Kelly Kuhn

N220

July 6th, 2023

**Lab 3 – Loops Exercises**

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

**Algorithm 1st Project - FizzBuzz**

**Requirements:** A loop, oriented horizontally, draws 25 black circles to the screen.

However,

1. When the iterand is divisible by 3, draw a purple circle instead
2. When the iterand is divisible by 5, draw a green square instead
3. When the iterand is divisible by 3 AND 5, draw a blue square instead

**Expected Output:**

A blue background with black and green circles

Description automatically generated

**Sudo Code:**

Create the canvas (1000,1000)

Function()

Create background color light blue (135, 206, 250)

Create the variable y for the function var y = 100

For (let i=0; i<25; i++)

Set x variable to x = i\*50 // this takes the value in “i” and multiplies it to 50

Start if state i modulus 5 === 0; this checks whether its two operands are

Equal and returns a Boolean result.

Fill color is green (0, 255, 0)

Square (x, y, 50)

Else if (i % 3 === 0)

Fill(153, 31, 240)

Used the ellipse command to make the circle (x+25, y+25, 50,50), making

the ellipse is the same size as the squares

Else if (i % 3 == 0 && I % 5 == 0) // stating that if both conditions are met run

this if statement

fill is blue (0, 0, 255)

square (x, y, 50);

Else is the final option if all other conditions are not met run this code

Fill (0) – color black

Ellipse (x+25, y+25, 50,50)

**Algorithm 2nd Project – Pyramid**

**Requirements:** Canvas size: 400x400

**Requirements:** Using nested loops (one within the other) recreate this composition with only one line having a drawing command on it (rect/circle/line/etc..).

**Expected Output:**

A red squares in a row

Description automatically generated

**Sudo Code:**

Setup canvas size 400x300

Function draw()

Background white

Stroke(255)

Color fill is red (255, 0, 0)

Using size 40 and grid\_size to create the blocks

For loop (var i=0; i<4; i++) – only looping 4 times

For (var j=0; j<=I; j++) – this sets up the looping between j and I grid\_size

Draw a rectangle with (j\*grid\_size, i\*grid\_size as top let coordinates and

Grid\_size as length and width

**Algorithm 3rd Project – Circle inside of Circles**

**Requirements:** Canvas size: 400 x 300

Use a loop to recreate this composition with only one line having a drawing command on it (rect/circle/line/etc..).

**Expected Output:**

A black and white circular pattern

Description automatically generated

**Sudo Code:**

Function setup ()

createCanvas(400, 300)

background (211, 211, 211) gray-ish

function draw()

background (211, 211, 211)

use noFill so the lines will show up on the code otherwise, you get one big white

circle

for (var i=0; i<15; i++) – this is the loop to create 15 circles starting at the center.

Circle (200, 150, 1\*10) – this creates the circle starting at the center and increases by 10

each loop until it ends.

***Reflection:*** I think the hardest one to code was the first assignment, “FizzBuzz.” I had to research on the internet to get my circles, squares, and colors to work correctly. I did use Stack Overflow. I enjoyed the challenge and kept at it until I found a working solution. The “Circles Inside of Circles” was easy. I remembered it from our lecture. For the “Pyramid” one, I had to laugh when I saw the comment, “Use nested loops”. Professor Harris docked me on my final project for using nested loops. I am pretty good at writing nested loops. I know he wanted me to expand my knowledge and try to code it in a different way.