Download ***addEventListener.zip*** from Moodle

Inside ***script.js*** add the **window.onload** event, this will fire when the window is **fully loaded** - the anonymous function assigned will then run

```
1   window.onload = function () {
2
3   } //END window
```

(***Fully loaded*** *means all html, images etc…will be completely loaded into the browser and the DOM also created, ready to go, therefore any JavaScript will be able to target any element in page without error*)

Global variable named **counter**……Adding an ***event listener*** to the buttons which calls a **function** when clicked

```
1   window.onload = function () {
2       //Global counter variable
3       let counter = 0;
4       //Add eventlisteners to the buttons
5       document.getElementById("btn").addEventListener("click", storeClickCount);
6       document.getElementById("reset").addEventListener("click", reset);
7
8   } //END window
```

Create the **storeClickCount** function

NOTE:

1. The **+=** assignment operator stores and **UPDATES** the **counter** variable each time the button is clicked.

2. **Number()** method will parse input value to an integer

3. Using back ticks starting line 13 – you can break/return to the next line without using many quotation marks - the code will not break

counter += userInput *is shorthand for saying* **counter** = **counter** + userInput

```
8     //Function captures the input value & parse to a number
9     function storeClickCount() {
10        let userInput = Number(document.getElementById('userinput').value);
11        counter += userInput;
12        let dspOutput = document.getElementById("output");
13        dspOutput.innerHTML = `You have chosen to increment the counter by
14        <span class='countby'>${userInput}</span>`;
15
16        dspUserCount = document.querySelector(".countdisplay");
17        dspUserCount.innerHTML = counter;
18    } //END storeClickCount
```

Add a ***reset function*** that sets everything back to default settings

```
function reset() {
    document.getElementById('userinput').value = "";
    counter = 0;
    userInput = 0;
    let countOutput = document.getElementById("output");
    countOutput.innerHTML = `output here ${userInput}`;
    let countUpdate = document.querySelector(".countdisplay");
    countUpdate.innerHTML = counter;
}
```

Add validation inside the storeClickCount function

Below the *userInput* variable insert an **IF Statement** that only allows a Number between **1 & 10**

```
let userInput = Number(document.getElement
//validation allows number between 1 & 10
if (userInput < 1 || userInput > 10) {
    alert("Number not in range");
    return false;
}
```

NOTE:
The ***return*** keyword **stops** the execution of a function, in this case **storeClickCount** and returns a value
(The value here being ***false)***

TASKS:

1. Add a new **<p>** tag somewhere inside ***index.html*** to be a placeholder for an error message

2. Send an error message to the **<p>** tag instead of using an alert box