Introduction to Programming II Project Log

Project title:	Drawing App
Topic:	Finished the Stamp Tool's modifications (14th August 2023)

What progress have you made this topic?

I have completed all of my desired modifications for the Stamp Tool. Problems I mentioned in the previous report (~~uploaded pictures override local stamps, and inefficient method of coding which only allow users to upload at most 5 pictures~~) and as well as planned change (~~allowing uploaded stamps to retain their aspect ratio~~) are all solved and implemented.

What problems have you faced and were you able to solve them?

• Uploaded pictures override previous/local stamps: Apparently this issue is caused by how I reinitialise the stamp selection dropbox. With the context of the chosen stamped will be stored in var tempImg, reinitialising every drop box content (that are stored in an array) will cause all of the previously loaded 'stampList' array elements to become undefined despite showing up normally in the dropbox. Solution: Instead of using loops to reinitialise the whole drop box, I'll be only using `stampDropbox.option(str(stampList[stampList.length-1]));` to initialise the most recently uploaded stamp.

```
for (var i=stampList.length-1; i<stampList.length;i++){
    stampDropbox.option(str(stampList[i]));
}
alert("stamp added");
```


Before

• Inefficient codings: Previously, it can be seen that I hard-coded the number of space available to upload custom stamps, and uses too many if function which resulted in the codes being too long. Solution: By making an array to store all stamps in an ordered manner and

```
let customNameStart = 'custom';
let customNameEnd = stampList.length -4;
stampList.push(str(customNameStart+customNameEnd));
```

I am able to shorten the codes.

After

```
tempImg;
var pickedStamp = selectedStamp();
for (var i = 0; i<stampList.length;i++){
    if (pickedStamp == stampList[i]){
        tempImg = stampImgList[i];
    }
}
```

This works by having the array that store all stamps, ordered in the same manner as the array that store the drop box content

Before
After

allows me to generate strings for the drop box content dynamically depends on how many custom stamps uploaded.

• Let stamps to retain their original aspect ratio: Previously, I hardcoded the size of the stamps. Which cause the stamps to all be square. via ⌐

```
//To set the size of images.
var thickness = sliderSelected();
if (thickness > 0){
    var sW = map(thickness,0,50,10,400);
}
else if(thickness <= 0){
    var sW = 1;
}
var stampSize = sW;
```

and 'image(tempImg,0,0,stampSize,stampSize)'

OLD

Solutions: Rather than using "sW" as a fixed length value, I change it to be a value multiplier. Additionally, I added a section of codes that will allow stamps of non-square aspect ratio (eg 16:9, 4:3, 9:16, etc) to retain their aspect ratio while resized to fit the canvas better.

Before   After

```
let lengthMultiplier = 228;
if (tempImg.height>lengthMultiplier){
    tempImg.width = tempImg.width*(lengthMultiplier/tempImg.height)
    tempImg.height = lengthMultiplier;
} else if (tempImg.height<lengthMultiplier){
    tempImg.height = tempImg.height*(lengthMultiplier/tempImg.width)
    tempImg.width = lengthMultiplier;
}
```

```
//To set the size of images.
var thickness = sliderSelected();
if (thickness > 0){
    var sW = map(thickness,0,50,0.04,2);
}
else if(thickness <= 0){
    var sW = 0.04;
}
var stampSize = sW;
```

and

NEW   'image(tempImg,0,0,tempImg.width*stampSize,tempImg.height*stampSize);'

Are you on target to successfully complete your project? If you aren't on target, how will you address the issue?

Personally, I have completed this project. I may make small changes in the future if I realised there issues with the presentation of codes or small bugs.