*Static Piece Rationale*

My static piece of a canvas drawing originally started out as a guitar as the shapes were diverse, including using the arc functions to create the body. This eventually evolved into the finished piece I present today. The guitar shaped body remains, it seemed this was a transferrable part of canvas as it could represent many animals well, such as sheep, bears and, as I chose, cats.

I acquired my code sources from the internet with the first being from “MDN Web Docs”. This helped me understand the importance of beginning and closing paths to ensure the desired shapes were achieved. Furthermore, I also found it useful to include “x, y, radius, startAngle, endAngle, anticlockwise” in the comments so I could quickly refer to which part I was editing.1

My goal was to create a recognizable cat, with paths wrapped in functions for easier compartmentalization. This would mean it would be easier for me, and future readers of my code, to move them about. One part I found particularly difficult was adjusting to the “new” co-ordinates system, with “x” being in the upper left-hand corner. Thus, a solution for this issue was to put in the notes what raising and lowering the integers would do. This saved a lot of time as my guesses were more precise.

In accordance with the guidelines, I believe I have demonstrated best practice by including comments, to make it easier for people to follow the structure.

In addition to this, I downloaded a “Beautifier” to automatically indent my code so that it looked neat and easy to follow. This made it a lot easier to follow when I came back to correct mistakes it not only looked visually pleasing, but it was much easier to find certain parts.

*Animated Piece Rationale*

My animated piece of a canvas drawing came about after I was browsing “HTML5 Canvas Tutorials” and came across a rectangle that moved from left to right. This evolved into making two rectangles move at once. As one rectangle was blue, and one was red, I decided to use the colours of the rainbow as my style guide. This would create a colourful piece and it too would demonstrate my animation skills.

I acquired my code from sources such as “MDN Web Docs” which helped to give me a clear explanation over the code. For example, from line 304, I found it interesting to learn about this method as it would help me reach a wider range of audiences, no matter how slow their PC was, due to the repaint only occurring when the last viewing was completed. As soon as I learnt about this I was eager to include it.

My goal was to not simply take an animation but to make it my own creatively too. It was important for me to understand all my code before submitting it. An example of this was fully understanding some of the harder mathematical elements. However, I wanted to give myself a bit of a challenge. For example, the operator “+=”, however in my code comments I made sure I explained it.

In accordance with the guidelines, I believe I have demonstrated best practice by making sure I fully understand the code I am using before implementing it.

*Static Piece Pseudo Code*

* Declare canvas as 400 x 450
* Declare the draw function,
* Call the canvas by its id name,
* State the context will be two dimensional,
* Set the shadow blur to around 60,
* Shadow off set on X = 0, Y = 400,

*Draw bar stool.*

* Function **cone**, (ctx, 290, 270), create triangle with three lineTos, (195 & 450, 100 & 270, 195 & 270),

Fill = #909090,

End **cone**

*End bar stool.*

*Draw head.*

* Function **head**, (ctx, 200, 110), create circle with the arc function, (x, y, 60, 0, 2 \* Math.PI)

Fill = #B16C80,

*End head.*

*Draw left eye.*

leftEye: create circle with arc function, (177, 85, 3, 0, 2 \* Math.PI)

Fill = #000,

*End left eye.*

*Draw left eyebrow.*

leftEyebrow: create crooked line, (210 & 74, 230 & 79),

*End left eyebrow.*

*Draw right eyebrow.*

rightEyebrow: create crooked line, (185 & 75, 165 & 77),

*End right eyebrow.*

*Draw nose.*

nose: (190 & 87, 220 & 100),

noseSecond: (200 & 80, 190 & 110),

*End nose.*

*Draw right eye.*

rightEye: create circle with arc function, (217, 85, 3, 0, 2 \* Math.PI),

*End right eye.*

*Draw left ear.*

leftEar: create triangle with three lineTos, (190 & 50, 150 & 75, 150 & 30),

*End left ear.*

*Draw right ear.*

rightEar: create triangle with three lineTos, (250 & 75, 200 & 50, 235 & 30),

*End right ear.*

*Draw body.*

* Function **body**, (ctx, 200, 180), create circle with the arc function (x, y, 90, 0, 2 \* Math.PI),

*Draw mouth.*

Mouth: (185 & 105, 210 & 100),

*End mouth.*

*Draw right leg.*

rightLeg: (170 & 380, 160 & 250),

*End right leg.*

*Draw left leg.*

leftLeg: (320 & 340, 230 & 250),

*End left leg.*

*Draw right arm.*

rightArmone: (135 & 130, 90 & 170),

rightArmtwo: (90 & 170, 70 & 180),

*End right arm.*

*Draw left arm.*

leftArmone: (315 & 160, 260 & 125),

leftArmtwo: (336 & 161, 313 & 159),

*End left arm.*

*Draw chest.*

chest: create circle with arc function, (x, y, 60, 3, 3 \* Math.PI),

*End chest.*

*End body.*

*Draw cocktail.*

* cocktailTop: create triangle with three lineTos, (53 & 144, 80 & 142, 68 & 160),

Fill = #2D0157,

cocktailBottom: (72 & 180, 68 & 180),

Fill = #000,

cocktailEnd: (68 & 158, 70 & 180),

Fill = #6139FF.

*End cocktail.*

*Animated Piece Pseudo Code*

* Declare canvas as 380 x 380,
* Assign variables to integers (x = 3, y = 20, speed = 5),

*Start animate function*

* Function **animate,**

Perform animation,

Call “**animate**” function,

Update animation before next repaint,

3 with the addition of 5 (x += 5)

If 3 is less than 5, which is true, then the outcome will be true. If it weren’t, it would be false;

Then if 3 is greater than or equal to 10;

*End animate function.*

D

*End animate function*

* Declare the draw function,
* Call the canvas by its id name,
* State the context will be two dimensional,
* Clear the rectangle (making it transparent)

Animate the animation.

*Static Piece Critique[[1]](#endnote-1)*

The subject of this critique is my static piece of HTML Canvas. I believe that it fulfils the brief as it **demonstrates knowledge**; for example, I learnt a lot of the meaning of some code I had just mindlessly used before like operators such as “+=” meaning “when followed by an expression with a numerical value, will add the value of that expression to the value of the variable to the left of the operator, and assign the result to that variable.”[[2]](#endnote-2) This gave me a greater scope when creating my examples as it actually made sense what I was writing.

I also believe I demonstrate **functionality** well too as from the pseudo code to the final product I have stayed true to my idea. In addition to this, I used best practice methods such as using camelCase which is industry standard and makes it easier for me or any future readers to follow the structure of the code. In addition to this, I also ran my code through the W3C Validator and it came back with no issues which was both a relief and a pleasure as it meant learning all the new code worth it.

One method I would do differently next time is **guesswork.** Although it was hard for me to get my head around this new co-ordinate system, endlessly typing random integers with the hope at SOME point the arc would move where I wanted it to, became tiresome and unrewarding. Adding code comments to remind myself whether to raise the number or lower it if I wanted the object to go downwards really helped me as it stopped a lot of time wasting due to guesswork.

The expected result was very different compared to what it is now. I started with drawing a guitar, and you can probably see that the cocktail cat’s body is reminiscent of a guitar’s, however did not realize that using the rotate function would affect every part. Therein lay the issue that I would have to re-do all the co-ordinates. I did not use pseudo code for the guitar which meant I was very unprepared which is why I used pseudo code for the cocktail cat.

I particularly liked the seemingly simple shapes such as circles and triangles being used as it was interesting to think about these elementary shapes being complex.

*Animated Piece Critique[[3]](#endnote-3)*

The subject of this critique is my animated piece of HTML canvas. One key development in this process was coding techniques. Learning to compartmentalize sections using functions was groundbreaking for me as it helped me edit the piece. For example, knowing the code for the eyes would be in the head function saved a lot of time trying to scroll for ambiguous words.

In addition to this, another technique I used was simply practicing. When I was added multiple moving rectangles, I couldn’t tell which bit to keep repeating without them overlapping. However, by using pseudo code and writing it out in plain English, I was easily able to understand it.

One issue I had with this task was either not doing enough or being too creative. It was important for me to read the brief several times to not go too far. On the other hand, it was difficult to fine a fine line between simply using someone else’s code and being creative. I was originally only going to have one moving rectangle however I used techniques such as making a mind map which is how I came up with the idea of doing the colors of the rainbow.

Something I would do differently next time would be to include a button for the audience to refresh it manually rather than having to reload the page. This would add an element of user interaction. In addition to this it would give me the chance to learn more difficult code and challenge myself.

*­*

­

1. Studyskills.southwales.ac.uk. (2018). [online] Available at: http://studyskills.southwales.ac.uk/media/files/documents/2013-08-21/How\_to\_Write\_a\_Critique.pdf [Accessed 10 Mar. 2018]. [↑](#endnote-ref-1)
2. Codecademy. (2018). *What is the usage of += ?*. [online] Available at: https://www.codecademy.com/en/forum\_questions/5455392252f8631c950028c1 [Accessed 10 Mar. 2018]. [↑](#endnote-ref-2)
3. Studyskills.southwales.ac.uk. (2018). [online] Available at: http://studyskills.southwales.ac.uk/media/files/documents/2013-08-21/How\_to\_Write\_a\_Critique.pdf [Accessed 10 Mar. 2018]. [↑](#endnote-ref-3)