Assignment 3

-Developing a Web Page 3 (150 points)-

Important: Students are supposed to do all the exercises in the Module 1, 2, and 3, prior to try this Assignment.

Assignment Submission Rules:

1) **Detection of plagiarism will result in receiving the failing grade.**
2) After completion, students must submit the following three types of deliverables through D2L assignment box (previously Dropbox) by the deadline:
   a. **Your CODE (".html", ".css", and ".js")** must be submitted in files separate from the report, so that I can compile and run it in my environment. It is better submitting a compressed project folder. Each function, method, class, and variable should be identified and their functions explained in comments.
   b. **Your CODE in document type:** All of your implementation must be converted into any of three format: ".txt", ".doc or .docx", or ".pdf". You can simply copy and paste your CODE on "notepad" or "MS word". The converted files must not be compressed and must be submitted separately.
   c. **A REPORT FILE (MS word or pdf):** This text file should include i) A description of your solution; and ii) the output displayed when running your code. Your solutions description should include a synopsis of how your code is intended to work and the tests that you created to prove that it works as intended. The output can be a screen-shot from your computer.
   d. **Please submit the bullet "a" to the submission folder named "Homework 3-Codes (zipped file)", and the bullets "b" and "c" to the submission folder named "Homework 3- Codes in Document and Report".**
   e. **Failure to follow the rules will result in deducting points.**
   f. **Submission Example:**
      i. Assignment folder: **Homework 3-Codes (zipped file)**
         1. Codes.zip
      ii. Assignment folder: **Homework 3-Codes in Document and Report**
         1. Html.txt
         2. Css.txt
         3. Js.txt
         4. Reports.docx
3) It is much better to submit a partial/failed-attempt solution than none. Include the circumstances of the incompletion in your report.

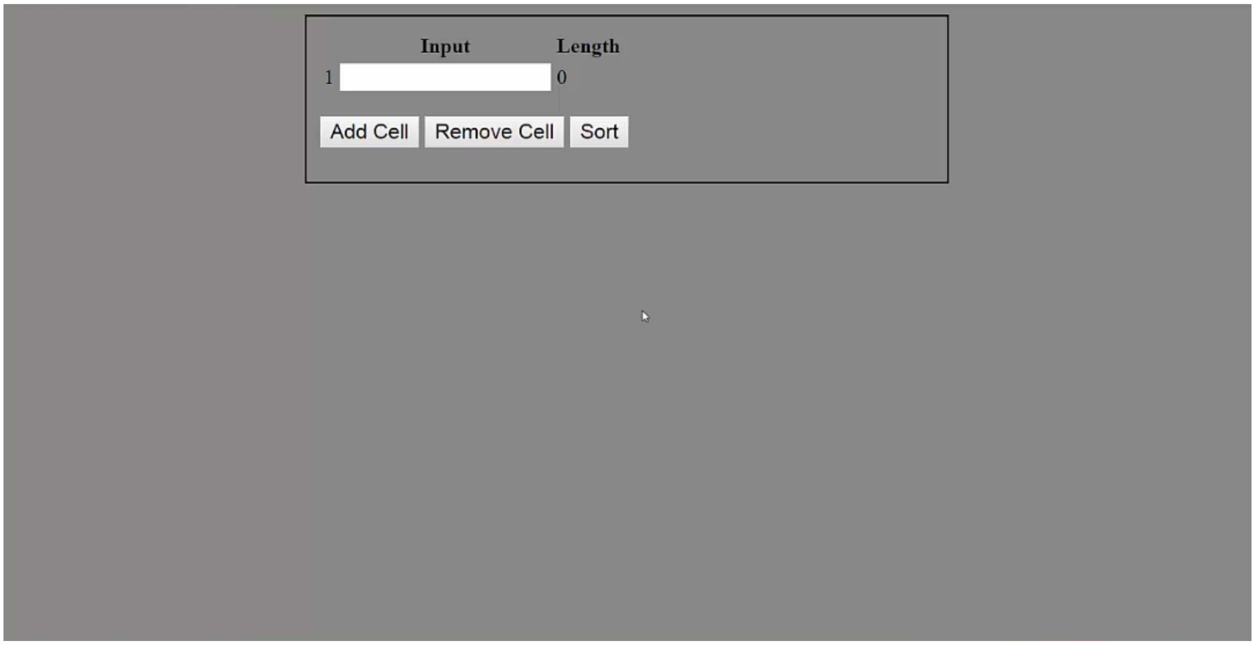
Problems:

Develop a Javascript application that allows the user to enter strings into a variable number of input widgets. As a string is entered by the user its length is automatically displayed next to the input widget. Buttons are provided to control the interface and to provide some functionality.

- Clicking “Add Cell” button will add another input widget.
- Clicking “Remove Cell” button will remove an input widget, always leaving at least one input widget.
- Clicking “Sort” button will cause the visible strings to be sorted.
- Make a button named ‘Assignment 3’ on your web page (Assignment 1), and let clicking the button display the result of this assignment.

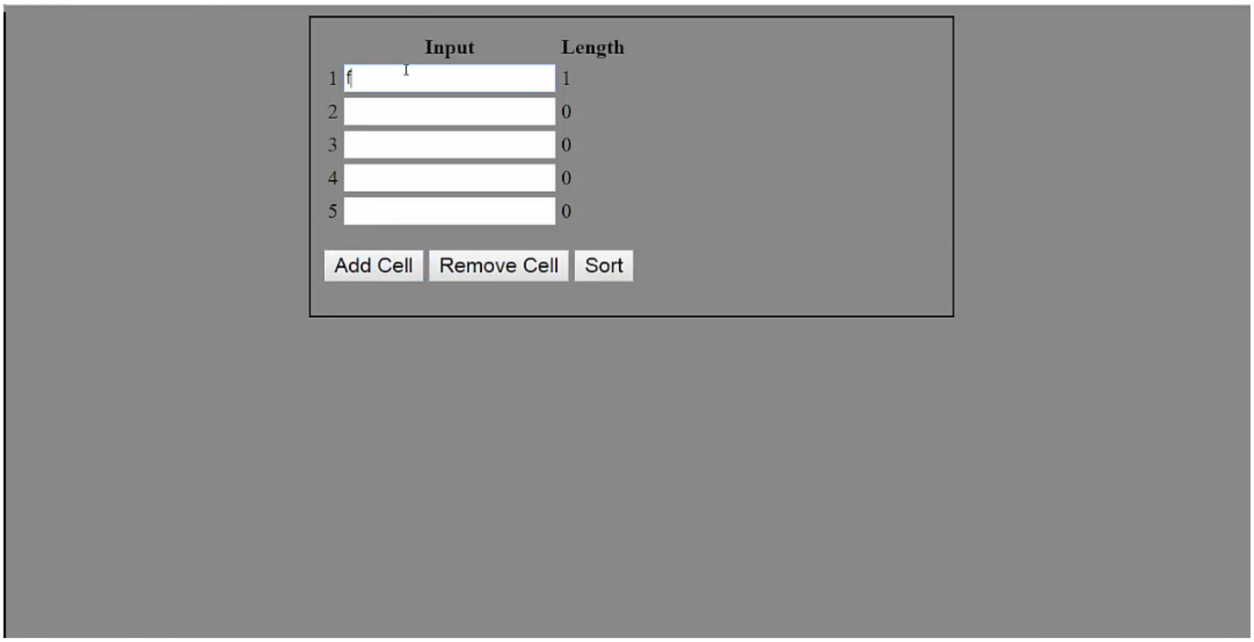
Organize the application **in three files**: the HTML file that is the user interface to the application; a file with most of the Javascript code; a file with the styling rules.

1. Basic screen with simple styling (30 points)
   A. Provide styling that will approximately center the visible components. Also, change the default color for the background.



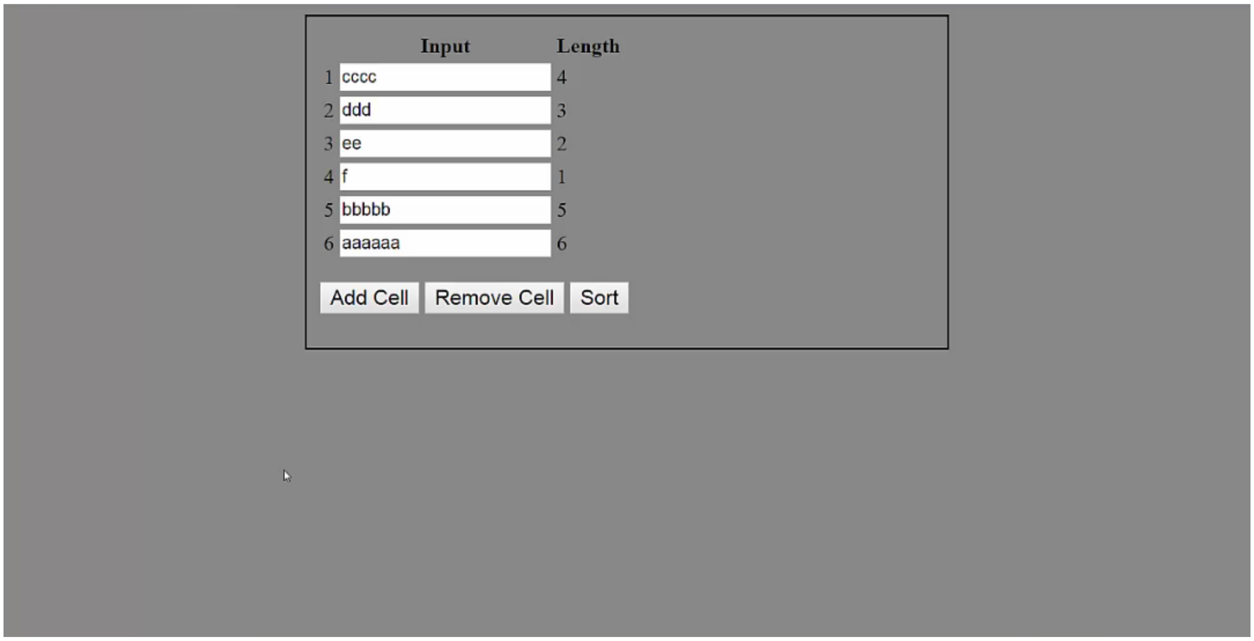
	Input	Length
1		0

2. Add cell/Remove cell (40 points)
   A. As a convenience to the user, the application will remember the string entered into a widget even after the widget is removed.
   B. If the widget is added back again, then the saved value will be displayed in the widget.



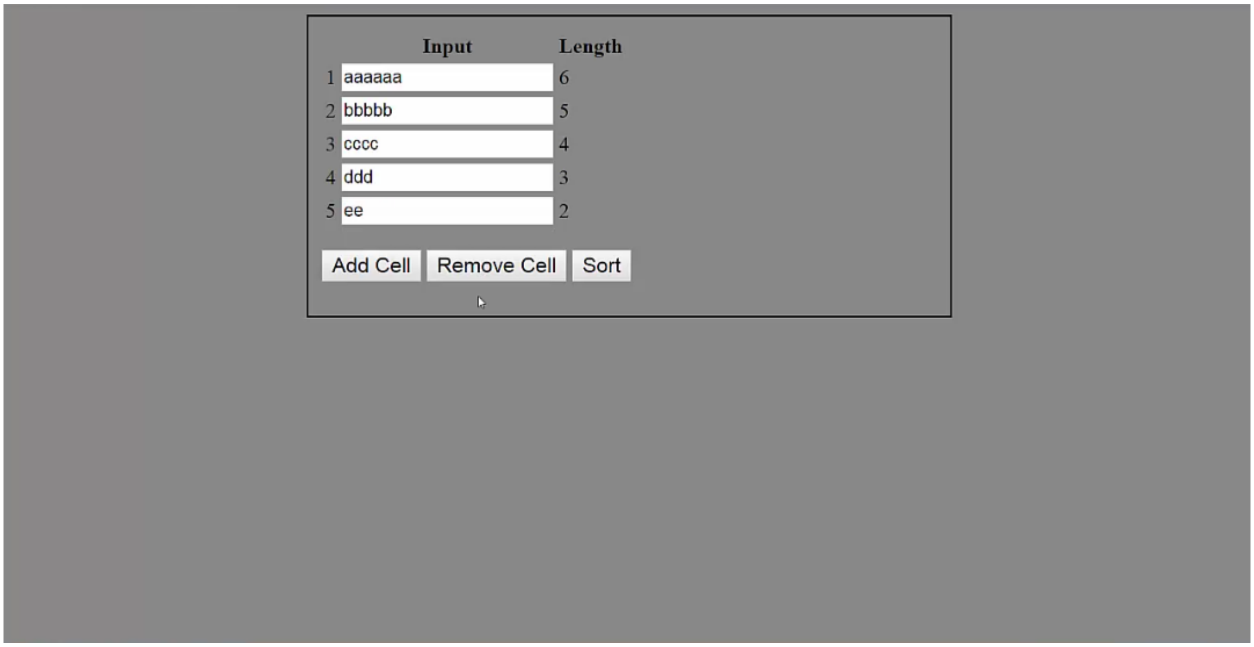
	Input	Length
1	f	1
2		0
3		0
4		0
5		0

3. Typing and Length (30 points)
   A. Every time a user types a key, it checks and shows the length of a string in each widget on the next (in a real-time manner).



	Input	Length
1	cccc	4
2	ddd	3
3	ee	2
4	f	1
5	bbbbb	5
6	aaaaaa	6

4. Sorting (40 points)
   A. Be careful, the saved strings do not participate in a sort operation if not visible.
   B. You can use array sort() method.



	Input	Length
1	aaaaaa	6
2	bbbbb	5
3	cccc	4
4	ddd	3
5	ee	2

5. Make a button named ‘Assignment 3’ on your web page (Assignment 1), and let clicking the button display the result of this assignment (10 point).

Testing

Your application will be either tested from your project directory or the files will be copied to another directory. Use **relative paths** to link parts of the application together.