Slot Machine Iterative Steps

You should attempt to design the program and develop it iteratively as a group, but if you get stuck, the following steps provide one way of doing most of the steps:

1. Fill in **workshop2.html** and **workshop2.js** (there is no css for this first exercise) with the code necessary to:
    a. draw the slot machine lever (which I call "handle" throughout this document) on the web page(without the rest of the machine yet)
    b. on click, swap the slot machine lever with the fully-pulled-down version
    c. OPTIONAL: on second click, swap the slot machine lever with the lever back in the "up" position.

The code for this first section can be found in **Workshop_2q1_sol.zip**

2. Add background graphic **machine_bg.jpg** to **workshop2.html**. Then open up **workshop2.css** and edit the CSS to position the machine and handle so that they look like the layout provided. Your "click" event from exercise 1 should still function after this.

The code for this second section can be found in **Workshop_2q2_sol.zip**

3. Add graphics. Remember that now you're stacking images on top of each other so the z-order matters!
    a. Add "heart.jpg", "clover.jpg", and "star.jpg" to the appropriate positions in your **html** and **css** files (so that they appear over slot1, slot2, and slot3 on the machine). Give these images the IDs "slot1" "slot2", and "slot3".
    b. Add the "display_start.jpg" image so that it appears in the correct spot on the machine and give the ID "machine_text" ("display" is a reserved word in too many places so we'll avoid calling it that!).

The code for this third section can be found in **Workshop_2q3_sol.zip**

4. Add code to the "click" event handler that you wrote for the slot machine lever in step 1 so that:
    a. Your DOM elements "slot1", "slot2" and "slot3" will each get replaced with a random image selected from "heart.jpg", "clover.jpg", and "star.jpg". TIP: You might want to set up an array containing the 3 image file names and then pick a random number and grab that file name from the array. Or, you could write an "if/else if/else" statement, or a "switch" statement that looks at the random number you chose (I suggest between 0 and 2) and picks the image accordingly. The method is up to you!
    b. If all 3 DOM elements get replaced with the same image (which you could track by storing the random number selected for each slot or by inspecting each slot for what image is there – again, your choice!) then change the "machine_text" DOM object to "display_win.jpg".

c. If you didn't do step c in part 1 (making clicking the "down" handle reset it), go back and do it for this exercise. Then have the "machine_text" change back to "Try your luck!!!" when the player clicks to put the machine handle back up.

The code for this fourth section can be found in **Workshop_2q4_sol.zip**

Once you have completed the above steps, you will have the basic functions of the slot machine working. The remaining elements, such as timing the image changes, are not critical to the program's logic, so we save them until last. Yes, you will have to remove a little bit of code (such as clicking to reset the machine handle) in order to make these timing elements work, but that's a part of the iterative coding process.

1. Add new event handlers to the slot machine's lever so that the "handle_over.jpg" graphic is swapped in when the user rolls over the lever. The effect should be that the lever appears to "lean forward" a little as if it's been pulled part way but not all the way. As with any standard rollover, you should also handle swapping back in the "handle_up" graphic once the lever has been pulled. Note that this may cause some funny behavior with your current "click" handler, in terms of the slot machine lever remaining down until the user's mouse leaves the area. There are other mouse events that you can use to try to fix this. Properly handling rollovers is one of those tasks that seems easy but there's usually some fine-tuning necessary!

2. Use *setTimeout* (see http://www.w3schools.com/js/js_timing.asp) within one of your handlers for the slot machine lever so that when the lever is "pulled", it stays down for a second or two and then comes back up.

3. Use *setInterval* in conjunction with *setTimeout* (or a series of *setTimeout* methods if you prefer) to randomly change the images in slot1, slot2, and slot3 so that the machine looks more like it's really "mixing up" the images (we're not going to go so far as to animate them, though with JQuery, you could do this).

4. Use *setTimeout* to change the message from "Winner!!!" back to "Try Your Luck!" a few seconds after the player wins.