# Assignment 1: Write your first script (25 points)

**Due:** see course outline for a specific date

In this assignment, you will practice how to manipulate content and visibility filters. You only need materials from lesson 1 and 2 to complete this assignment. Please check the instruction and requirements to complete this assignment.

## ■ Instructions

You have a page with its stylesheet (Download the assignment_1_source_files under Assignment 1 module under CONTENT section). There are five buttons under the header element. You only need to create jQuery functions and attach them to these buttons as follows:

1. Select buttons by their ID names. For example, the first function should select a button by its ID called '#add_new_box' (see the index.html, under the script tag or block).
2. After selecting a button, we need to fire an event when a button clicked. To do that use a function called '.click' (see the function under '// 1. Add new box' comment).
3. Now, we need to pass a function to the '.click()'. If you follow the three steps, you code should look like this:

```
$('selector').click(function(){
  //Add your jQuery code here
  });
```

### Here are the button function details:

### 1. Add a box with a default style function:

This button should add a new box to the page layout. The new box will have the default style as showing in the CSS style sheet. The box content should not be the same as the last box created, for example, if the last box content is '5' the new box content should be '6' and so on. So as users keep adding boxes, the content should be updated (see Figure 1 and 2).

***JavaScript tips:***

You may need to use a mix of JavaScript and jQuery codes to write this function. You may need to use these JavaScript code statements: variables, addition operations, comparison operations, and conditional statements.

### 2. Remove a box function:

Similar to the add button function. Create a function and attach it to the button called 'Remove a box'. This button should remove the last box in the layout and update the content for the page (see Figure 2 and 1).

***JavaScript tips:***

You may only need to use JavaScript variables for this function.

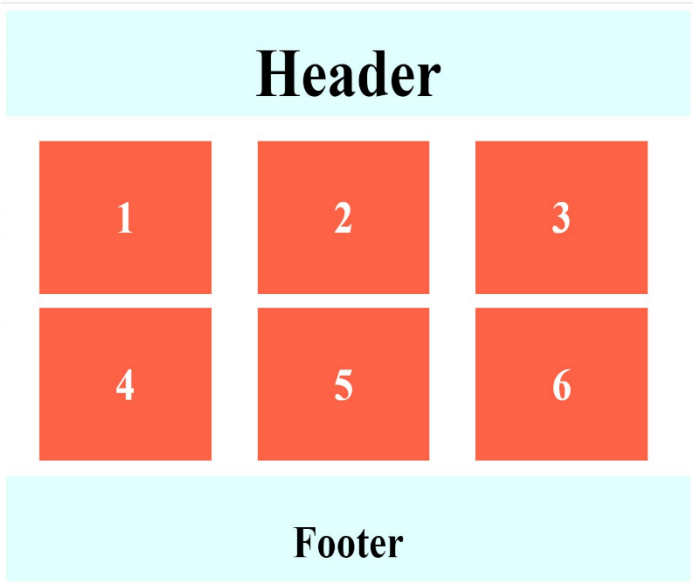


*Figure 1: An example of add a new box with content of '6'*

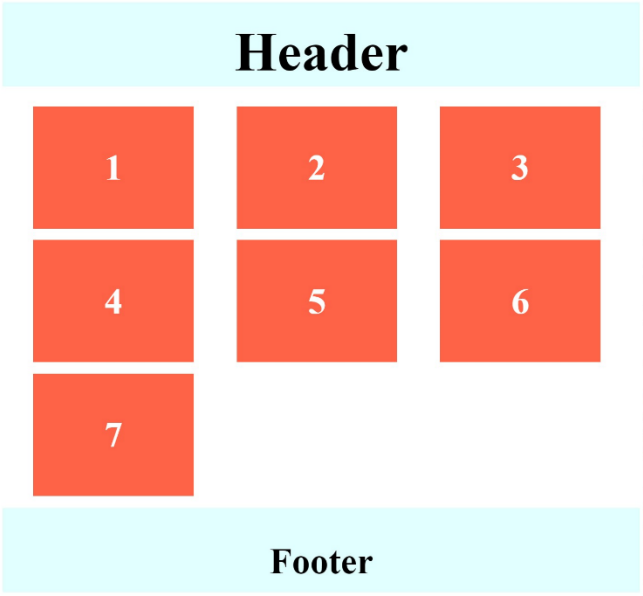


*Figure 2: An example of add a new box with content of '7'*

### 3. Add a box with a yellow background function:

This function will add a box to layout but with a different style from the default box's style. The new box should have a yellow background. Note that the content should be updated the same as the 'add a new box' function (see Figure 3).

***JavaScript tips:***

You may need to use a mix of JavaScript and jQuery codes to write this function. You may need to use these JavaScript code statements: variables, addition operations, comparison operations, and conditional statements.

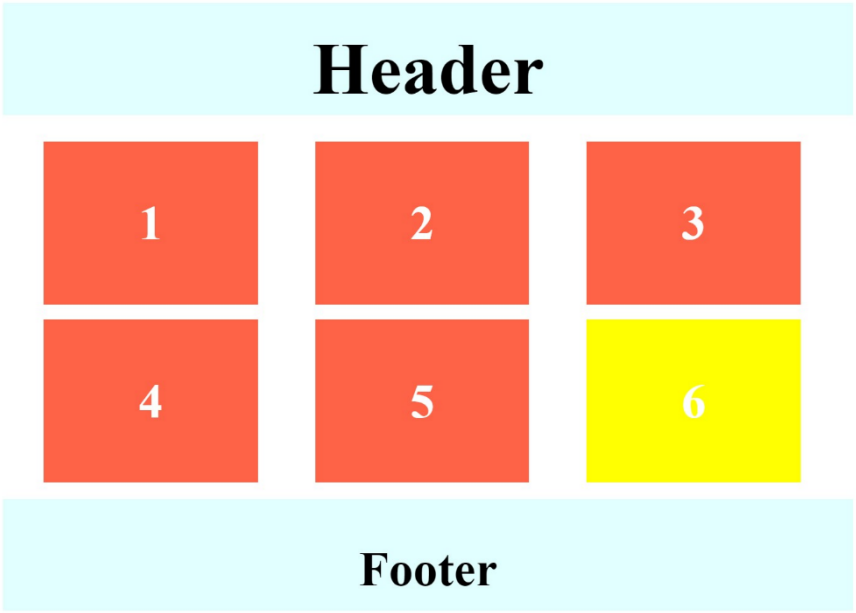


*Figure 3: An example of add a new box with content of '7'*

## 4. Order boxes into rows function:

Create a function and attached it to ‘Change page a layout’. This function should sort boxes into rows and change its width to the full width of the page layout (see Figure 4).

***JavaScript tips:***

You may need to use a mix of JavaScript and jQuery codes to write this function. You may need to use these JavaScript code statements: variables, addition operations, comparison operations, and conditional statements.

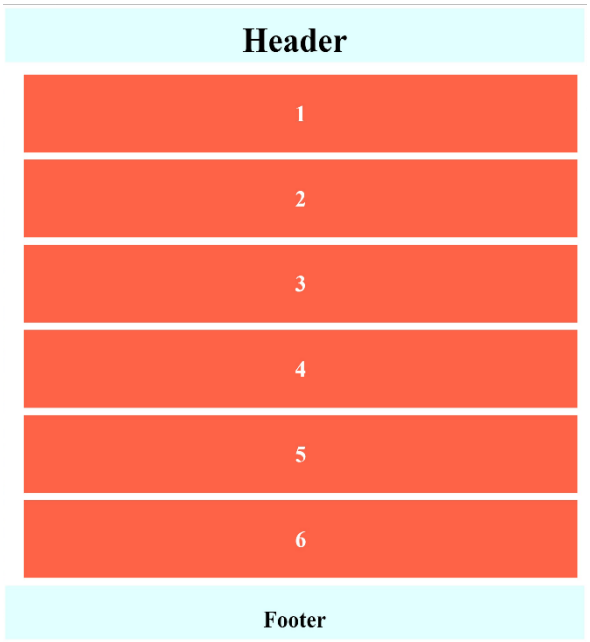


*Figure 4: An example of add a new box with*

## 5. Print a DOM tree function:

Create a function and attach it to the ‘Print DOM tree’ button. This function should only print the DOM tree for boxes area. Each element must be print in single lines. You can format the output as a tree by using dashes. Single ‘-‘mean top element or the root of the tree, then inside element ‘branches’ should start with two dashes ‘- -‘ (optional) (see Figure 5).

***Javascript tips:***

You may need to use these JavaScript code statements: variables and addition operations.

```
- <ul class="grid-container">
-- <li class="grid-item">1</li>
-- <li class="grid-item">2</li>
-- <li class="grid-item">3</li>
-- <li class="grid-item">4</li>
-- <li class="grid-item">5</li>
-- <li class="grid-item">6</li>
```

*Figure 5: this is the DOM tree of figure 4.*

## Requirements:

1. Your code should not have too many errors, and it should be organized.
2. The layout should not change when boxes added or removed from the page.
3. The removing button should disable (You can hide it) when the page layout has 1 box.
4. Each button should function correctly and complete its task.
5. Page elements should not overflow, so make sure always to check out the layout.
6. Limite adding boxes to a max of 10 boxes. The adding button should be disabled (You can hide it) when the page layout has 10 boxes.
7. Comments should be organized and have meaning.
8. The variables should have meaning names that related to their content.

**Please refer to the rubric at the end of this document for evaluation details.**

## Submit

1. Name your folder assignment_1_project.
2. Right-click on the folder and select 'Send to -> Compressed (zipped) folder'.
3. Upload the zipped folder to the Assignment 1 Dropbox

## How to zip your project folder:

Here are some links that may help with compressing your main/root folder and only send that zip folder

- How to zip a file in Windows 10: https://www.laptopmag.com/articles/how-to-zip-files-windows-10
  - Free file compressor application for Windows: http://www.7-zip.org/download.html
- How to zip a file in Mac: https://www.lifewire.com/how-to-zip-and-unzip-files-and-folders-on-a-mac-2260188
  - Free file compressor application for Mac: https://theunarchiver.com/
- How to save a pdf in MS Word document: https://www.bettercloud.com/monitor/the-academy/save-word-doc-pdf/

## Evaluation

This assessment is graded out of 25 points and will be evaluated using the following rubric.

Learners may receive partial scores or zero for unacceptable work.

| | Excellent: 5 point | Good: 3 point | Fair: 1 point | Poor: 0 point | Score |
|---|---|---|---|---|---|
| **Add a box with a default style and Remove a box** | Boxes added or removed with default style and its content updated correctly. The page layout is organized. | Boxes added or removed with default style and its content updated correctly. However, the page layout is not organized. | Boxes added or removed with the default style, but its content not updated correctly. However, the page layout is organized. | Boxes added or removed but not styled and its content updated incorrectly or correctly. The page layout is not organized and element overflow. | **/5** |
| **Add a box with a yellow background** | Boxes added with required style and its content updated correctly. The page layout is organized. | Boxes added with required style and its content updated correctly. However, the page layout is not organized. | Boxes added with required style, but its content not updated correctly. However, the page layout is organized. | Boxes added but not with required styled and its content updated incorrectly. The page layout is not organized and element overflow. | **/5** |
| **Order boxes into rows** | Boxes are organized, and their width fit into the page. Boxes order as required. The layout is organized and there no overflow issues. | Boxes are organized, and their width fit into the page — boxes order as required. However, the layout is not organized, and there are a few overflow issues. | Boxes are organized, and their width doesn't fit into the page — boxes order as required. However, the layout is not organized, and there are some overflow issues. | Boxes are not organized, and their width doesn't fit into the page. Boxes order as required. However, the layout is not organized, and there are a lot of overflow issues. | **/5** |
| **Print a DOM tree** | A tree only includes the required elements. The output formatted as requested, and the output is readable. | A tree only includes the required elements. The output is not formatted as requested, but the output is readable. | A tree includes other elements then the required elements. The output is not formatted as requested, but the output is readable. | A tree includes other elements then the required elements. The output is not formatted as requested, and the output is not readable. | **/5** |
| **Coding part** | There are no errors in another coding on the site as found by me or an online validator. | There are 1-3 coding errors on the site as found by me or an online validator. | There are 4-5 coding errors on the site as found by me or an online validator. | There are more than 6 coding errors on the site as found by me or an online validator. | **/5** |
| | | | | **TOTAL** | **/25** |