Advanced JQuery (Quantity = 4): \$220

Final Total: \$497.99

Figure 5: After the “Place Order” button is pressed

Exercise 5: Pop up Windows

Given the following HTML page:

```
<html lang="en">
  <head>
    <title>Exercise 5</title>
    <meta charset="UTF-8">
  </head>
  <body>
    <p><strong>User Information:</strong></p>
    <div id="content"></div>
  </body>
</html>
```

Create an embedded JavaScript function named **getUserInfo** that prompts the user with the following two questions after the HTML page has loaded:

- What is your full name?
- How old are you?

The function should then build a string in the form of:

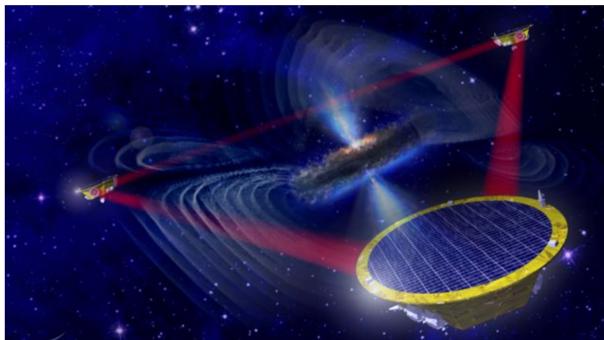
Hi, my name is *NAME* and I'm *XX* years old.

The string must then be inserted into the container div with id: *content*.

Exercise 6: A simple slide show

Slide show is normally implemented by JavaScript. Here is a simple one. This page shows a news about gravitational wave discovery. Under the image, there are two links. When you click the link, the image changes to the next image. Please implement this slide show. The text and the images used for this exercise are from a BBC news report on Feb 11, 2022. The link <http://www.bbc.com/news/science-environment-35523676>. Just for classroom use.

Gravitational waves have been detected for the first time



[Previous](#) [Next](#)

The first direct detection of gravitational waves is without doubt one of the most remarkable breakthroughs of our time. The Advanced LIGO laboratories in the US states of Washington and Louisiana have traced the warping of space from the merger of two black holes about 1.3 billion light-years from Earth.

Figure 6: Slide 1

Gravitational waves have been detected for the first time



[Previous](#) [Next](#)

The first direct detection of gravitational waves is without doubt one of the most remarkable breakthroughs of our time. The Advanced LIGO laboratories in the US states of Washington and Louisiana have traced the warping of space from the merger of two black holes about 1.3 billion light-years from Earth.

Figure 6: Slide 2

Gravitational waves have been detected for the first time



[Previous](#) [Next](#)

The first direct detection of gravitational waves is without doubt one of the most remarkable breakthroughs of our time. The Advanced LIGO laboratories in the US states of Washington and Louisiana have traced the warping of space from the merger of two black holes about 1.3 billion light-years from Earth.

Figure 6: Slide 3

Gravitational waves have been detected for the first time



[Previous](#) [Next](#)

The first direct detection of gravitational waves is without doubt one of the most remarkable breakthroughs of our time. The Advanced LIGO laboratories in the US states of Washington and Louisiana have traced the warping of space from the merger of two black holes about 1.3 billion light-years from Earth.

Figure 6: Slide 4

Exercise 7: Step 2 of the Running Project - Adopt a Dog/Cat

Following are the additions for the Pet Adoption Website started in assignment 1

1) *Date and Time:*

In the header area of your website, use JavaScript to display the current date and time with second precision. The Time should automatically refresh and show the updated time every second.

2) *Browse Available Pets* page:

Create a **Browse Available Pets** page [pets.html](#) linked from the side menu. Inside the content area of this page, create a list of a couple of fictitious pets available for adoption.: [pets.html](#). Include an image, along with the information found in the **Have a pet to give away** page. Make sure the format of the page is easy to read and that the look is consistent with the rest of your site.

Next to each pet, add a button labeled "*Interested*". In this iteration of the project, clicking on the button need not do anything.

3) *Find a dog/cat* page:

The form was created in the previous assignment. In this assignment, you are required to implement client-side validation using JavaScript. Upon pressing the form's submit button, ensure that no fields are left blank. If any fields are blank upon submitting the form, display an error message, but retain the rest of the form as the user had input before, without resetting the filled values.

4) *Have a pet to give away* page:

The form was created in the previous assignment. In this assignment, you are required to implement client-side validation using JavaScript. Upon pressing the form's submit button, ensure that no fields are left blank and that the provided email follows a valid format. If any fields are blank or invalid upon submitting the form, display an error message, but retain the rest of the form as the user had input before, without resetting the filled values.

Hint: Regular Expressions are your friend, but you should look up what a valid format for an email is.

Note on URL Paths

Ensure that you consistently use "**relative file paths**" in your project when linking or loading JS, CSS, and image files in your HTML code. Avoid using "**absolute file paths**," such as

"*C:\Users\Bob\Projects\SOEN287\A2\Q2.CSS*" or "*http://localhost:1313/SOEN287/A2/Q2.CSS*," as they cause your project to malfunction on a different machine during marking.

For example, the URL of the menu button linking to the "Find a Dog/Cat" page should be simply written as `href="find.html"`, instead of using the absolute file path like
`href="C:\Users\Bob\Projects\SOEN287\A2\Q7\find.html".`

For more information, please refer to the following link:

https://www.w3schools.com/html/html_filepaths.asp

Submitting Assignment 2

- Place all HTML files for Q1 to Q6 in the root directory of your project, while their corresponding CSS and JS files can be organized in subdirectories as per your preference. As for Q7, place all files in a subdirectory named "Q7". When submitting your assignment, create a single zip file that includes all folders and files required for the project.
- Please give meaningful names to files to make the evaluation process easier.
- The zip file should be called *a#_studentID*, where # is the number of the assignment and *studentID* is your student ID number.
For example, for this second assignment, student 123456 would submit a zip file named *a2_123456.zip*
- Please make sure to review your final version right before submission and avoid making unreviewed last-minute changes. The submitted file will be graded, and it is your responsibility to ensure it is the correct version.

Evaluation Criteria for Assignment 2 (26 points)

Question 1 - 3 pts.	
<ul style="list-style-type: none">• "changeit" Function• Event Handler• Correct Functionality	0.5 pts 0.5 pts 2 pts
Question 2 – 3 pts.	
<ul style="list-style-type: none">a) <code>toString</code>b) <code>join</code>c) <code>index: value</code>	1 pt 1 pt 1 pt
Question 3 – 6 pts.	
<ul style="list-style-type: none">a) <code>addNumbers()</code>b) <code>findMaxNumber()</code>c) <code>addOnlyNumbers()</code>d) <code>getDigits()</code>e) <code>reverseString()</code>f) <code>getCurrentDate ()</code>	1 pt 1 pt 1 pt 1 pt 1 pt 1 pt
Question 4 - 4 pts.	
<ul style="list-style-type: none">• Order Form Page	1 pt

• Make sure all fields have an integer	1 pt
• Calculate & display total as well as summary of order	1 pt
Question 5 - 2 pts	
• Prompt User for Inputs	1 pt
• Display the Output String	1 pt
Question 6 - 2 pts	
• Implement simple slide show	2 pts
Question 7 - 6 pts.	
• Date and time in header	1 pt
• Browse available pets	1 pt
• Validate input for Find a dog/cat page	1.5 pts
• Validate input for Have a pet to give away page	2.5 pts
TOTAL	26 pts