MuW201 TANGIBLE COMPUTING

PROJECT 01 – TIME BASED MEDIA

RAFIF AL HASSEN
180430

Description

As Dostoyevsky once said, "To think too much is a disease". As humans, we tend to overthink situations which results in a multitude of unnecessary stress as we get lost in our thoughts. We tend to focus on the bad and ignore the good. This creates a lot of internal conflict between ourselves and our minds. Do you ever just want to shut your mind off for a moment of peace?

Not only do we create inner conflict but we can also slow down or speed up time-based on our thoughts. I have always been an over-thinker. I always find myself thinking the worst of situations and this results in a lot of stress on my part. This makes time pass slowly as I begin to get lost in my thoughts mentally and emotionally. Overthinking is a disease of the mind that can eventually drive you crazy.

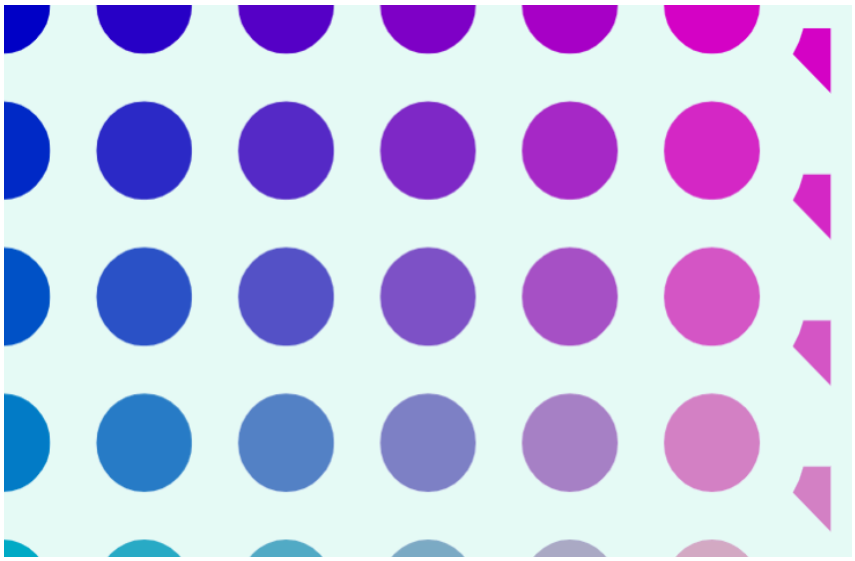
Ever since I have become an anxious person, I have found myself overthinking constantly. Thinking began to become a method of how I measured time. My thoughts defined time. The more I get lost in emotional thoughts, the slower I find time to pass. The less in-depth I think, the faster I feel time passing. With each thought, an emotional weight is carried and depending on how negative that weight is, time will pass either faster or slower. I felt encouraged to explore the way I could illustrate how overthinking can define time as it is something many people including myself struggle with daily.

This exploration was based on the concept of measuring time through human thoughts. A random thought is generated as you click within the boundaries of the ellipse. These thoughts are generated based on different sets of arrays. The use of color in this sketch is tied down with emotions as our emotions are usually associated with color depending on the thought itself. As the mouse is pressed and a new thought has generated the canvas as well as the thought changes color. The thoughts also have a lifespan to replicate the way our thoughts either linger or fade away after a certain amount of time. The font size is also changing within the boundaries of 80-150 to keep the generated thoughts legible. The size also represents the strength of our thoughts. The thoughts that appear also rotate to illustrate how our thoughts are random and pretty much all over the place. I also added a 3D blur effect to replicate the disorder and the randomness of our thoughts. Furthermore, I added smileys and some punctuation marks that are randomly generated into each thought.

The main reason for incorporating the mouse pressed function was to make the sketch more interactive. This also allows the user to control the number of thoughts that appear and where they appear. Making it interactive creates an emotional interaction between the user and the sketch. However, when the user does not click within the boundaries of the ellipse thoughts are generated as they are no longer in the human 'head'.

Process

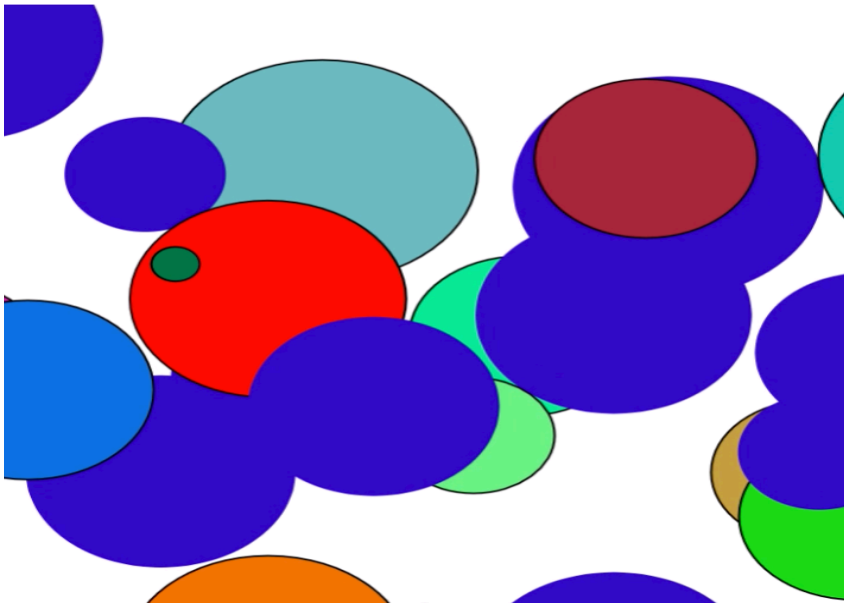
Sketch 1: Exploration of Gradual Decrease in Size



```
// Sketch Title - for MuW201 Tangible Computing, 2019
// Firstname Lastname
// GitHub Username
// Date

function setup(){

    let canvas = createCanvas(windowWidth/3,windowHeight/3)

    // Move the canvas so it's inside our <div id="sketch-holder">.
    canvas.parent('sketch-holder');
    // background(255,136,182)
}

function draw(){

    background(0)
    var numElements = 6
    textSize(10)
    var step = width/numElements
    var colorStep = 255/numElements
    textAlign(CENTER,CENTER)

for (var x = 0 ; x < numElements; x++){

    for(var y = 0; y < numElements; y++){

        // noStroke()
        stroke('#E9FBF9');
        strokeWeight(second()+10)
        fill(x*colorStep,y*colorStep,200)
        // fill(random(255),random(255),random(255))
        // var string = x + "," + y
        // line(x*step,y*step,step,step)
        rect(x*step,y*step,step,step)
        ellipse(x*step,y*step,step,step)
        arc(x*step+50,y*step+50,80,80,0, PI + QUARTER_PI, PIE);


        }

    }
}
```

Link: file:///Users/rafifalhassen/Documents/GitHub/Tangible-Computing-2019/students/rafif_alhassen/week2/p5%20template%205/index5.html

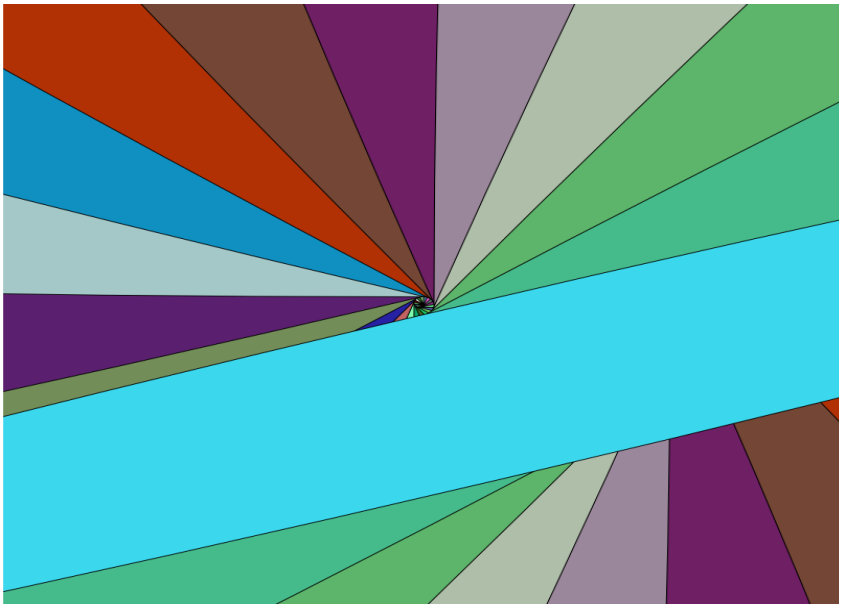
Sketch 2: Exploration of Mouse Pressed & Class Jitter



```
let con = [];
function setup() {
let canvas = createCanvas(windowWidth/3, windowWidth/3)
  canvas.parent('sketch-holder');

  for (let i = 0; i < 25; i++) {
    con.push(new Jitter());
  }
}

function draw() {
  background(255);
   for (let i = 2; i < con.length; i++) {
  con[i].move();
  con[i].display();

}
}


class Jitter {
  constructor() {
    this.x = random(width);
    this.y = random(height);
    this.diameter = random(150, 20);
    this.speed = 5;
  }

  move() {
    this.x += random(-this.speed, +this.speed);
    this.y += random(-this.speed, this.speed);
  }

  display() {
    for (var angle=1; angle < TWO_PI; angle+= TWO_PI*con){
      push()
    translate(150,10)
    // rotate(angle)
    fill(50,10,200)
    noStroke()
    ellipse(this.x, this.y, this.diameter, this.diameter);
    line(this.y, this.x, this.diameter, this.diameter);
      pop()

        if (mouseIsPressed){
    fill(random(255),random(255),random(255))
        ellipse(this.x, this.y, this.diameter, this.diameter);
  }
   }
  }
}
```

Link: file:///Users/rafifalhassen/Documents/GitHub/Tangible-Computing 2019/students/rafif_alhassen/week3/p5%20template%202/index1.html

Sketch 3: Exploration of Drawing Different Shapes & Changing Colors Each Time



```
function setup(){

  createCanvas(800,800);
  background(255)
  let flower = createFlower()
  push()
  translate(width/2,height/2)
  drawFlower(flower)
  pop()


}

function createFlower(){

  let flower = {}
  flower.numPetals = 20
  flower.innerRadius = 50
  flower.outerRadius = 100
  flower.rotation = PI/10
  flower.scale = 10
  flower.color1 = [123,36,222]
  flower.color2 = [123,36,222]
  flower.posX = width/2
  flower.posY = height/2
  flower.speed = 2
  // drawFlower (numPetals,innerRadius,outerRadius,rotation,scale,color1,color2,positionX,positionY)

  return flower

}



function drawFlower (flower){

    fill(flower.color1)
    //ellipse(positionX,positionY-innerRadius, (outerRadius-innerRadius)/2, outerRadius-innerRadius)
    //fill(random(255),random(255),random(255))
    //ellipse(positionX,positionY+innerRadius, (outerRadius-innerRadius)/2, outerRadius-innerRadius)

for (var angle = 0; angle < TWO_PI+4; angle+= TWO_PI/flower.numPetals){

  console.log(angle)

  push()

  translate(flower.positionX,flower.positionY)
  rotate(angle)

  ellipse(0,flower.innerRadius,(flower.outerRadius-flower.innerRadius)/2,flower.outerRadius-flower.innerR
  fill(random(255),random(255),random(255))
  ellipse(0,flower.innerRadius,(flower.outerRadius/flower.innerRadius)/4,flower.outerRadius-flower.innerR
  fill(random(255),random(255),random(255))
  ellipse(0,flower.innerRadius+,(flower.outerRadius-flower.innerRadius)/4,flower.outerRadius-flower.inne

  pop()
}

    fill(0)
    ellipse(flower.positionX,flower.positionY,flower.innerRadius,flower.innerRadius)
}
```

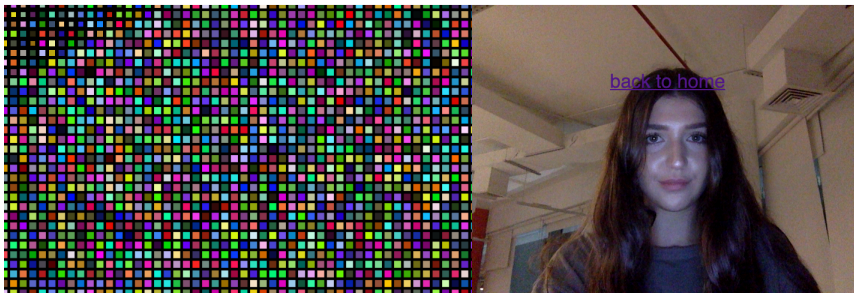
Link: file:///Users/rafifalhassen/Documents/GitHub/Tangible-Computing-2019/students/rafif_alhassen/week3/p5%20template%203/index1.html

Sketch 4: Exploration of Mouse Pressed & Moving Shapes

```
// Sketch Title - for MuW201 Tangible Computing, 2019
// Firstname Lastname
// GitHub Username
// Date
let rad = 60;
let xpos, ypos;

let xspeed = 3;
let yspeed = 2.2;

let xdirection = 3;
let ydirection = 1;

function setup(){

  createCanvas(720,400);
  // background(255)
  noStroke();
  frameRate(30);
  ellipseMode(RADIUS);
  xpos = width / 4;
  ypos = height / 2;

}


function draw(){
  background(255);

    xpos = xpos + xspeed * xdirection;
    ypos = ypos + yspeed * ydirection;

    if (xpos > width - rad || xpos < rad) {
      xdirection *= -1;
    }
    if (ypos > height - rad || ypos < rad) {
      ydirection *= -1;
    }

    rect(xpos, ypos, rad, rad);
    fill(random(255),random(255),random(255))

    ellipse(xpos, ypos, rad, rad);
    fill(random(255),random(255),random(255))


    if (mouseIsPressed){
    fill(255)
      ellipse(xpos, ypos, rad, rad);


    }

  }
```

Link: file:///Users/rafifalhassen/Documents/GitHub/Tangible-Computing-2019/students/rafif_alhassen/week3/p5%20template%205/index1.html

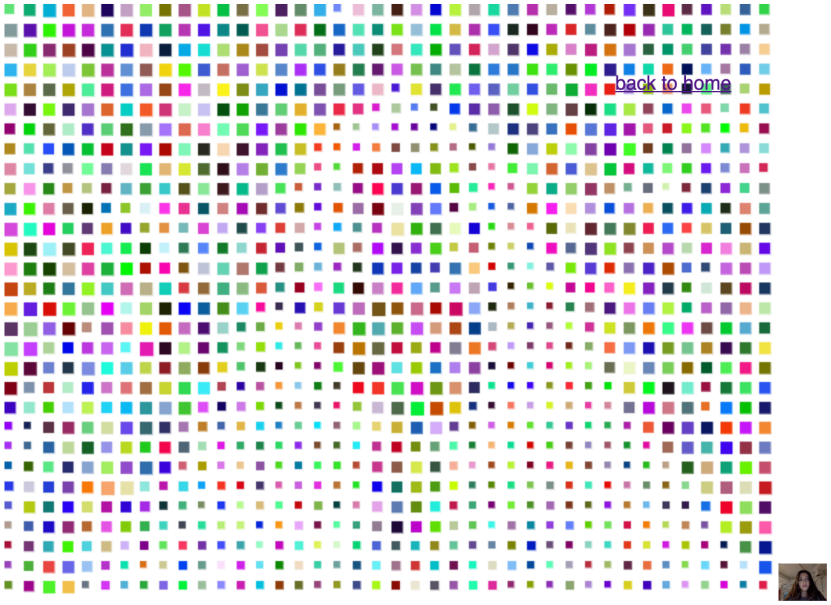
Sketch 5: Exploration with Video Input




```
// let capture;
var video;

function setup() {
  createCanvas(390, 240);
  pixelDensity(5);
  video = createCapture(VIDEO);
  video.size(320, 240);
  // video.size(width / vScale, height / vScale);

}
  // capture = createCapture(VIDEO);
  // capture1 = createCapture(VIDEO);
  // capture.size(320, 240);
  // capture1.size(320, 240);
  // //capture.hide();

function draw() {
  background(0);

  video.loadPixels();
  loadPixels();
  for (var y = 0; y < video.height; y++){
    for (var x = 0; x < video.width; x++){
      var index = (x + y * video.width)*4;
      var r = video.pixels[index+3];
      var g = video.pixels[index+4];
      var b = video.pixels[index+5];

      var bright = (r+g+b)/3;

      var vScale = 8

      var w = map(bright, 0, 255, 0, vScale);

      noStroke();
      fill(random(255),random(255),random(255));
      rectMode(CENTER);
      rect(x*vScale, y*vScale, w, w);

    }
  }

}
```

Link: file:///Users/rafifalhassen/Documents/GitHub/Tangible-Computing-2019/students/rafif_alhassen/week4/p5%20template%202/index1.html

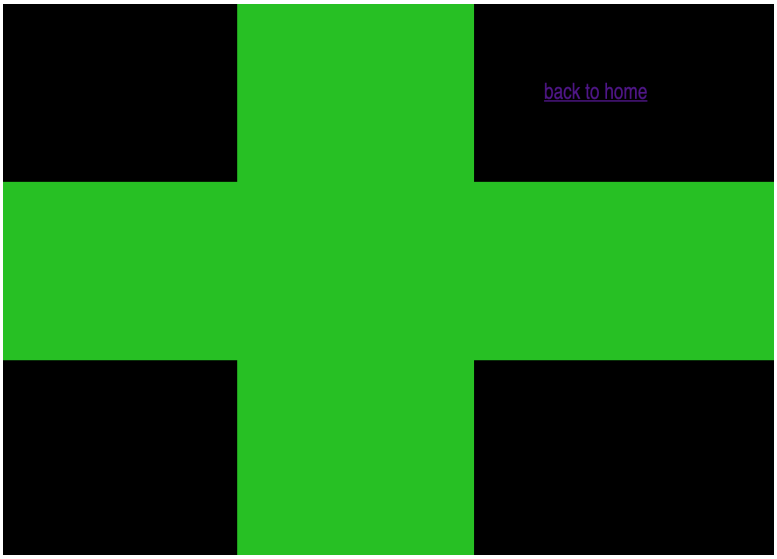
Sketch 6: Exploration with Video Input




```
// let capture;
var video;

 var vScale = 16

function setup() {
  createCanvas(640, 480);
  pixelDensity(1);
  video = createCapture(VIDEO);
  video.size(width/vScale, height/vScale);
  // video.size(width / vScale, height / vScale);

}
  // capture = createCapture(VIDEO);
  // capture1 = createCapture(VIDEO);
  // capture.size(320, 240);
  // capture1.size(320, 240);
  // //capture.hide();

function draw() {
  background(255);

  video.loadPixels();
  loadPixels();
  for (var y = 0; y < video.height; y++){
    for (var x = 0; x < video.width; x++){
      var index = (x + y * video.width)*4;
      var r = video.pixels[index+1];
      var g = video.pixels[index+2];
      var b = video.pixels[index+3];

      var bright = (r+g+b)/3;

      var w = map(bright, 0, 255, 0, vScale);

      noStroke();
      fill(random(255),random(255),random(255));
      rect(CENTER);
      rect(x*vScale, y*vScale, w, w);

    }
  }

}
```

Link: file:///Users/rafifalhassen/Documents/GitHub/Tangible-Computing-2019/students/rafif_alhassen/week4/p5%20template%204/index1.html

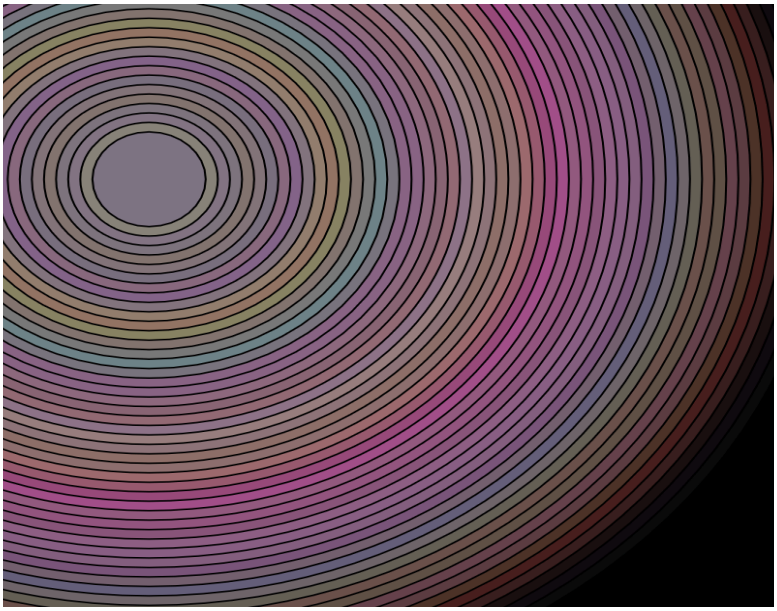
Sketch 7: Exploration with Looping




```
let y = 100;
let x = 200

// The statements in the setup() function
// execute once when the program begins
function setup() {
  createCanvas(720, 400); // Size must be the first statement
  // stroke(255); // Set line drawing color to white
  noStroke();
  frameRate(40);
}
// The statements in draw() are executed until the
// program is stopped. Each statement is executed in
// sequence and after the last line is read, the first
// line is executed again.
function draw() {
  background(0); // Set the background to black
  y = y - 1;
  if (y < 0) {
    y = height;

    x = x + 20;
    if (x < 1) {
      x = width;
    }
  }
  fill(random(255),random(255),random(255))
  rect(0, y, width+20, y);
  rect(x, 0, x, height);
}
```

Link: file:///Users/rafifalhassen/Documents/GitHub/Tangible-Computing-2019/students/rafif_alhassen/week5/p5%20template%205/index1.html

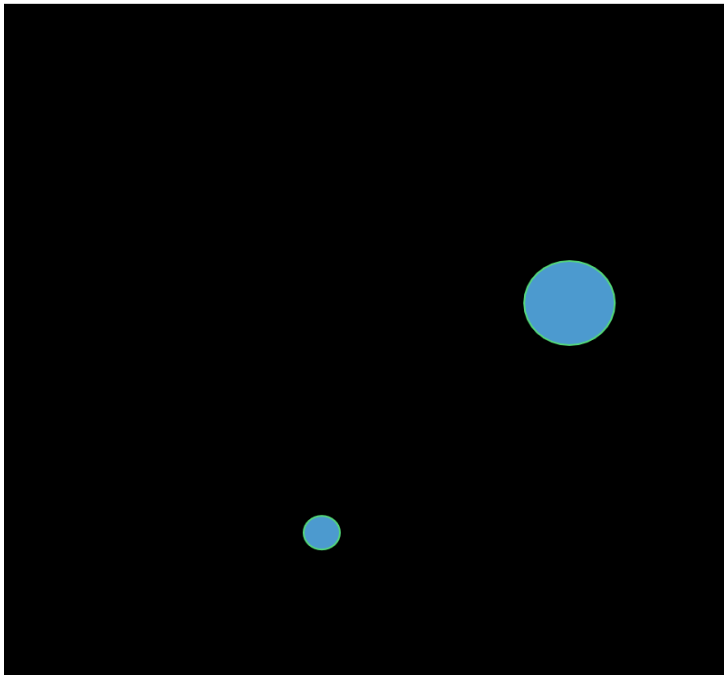
Sketch 8: Exploration with Recursion



```
function setup(){
 createCanvas(400,400)
 background(0);

 let line = {

 x1: width/2,
 y1:0,
 x2: width/2,
 y2: height,

 }

 let numlines = 28

 drawlineRecursion(line,numlines);
}

function drawlineRecursion(l,numlines){

 fill(random(255),random(255),random(255),30)
 //strokeWeight(numlines*25)
 //line(l.x1,l.y1,l.x2,l.y2
 push()
 translate(width/6,height/6)
 ellipse(10,50,numlines*25-4,numlines*25)
 pop()

 numlines=numlines -0.5

 if (numlines>2){
  drawlineRecursion(l,numlines)
  //console.log(numlines)
 }
 // line(l.x1,1.y1,1.x2,y2)
 //drawlineRecursion()
}
```

Link: file:///Users/rafifalhassen/Documents/GitHub/Tangible-Computing-2019/students/rafif_alhassen/week6/p5%20template/index1.html

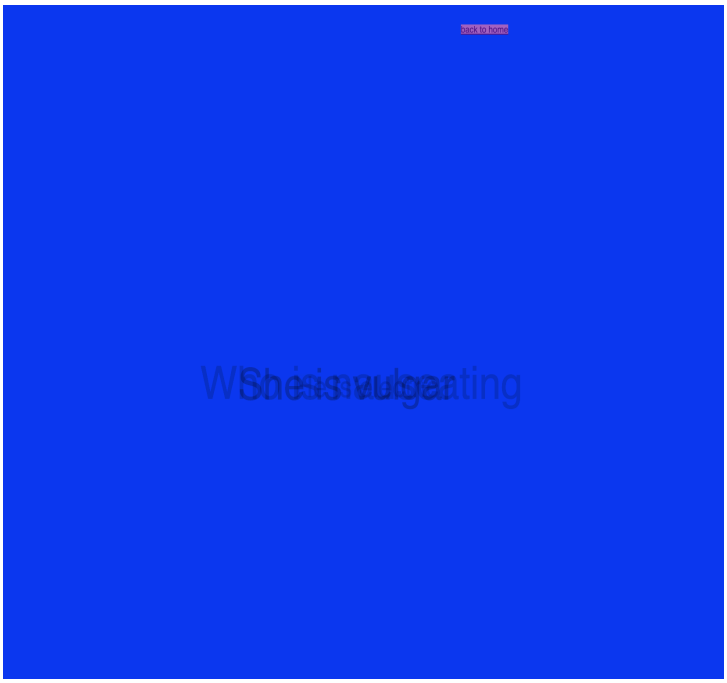
Sketch 9: Exploration of Creating Graphics (Moving Shapes)



```
let x = 200;
let y = 200;


// let thoughts = ["i am hungry", "who is texting me", "what is my dog thinking about"]
// let freeThoughts = []

function setup() {

  createCanvas(400, 400);
  stroke(0)
  background(0);
  fill(255)

}

function draw() {

  // for (var i = 0; i < freeThoughts.length; i++){
  //   let tht = freeThoughts[i]
  //   text(tht.txt,tht.x,tht.y)
  // }

  background(0);
  x += random(-5, 5);
  y += random(-5, 5);
  fill(random(255),random(255),random(255));
  stroke(random(255),random(255),random(255));
  rectMode(CENTER);
  ellipse(x, y, 20, 20);
  ellipse(y, x, 50, 50);



  // if (mouseIsPressed) {
  //   fill(255, 150);
  //   noStroke();
  //   ellipse(mouseX, mouseY, 60, 60);
  }

```

Link: file:///Users/rafifalhassen/Documents/GitHub/Tangible-Computing-2019/students/rafif_alhassen/week6/p5%20template%206/index1.html

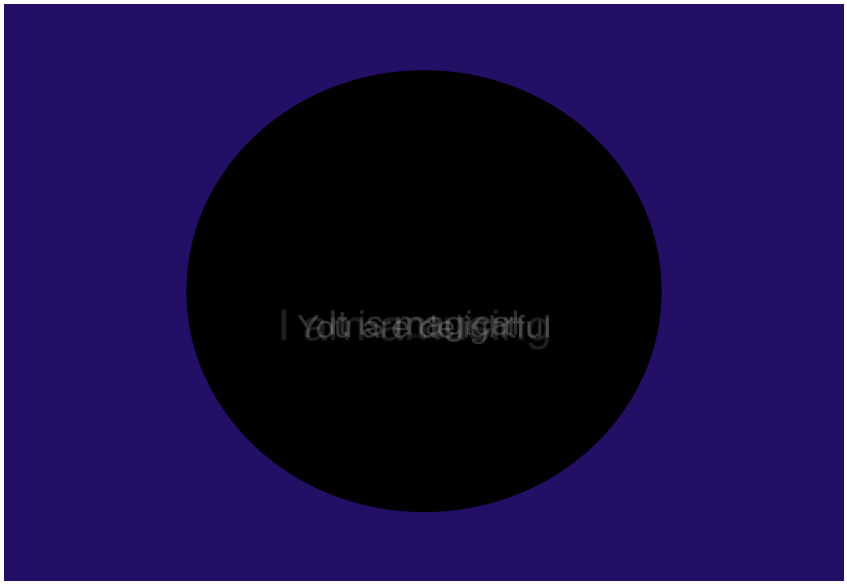
Sketch 10: Exploration with Randomly Generated Thoughts & Mouse Pressed



```
let x = 200;
let y = 200;

let fr = 30;

let thoughts = ["i am hungry", "i am stressed out", "who is texting me", "what should i have for lunch",

let firstPart = ["I am", "Who is","She is","He is","We are","They are"]

let secondPart = [" hungry", " late for class", " eye infection, miss"]

let fonts = ["Helvetica, Times New Roman, Comic Sans, Arial"]

let freeThoughts = []

function setup() {
  createCanvas(window.innerWidth, window.innerWidth);
  textAlign(CENTER)
  frameRate(fr);
  background(0);
  console.log(adjectives)
  lifeSpan = 0.2;

}
  var r = 80
  var g = 120
  var b = 255

function draw() {
  background(r,g,b);
  drainBrain()
  // fill(random(255),random(255),random(255));
  stroke(0);
  frameRate(fr);
  fill(0);
  noStroke()

  for (var i = 0; i < freeThoughts.length; i++){
    let tht = freeThoughts[i]
    console.log(tht.lifeSpan)
    fill(0,tht.lifeSpan)
    textSize(tht.Size);
    textFont(tht.Font);
    text(tht.txt,tht.x,tht.y)
    move(tht)
  }
  }

function mousePressed(){
  console.log('mouse pressed')
  // creates a new thought
  let tht = {}
  tht.x = width/2
  tht.y = height/2
  tht.Size = random(1,200)
  tht.Font = random(fonts)
  tht.txt = getRandomThought(firstPart) + " " + getRandomThought(adjectives)
  tht.dead = random(1,10)

  frameRate(fr);
  freeThoughts.push(tht)
  tht.lifeSpan = 255
  fill(random(255), random(255), random(255))
  r = random(255);
  g = random(255);
  b = random(255);

}
function drainBrain(){

  console.log('brain')
  noStroke()
  fill(255);
  fill(r,g,b);
  frameRate(fr);
  ellipse(width/2,height/2,300,500)
  // ellipse(300,60);
}

function move(tht){

  tht.x += random(-5, 5);
  tht.y += random(-5, 5);

  tht.lifeSpan -= tht.dead
  frameRate(fr);

}

function getRandomThought(wordList){

  let string =  wordList[Math.floor(random(wordList.length))] // get random thought

  return string
   frameRate(fr);

}
```

Link: file:///Users/rafifalhassen/Documents/GitHub/Tangible-Computing-2019/students/rafif_alhassen/week6/p5%20template%207/index1.html

Sketch 11: Exploration with Text Generated Within a Shape



```
let x = 200;
let y = 200;

let fr = 30;

let thoughts = ["i am hungry", "i am stressed out", "who is texting me", "what should i have for lunch",

let firstPart = ["I am", "Who is","She is","He is","We are","They are","You are","It is"]

let secondPart = ["hungry", " late for class", " eye infection, miss"]

let fonts = ["Helvetica, Times New Roman, Comic Sans, Arial"]

let freeThoughts = []

function setup() {
  createCanvas(window.innerWidth, window.innerWidth);
  textAlign(CENTER)

  frameRate(fr);
  background(0);
  console.log(adjectives)
  lifeSpan = 0.2;
  ellipseMode(CENTER);
}
  var r = 80
  var g = 120
  var b = 255

function draw() {

  background(r,g,b);
  drainBrain()
  // fill(random(255),random(255),random(255));
  stroke(0);
  frameRate(fr);
  fill(0);
  noStroke()
  ellipse(2200,2000,2000,2000);

  for (var i = 0; i < freeThoughts.length; i++){
    let tht = freeThoughts[i]
    console.log(tht.lifeSpan)
    fill(255,tht.lifeSpan)
    textSize(tht.Size);
    textFont(tht.font);
    text(tht.txt,tht.x,tht.y)
    move(tht)
  }
  }

function mousePressed(){
  console.log('mouse pressed')
  // creates a new thought
  let tht = {}
  tht.x = width/2
  tht.y = height/2
  tht.Size = random(1,200)
  tht.font = random(fonts)
  tht.txt = getRandomThought(firstPart) + " " + getRandomThought(adjectives)
  tht.dead = random(1,10)
  // tht.lifeSpan = 2;
  frameRate(fr);
  freeThoughts.push(tht)
  tht.lifeSpan = 255
  fill(random(255), random(255), random(255))
  r = random(255);
  g = random(255);
  b = random(255);
}

function drainBrain(){
  console.log('brain')
  noStroke()
  fill(255);
  fill(r,g,b);
  frameRate(fr);
  ellipse(width/2,height/2,300,500)
  // ellipse(500,80);
}

function move(tht){

  tht.x += random(-5, 5);
  tht.y += random(-5, 5);

  tht.lifeSpan -= tht.dead
  frameRate(fr);

}
function getRandomThought(wordList){

  let string = wordList[Math.floor(random(wordList.length))] // get random thought

  return string
  frameRate(fr);
}
```

Link: file:///Users/rafifalhassen/Documents/GitHub/Tangible-Computing-2019/students/rafif_alhassen/week7/p5%20template/index1.html

Sketch 12: Exploration with Rotation of Text & Boundaries

```
let x = 200;
let y = 200;

let fr = 30;

let thoughts = ["i am hungry", "i am stressed out", "who is texting me", "what should i have for lunch",

let firstPart = ["I am", "Who is","She is","He is","We are","They are","You are"]

let secondPart = ["hungry", " late for class", " eye infection, miss"]

let thirdPart = ["feeling"]

let fourthPart = ["in Abu Dhabi", "in Dubai", "in Sharjah", "in university", "at the movies", "in the co

let fonts = ['Helvetica', 'Times New Roman', 'Comic Sans', 'Typewriter']

let freeThoughts = []

function setup() {

  createCanvas(window.innerWidth, window.innerWidth);
  textAlign(CENTER)
  frameRate(fr);
  background(0);
  console.log(adjectives)
  lifeSpan = 0.2;
  ellipseMode(CENTER);
}

  var r = 80
  var g = 120
  var b = 255

  var clickPos;

function draw() {

  background(r,g,b);
  drainBrain()
  // fill(random(255),random(255),random(255));
  stroke(0);
  frameRate(fr);
  fill(255);
  noStroke()

  ellipse(width/2, height/2, 3000, 3000);
  // ellipse(2200,2100,3000,3000);

  for (var i = 0; i < freeThoughts.length; i++){
    let tht = freeThoughts[i]
    fill(0,tht.lifeSpan)
    textSize(tht.size);
    textFont(tht.font);
    push()
    translate(tht.x,tht.y)
    rotate(tht.rotate)

        for (var s = 3; s > 1; s--){
    // let tht = freeThoughts[s]
    textSize(tht.size-s);
    fill(0,0,0,tht.lifeSpan)
    text(tht.txt,0,s*10);
  }

    // rotate(PI / 3.0);
    pop();
    move(tht)
  }
  }
```

```
function mousePressed(){

  console.log('mouse pressed')
  let d = dist(mouseX,mouseY,width/2,height/2)
  console.log('distance: ', d)
  let radius = 1500
  if (d < radius){
  let tht = {}
  tht.x = mouseX
  tht.y = mouseY
  tht.size = random(1,200)
  tht.font = random(fonts)
  tht.txt = getThought()
  tht.dead = random(1,10)

  frameRate(fr);
  freeThoughts.push(tht)
  tht.lifeSpan = 255
  tht.rotate = random(0,TWO_PI) //random rotation in radians
  tht.r = random(255);
  tht.g = random(255);
  tht.b = random(255);
  fill(random(255), random(255), random(255))
  r = random(255);
  g = random(255);
  b = random(255);
  }
}
function drainBrain(){
  console.log('brain')
  noStroke()
  fill(255);
  fill(r,g,b);
  frameRate(fr);
  ellipse(width/2,height/2,300,500)
}
function move(tht){

  tht.x += random(-5, 5);
  tht.y += random(-5, 5);
  tht.lifeSpan -= tht.dead
  frameRate(fr);
}
function getRandomThought(wordList){

  let string = wordList[Math.floor(random(wordList.length))] // get random thought
  return string
  frameRate(fr);
}
```

Link: file:///Users/rafifalhassen/Documents/GitHub/Tangible-Computing-2019/students/rafif_alhassen/week7/p5%20template%205/index1.html