# 1) Goals and Overview

1) To create a simple calculator using HTML, CSS, JavaScript/jQuery.
2) There is **no** backend involved in this application.
3) **<u>You can only access links provided in the resources section of this document, unless you are stuck with something, please inform your instructor.</u>**
4) I have attached a screen recording of how your application should work.

# 2) HTML

1) A basic HTML file is provided to you.

# 3) Add CSS to make your calculator "pretty"

1) **All elements** will have the font 'Consolas', monospace;
   Resources : R1 & R2
2) **All elements** will have the color #1f262a
3) **body** will have the background color #3fb399
4) **div with class container** will have width set to 480px, background color white, and margin 120px auto;
   Resources : R3
5) table will have width 100% and a solid 1px wide border of color #333333
   Resources : R4
6) td elements will be 25% wide
7) All **button** elements will be 100% wide, height 60px, font size will be 24px, background color will be white, and **no** border
   Resources : R2 and R4
8) An element has ID calculatorInput. That element will have height 120px, font size is 40px, vertical-align bottom, text-align right, padding-right 16px, background color #ececec
   Resources : R1

# 4) JavaScript and jQuery:

1) Include the jQuery library inside head tag

```
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
```

2) Create and include a calculate.js javascript file. This should come after the jQuery include, for the jquery library to be available inside calculate.js
3) Use $.dcoument ready and add event listeners for all buttons inside the callback.
   Resources : R5, R10 and R11
   Example event listener for a Hello All

```
$( document ).ready(function() {
 $(".xyz").on( "click", function() {
      console.log( $( this ).text() );
 });
});
```

4) You should have 3 click event handlers:
5) On click of class “inputElements”, append that element’s text to element with ID calculatorInput
   Resources: R9. Also, refer to the above code to get clicked element’s text
6) On click of class “clear”, clear the calculatorInput content
   Resouces : R8
7) On click of class “evaluate”, i.e. the “=” button, you have to evaluate the calculatorInput’s HTML content. Display the evaluated result on calculatorInput.

   You can use JavaScript’s built-in method “eval”
   Example: eval(“2 * 6 / 5”) will return 2.4
   Resources: R6, R7 and R11

**5)** Resources :

R1 : https://www.w3schools.com/cssref/css_selectors.asp

R2 : https://www.w3schools.com/css/css_font.asp

R3 : https://www.w3schools.com/css/css_margin.asp

R4 : https://www.w3schools.com/css/css_border.asp

R5 : http://learn.jquery.com/using-jquery-core/document-ready/

R6 : https://www.w3schools.com/jsref/jsref_eval.asp

R7 : https://www.w3schools.com/jquery/jquery_dom_get.asp

R8 : https://www.w3schools.com/jquery/html_empty.asp

R9 : https://www.w3schools.com/jquery/html_append.asp

R10 : http://api.jquery.com/on/

R11 : https://www.w3schools.com/jquery/html_text.asp

R12 : https://www.w3schools.com/js/js_errors.asp

**6)** EXTRA CREDIT: Put the eval() method inside a try block. If there is an exception in eval, display an error message on calculatorInput saying, calculator input invalid
Resources: R12