Kelly Kuhn

N220

July 5th, 2023

**Lab 2 – Animation and Conditional Operators Exercises**

<https://rkkuhn.github.io/N220Spring2023/>

**Algorithm 1st Project - Counter**

**Requirements:** Create an application that outputs a number to the console for every draw call of P5. This number should start at zero and increase by one every frame.

**Expected Output:** The console will start at 0 and continue counting indefinitely.

**Sudo Code:**

Create the global vari i set to 0

Create the function setup for canvas size

I decided on a square 400 by 400

Create the function to take the global variable and add 1

Console.log(i++)

**Algorithm 2nd Project – Puck Slide**

**Requirements:** Canvas size: 400x300

**Requirements:** Create an application that draws a circle where the mouse is at. (Use the P5 variables mouseX and mouseY for this). When the mouse is on the right half of the canvas, draw the circle in red. When the mouse is on the left side of the canvas, draw the circle in blue.

**Expected Output:** The circle will change colors depending on which side of the canvas it is located. The right half is red, and the left side is blue. The background color is yellow.

**Sudo Code:**

Setup canvas size 400x300

Var diameter = 10 (creating the size of the ball. I preferred smaller so the user can see the colors change)

Use the:

if the statement to create the color for the left and right sides of the canvas

Fill color for mouseX>200 (255,0,0)

Add the circle argument (mouseX, mouseY, diameter)

ELSE state

Color fill (0, 0, 255)

Add the circle argument same as above

**Algorithm 3rd Project – World Wrap**

**Requirements:** Canvas size: 400x300

Create an application that draws a circle where the mouse is at. (Use the P5 variables mouseX and mouseY for this). When the mouse is on the right half of the canvas, draw the circle in red. When the mouse is on the left side of the canvas, draw the circle in blue.

**Expected Output:** A circle will move across the middle of the page, and when it comes to an end, it will wrap around to the starting point and move along the x-axis until it reaches the end of the canvas. “Rinse and Repeat.”

**Sudo Code:**

Create global variables

circleX = to null

radius = 25 (creates a nice round circle)

Set the function statements

createCanvas (800,600)

assign circle = 0

fill color is (124,252,0) “green-ish”

draw the circle (circle, height/2, radius\*2)

Add 5 to the value of the circle to ensure that when it gets to the end, it will wrap around

circle=(circle+5)%width;

***Reglection:*** This assignment was rather interesting. Starting with the first one was rather simple. Then I had to research the second one and look back at my notes. By the time I hit the 3rd project, I was doing a lot of research. I need help getting my landing page to work with the three files. I am still tinkering with that and trying to figure it out. At first, I thought I had it, it worked on my computer, but when I went to the .io, it stopped working. Researching google and using my html stuff to figure out what I coded incorrectly. My P5 .js files all work.

**Actual Code:**

// Author: Kelly Kuhn

// Class: N220

// Date: 7-5-2023

// Create an application that outputs to the console every

// draw call to P5. This number should start a zero and

// increase bye one every frame.

// Global variable

let i = 0;

// Creating the size of the canvas

function setup() {

createCanvas(400, 400);

}

function draw() {

// This is the function that starts at zero and continues to

// count in an infinite loop.

console.log(i++)

}

// Create an application that draws a circle where the mouse

//is at. (Use the P5 variables mouseX and mouseY for this).

//When the mouse is on the right half of the canvas, draw the

//circle in red. When the mouse is on the left side of the

//canvas, draw the circle in blue.

// Creating the size of the canvas and background color yellow

function setup() {

createCanvas(400, 300);

}

var diameter = 10; // size of the circle

// Function draw created

function draw() {

background(255, 255, 0); //background color is red

// If statement showing mouseX>200 (1/2 of the screen)

// then the color is blue, ELSE the color on the other

// side of the screen is red.

if(mouseX>200){

fill(255, 0, 0);

circle(mouseX,mouseY,diameter);

}

else{

fill(0, 0, 255);

circle(mouseX,mouseY,diameter);

}

}

// Create a canvas 800px x 600 px.

// Make an application that starts drawing a ball on the left

// hand of the screen - make it move to the right 5 pixels

// per frame.

// Write the code so that when the circle reaches 800 pixels

// to the right, move the circle to the far left of the screen.

// Global Variables

var circleX; //x variable position of the circle is null

var radius = 25; //radius of the circle

function setup() {

//creating a 800x600 canvas

createCanvas(800,600);

//assigning 0 as x coordinate for the circle

circleX=0;

// Using as fill lime greenish

fill(124,252,0);

}

function draw() {

// Using a light blueish background

background(176,230,250);

// Drawing a circle at x=circleX, y=height/2 (middle) and

// with diameter=radius\*2

circle(circleX, height/2, radius\*2);

//adding 5 to the x value of circle and if it reaches width (800), wrapping

//around from 0. So that circle will appear on far left when it reaches far right

circleX=(circleX+5)%width;

}