

## Javascript Programming

### Canvas #16 – Placing Text

Objective: Demonstrate that you understand how to place text on the canvas. You will also keep track of mouse clicks and display the number of times the mouse has clicked in the center of the canvas.

1. In your *Javascript* folder, create a folder titled *first\_last\_canvas\_16*. Open your *canvas\_template.html* file and save it in your *first\_last\_canvas\_16* folder as *canvas\_16.html*. Edit the *canvas\_16.html* file
  - a) Change the title to *First Last - Canvas 16*
  - b) Change the width and height of the canvas so that the canvas is a **square**. The side should be between 400 and 600. Set this up in *init()*.
  - c) You will need 3 colors and 2 silly words. You may choose the silly words and the colors. The words should be short. One color will be used as the background for the canvas, the other two will be used to fill and outline the words. Four of the same word will be filled with the first color and outlined with the second, and four of the other word will be filled with the second color and outlined with the first. Create constants for the 3 colors.
  - d) Choose a font and text size. You can play around with both the font and the size until your words **fit on the canvas**.
  - e) Decide which function you should put this code in:
    - i. Fill the entire canvas with your background color.
    - ii. Make the *lineWidth* 3 so when you outline the text, it will outline in a wider line.
    - iii. You will use *textAlign* values *left*, *right*, and *center* and *textBaseline* values of *top*, *alphabetic*, and *middle*. For x and y coordinates you will use 10, *can.width-10*, *can.height-10*, *can.width/2*, or *can.height/2*.
    - iv. Now, fill and outline one of your words. The words should be located in the 4 corners of the canvas, but 10 pixels from the top, bottom, right, and left (depending on which corner). Use one of your colors for the fill, and another for the outline.
    - v. Next, fill and outline the other word. These words should also be located 10 pixels from the top, bottom, right, and left, but they should also be centered along the edge of the canvas. You will use the same 2 colors for fill and outline, but whatever your fill color was for your first 4 words should be your outline color now, and vice versa.
  - f) Now, you are going to keep track of how many times you have clicked the canvas and put text in the center of the canvas with the number of clicks.
    - i. Create a variable called *clicks*. Initialize this variable to 0 in *init()*.
    - ii. Add an event listener for the *mousedown* event and create a function that will be called when the *left mouse button* is pressed.
    - iii. In the function you created, check to see if the left mouse button was pressed, and if so, add 1 to *clicks*.
    - iv. When you draw the rest of your text, you will also draw, centered on the screen, the number of clicks followed by the word “ Click” if the number of *clicks* is 1, but “ Clicks” if the number of clicks is any other number. (You will need an *if-else* to do this). Hint: use the + sign to append “ Clicks” to the variable clicks: *clicks* + “ Clicks”

v. The outline of this should be one of your 2 colors. The fill should be the color of the background.

2. Zip your *first\_last\_canvas\_16* folder. Turn in your *first\_last\_canvas\_16.zip* file.

