CPSC 1045: In-Lab Exercise 11 [10 marks]

Resources

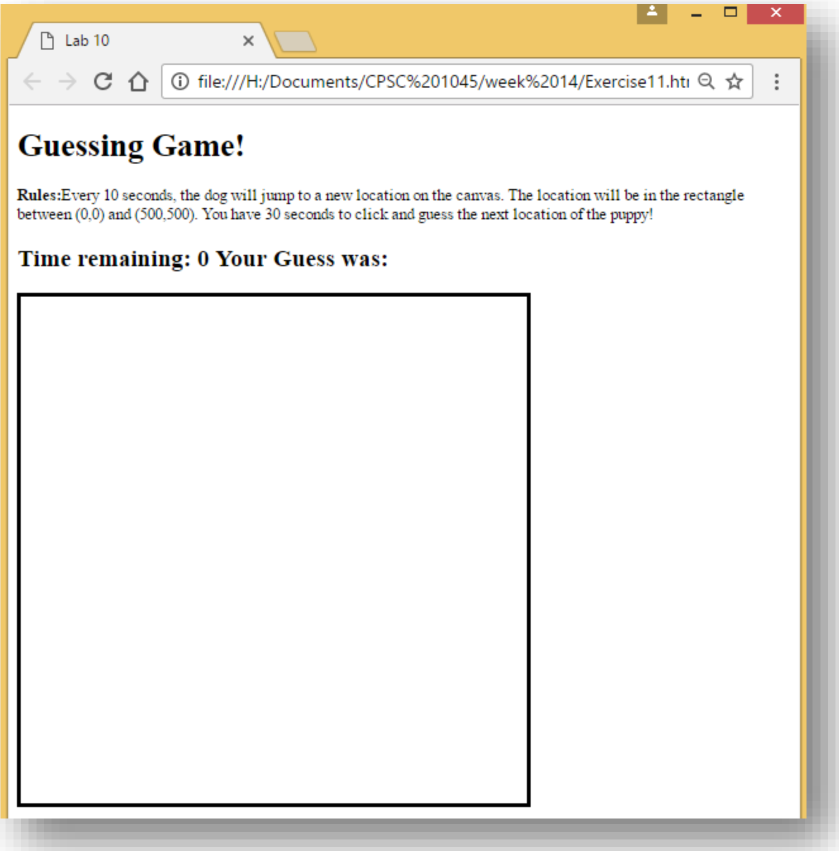
- http://www.w3schools.com/jsref/met_win_setinterval.asp
- http://www.w3schools.com/jsref/event_clientx.asp

Overview

For the last exercise of the semester, you are going to create a simple guessing game using canvas. You will use the setInterval method to create a timer, the event object to detect the mouse location on the screen, and the canvas drawImage function to draw a simple graphic on the canvas. Be careful to follow the lab instructions very closely, step-by-step, to ensure your game works properly when you're finished.

Lab work

1. Download the puppy.png file from D2L, and make sure to save it into the same folder where you create your HTML and JavaScript files. This image is 50px tall and 33px wide. Create your HTML file to look like the page below, where the canvas size is 500 by 500.



Guessing Game!

Rules: Every 10 seconds, the dog will jump to a new location on the canvas. The location will be in the rectangle between (0,0) and (500,500). You have 30 seconds to click and guess the next location of the puppy!

Time remaining: 0 Your Guess was:

2. In your JavaScript file create the following global variables:
    1. the typical canvas and context variables
    2. x and y, set to be the coordinates of the center of the canvas
    3. time, initialized to 10
    4. guessX and guessY, initialized to 0 (these will represent the user’s guess)
    5. var puppy = new Image();
       puppy.onload = drawPuppy;
       puppy.src = "puppy.png";

3. Now create the function called drawPuppy(). This function should do the following steps, in the following order, to draw the puppy on the canvas:
    1. Clear the entire canvas
    2. Fill the entire canvas with your favorite color
    3. Begin a new path, draw the puppy image object at the x,y coordinates, and close the path

4. Create a function called randomize(). This function should reset the x and y global variables to **each be a new number** between 0 and 500.

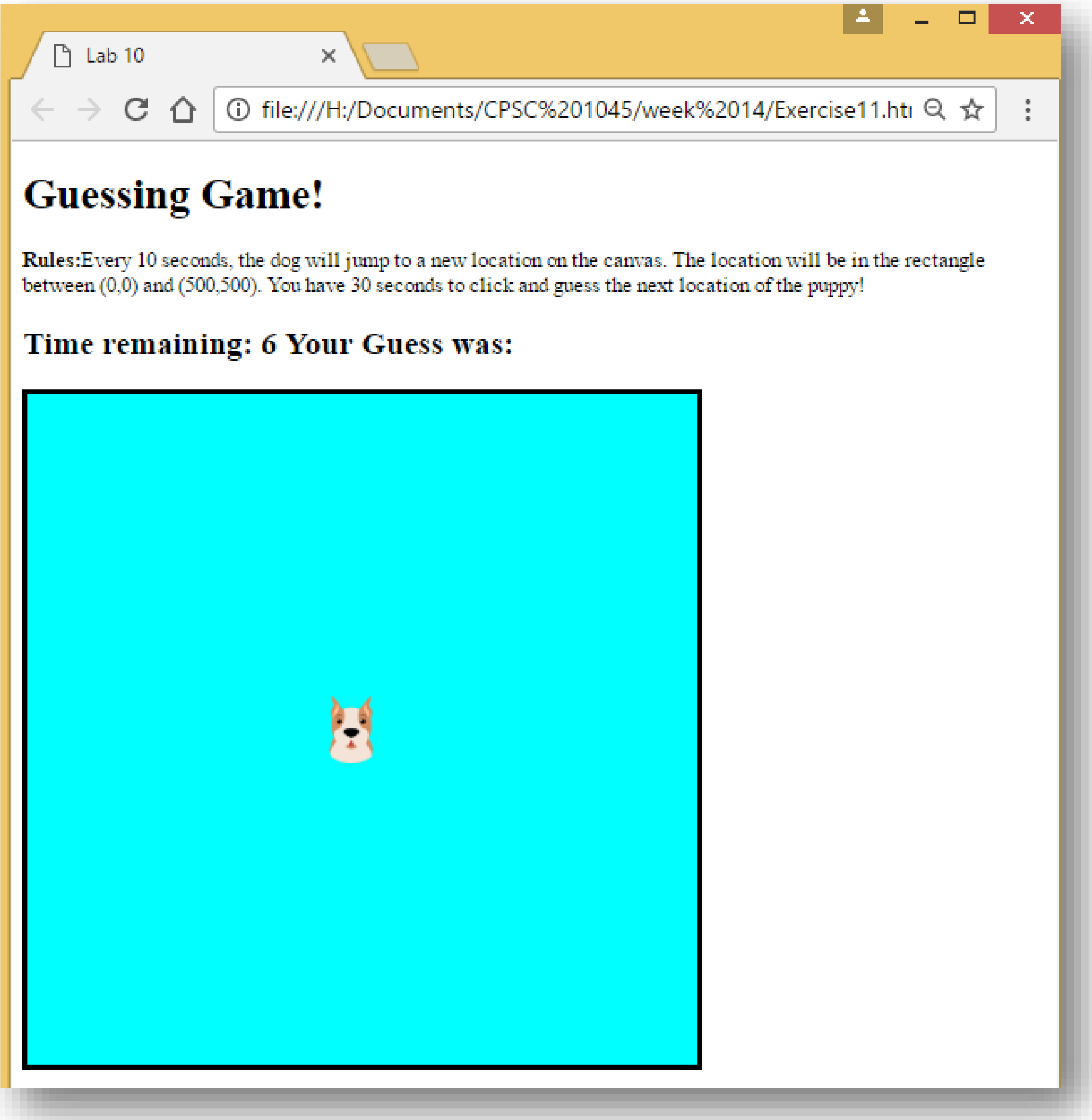
5. Create a function called storeGuess(event). This function should use the event object to store the mouse cursor’s current location as the user’s guesses (guessX and guessY) using event.offsetX and event.offsetY.

6. Add an onclick event to the canvas element, that calls the storeGuess(event) function.

7. Create a function called checkGuess(). This function should check if the user’s guessX and guessY are within the current location of the puppy image. If the user’s guess overlaps with the puppy image, you should print “Correct!” to the screen, otherwise you should print “Incorrect!” to the screen.

8. Finally, create a Timer using the setInterval function. This timer should occur every second and it should do the following things.
    1. Update the timer display and the time global variable to count down from 10 to 0
    2. When the timer reaches 0, the timer should be reset back to 10, the randomize(), drawPuppy(), and checkGuess() functions should all be called.

9. Your game should now work, and your final game should look something like this:



Guessing Game!

Rules: Every 10 seconds, the dog will jump to a new location on the canvas. The location will be in the rectangle between (0,0) and (500,500). You have 30 seconds to click and guess the next location of the puppy!

Time remaining: 6 Your Guess was:

10. You can see a working version of the game at: http://mylinux.langara.bc.ca/~jmckeescott/1045/lab10.html

11. When you have finished, demonstrate to the instructor or the Lab Assistant that your game works correctly.