

# CPSC 2030 Assignment 3: Basic jQuery

Total Marks: 13

## Setup:

- Download the starter files from D2L.
- HTML code, and some CSS code, has been provided. You will write your styles in the file called **mystyles.css**. **You may edit the HTML files only where specifically indicated. Do no edit the files style.css and sticky.css!**
- **There are some screenshots and demo videos in the assignment folder that will show you what the site should look like when you're done. Study these carefully!**

## Hand in:

When you have completed the assignment, please create a **zip** archive (not some other kind of archive) from your **a3** folder and hand it in to the **Lab 3** folder on D2L. **Please do not hand in the screenshots and demo videos since it will make it difficult for the marker to download.**

## Task 1: Mobile Layout

- The mobile layout will be a simple, one-column layout; you shouldn't need to use flexbox at all for this. Look at the screenshot and apply the necessary styles to match it as closely as possible. Use the mobile-first approach. Also, note that some styles have been provided already in the file **style.css**, so you don't have to do everything.
  - o Fonts and Colors:
    - The h1 font is **Bungee**.
    - The h2 font is **Slabo 27px**.
    - Other text font is **Mukta Malar**.
    - The colors used are **tomato** and **lemonchiffon**

## Task 2: Desktop Layout

- Below your mobile styles, use a **min-width media query** to specify a desktop layout for **800px** and wider. You will use Flexbox to create the entire layout for the page. **You should not use the float property anywhere in your CSS!**
- The content of the page should have a maximum width of **1000px**.
  - o The header should have a **height** of **90px**.
  - o The header and image should take the entire width of the window or viewport, but the h1 text, nav menu and search button should all be contained within a maximum width of 1000px. Feel free to use **margin:auto** to center everything.

## Task 3: jQuery Sticky Header

- The strategy for implementing this functionality is to use jQuery to add a class called **sticky** to the header element when the user has scrolled down below the image. This class will set the position property of the header to **fixed** and will change a few other properties of the style. This class should then be removed, using jQuery, when the user scrolls back up to the area of the page where the image is visible. Note that I have already written the CSS code for the sticky class and placed it in **sticky.css**.
- Get the code linking to the jQuery library from the following source and paste it into the head of your page: <https://developers.google.com/speed/libraries/>
- There is a script tag in the head section of your document. This is where your jQuery code will go.
  - Use the **ready()** method to make sure your document is ready to go before the other functions are run. Your jQuery code will go inside the function of this method. See the following resource for an explanation of this method: [https://www.w3schools.com/jquery/event\\_ready.asp](https://www.w3schools.com/jquery/event_ready.asp)
- Use the **on()** method in jQuery to add a scroll event listener to the **window**.  
[https://www.w3schools.com/jquery/jquery\\_events.asp](https://www.w3schools.com/jquery/jquery_events.asp)
- When the user scrolls on the page, check the distance that the user has scrolled so far by using the **\$(window).scrollTop()** function. This will return the distance from the top of the screen that the user has scrolled. Use **if** and **else** statements to either add or remove the **sticky** class depending on whether **\$(window).scrollTop()** is above or below a certain distance. You can use the methods **addClass()** and **removeClass()** to achieve this.  
More info:
  - [https://www.w3schools.com/jquery/css\\_scrolltop.asp](https://www.w3schools.com/jquery/css_scrolltop.asp)
  - [https://www.w3schools.com/js/js\\_if\\_else.asp](https://www.w3schools.com/js/js_if_else.asp)
  - [https://www.w3schools.com/jquery/jquery\\_css\\_classes.asp](https://www.w3schools.com/jquery/jquery_css_classes.asp)
- Notice that when you scroll up and down in the area where the header becomes sticky, there is some flickering and jumping on the page. What is causing this? Look at the code I've provided in the file **sticky.css**, and see if you can find a way to use the **drop** class, along with jQuery, to fix this problem.
- Use the **transition** property to animate the size of the **h1** text and the height of the **header** when transitioning from the sticky to fixed header.
  - **Optional Challenge:** investigate the **position:sticky** property. How could this make the following tasks easier? Why might we choose to forego the CSS solution and use the jQuery solution instead?

## Task 4: jQuery Search Form

- I've written the CSS and HTML code for the search form, but there are a few things you'll have to do:
  - Add the magnifying glass icon and x icon, using Icomoon, to the buttons that display the search form, submit the search, and close the search form. **Feel free to link the necessary stylesheets in the HTML.**
  - In the HTML, you can see that I've added some text to the buttons to describe what they do. This text is necessary to describe the function of the buttons for users who

are accessing the page using screen readers. However, for users who are navigating visually, we would like to hide the text and use only the Icomoon icons to communicate the function of the buttons. To hide this text visually, you can use the visually-hidden method described here: <https://a11yproject.com/posts/how-to-hide-content/>. **You may edit the HTML to implement this method.** Please note that using **display:none** or **visibility:hidden** will make this text invisible to screen readers, thus defeating the purpose of having it in the document.

- Configure the search window to display when the search icon in the menu is clicked (The magnifying glass icon).
  - To do this, use **on()** to detect when the user clicks on the button, and use the **fadeToggle()** or **fadeIn()** method to cause the form to appear .
  - Using the same technique, configure the search screen to close when the user clicks on the **x** in the top right corner.
  - Make sure the search functionality is fully keyboard accessible and can be navigated using the **tab** key (use Chrome to test this):
    - When the user clicks the magnifying glass button to expands the search screen, use the **focus()** method to make sure focus is placed on the search bar.

## Check your Work:

- Look at the screenshots and demo videos in the assignment folder. Is there anything you've missed?
- **[2 marks]** Styling: fonts, colors, background colors, margins, paddings, etc.
- **[2 marks]** Flexbox layout
- **[4 marks]** Sticky header: header becomes sticky at correct position and flicker/jumping has been removed. Animation has been applied.
- **[5 marks]** Search form: icons have been added, text has been hidden correctly, buttons all work properly (with fade in/out animation) and focus is applied properly when expanding the form.

**Total: 13 marks**