

MART 340

Xijie Ren

Michael Musick

Final Project Paper

This final project is a hard work, but Michael give me many helps. My idea from break out game, I build a same game with p5.js. My first solution is how to use keyboard in my project, Michael told me I can find it in p5.js website. Then I find `keyIsDown` on p5 website, first time I only write `breaker.diection` in my code, so keyboard it can't work, when Michael help me add `breaker.speed` in my code, keyboard is working.

A screenshot of a code editor with a dark background. The code is written in a light blue font. It shows two if statements. The first if statement checks for the LEFT_ARROW key. If pressed, it sets breaker.direction to -1 and increments breaker.speed by 1. The second if statement checks for the RIGHT_ARROW key. If pressed, it sets breaker.direction to 1 and increments breaker.speed by 1. The code is as follows:

```
if ( keyIsDown(LEFT_ARROW)) {  
  breaker.direction = -1;  
  breaker.speed += 1;  
}  
if ( keyIsDown(RIGHT_ARROW)) {  
  breaker.direction = 1;  
  breaker.speed += 1;  
}
```

Second solution is `checkCollision` and brick break. Using `console.log` and `map` to make `checkCollision`, first I need to write `breaker` and `ball`'s `checkCollision`, add `brickArr` in `checkCollision` in `checkCollision`'s function. Then brick break is not hard, but I don't get how to make it. It is add `brickArr[i].make = false`. Use `false` it can make brick break.

```

Ball.prototype.checkCollision = function ( breaker, brickArr ) {
  if (this.x >= breaker.x && this.x<= breaker.x+breaker.width && this.y >= breaker.y-this.rad) {
    this.y = breaker.y-this.rad;
    this.deltaY = -abs(this.deltaY);

    // change deltaX
    var balloffset = breaker.x+breaker.width/2 - this.x;
    this.deltaX = map(balloffset, -breaker.width/2, breaker.width/2, 5, -5);

    fill(200);
    console.log(balloffset, 40, 400);
  }

  for (var i = 0; i < brickArr.length; i++) {
    if (this.x >= brickArr[i].x && this.x<= brickArr[i].x+brickArr[i].width && this.y <= brickArr[i].y+this.rad+brickArr[i].height && brickArr[i].make) {
      this.y = brickArr[i].y+5+brickArr[i].height;
      this.deltaY = -this.deltaY;
      brickArr[i].make = false;
    }
    // else if (this.x >= brickArr[i].x && this.x<= brickArr[i].x+brickArr[i].width && this.y <= brickArr[i].y+this.rad+brickArr[i].height) {
    //   this.y = brickArr[i].y+5+brickArr[i].height;
    //   brickArr[i].make = 0;
    // }
  }
};

```

When I first time write my code, I want to make all object in a function, but Michael told me it will be a bad code. So I sprite three objects to three function, then use html to contact these functions in sketch.js.

```

<html>
<head>
  <meta charset="UTF-8">
  <script language="javascript" type="text/javascript" src="libraries/p5.js"></script>
  <!-- uncomment lines below to include extra p5 libraries -->
  <!--<script language="javascript" src="libraries/p5.dom.js"></script>-->
  <script language="javascript" src="libraries/p5.play.js"></script>
  <script language="javascript" type="text/javascript" src="sketch.js"></script>
  <script language="javascript" type="text/javascript" src="brick.js"></script>
  <script language="javascript" type="text/javascript" src="brickBreaker.js"></script>
  <script language="javascript" type="text/javascript" src="Ball.js"></script>
  <!-- this line removes any default padding and style, you might only need one of these values set. -->
  <style> body {padding: 0; margin: 0;} </style>
</head>

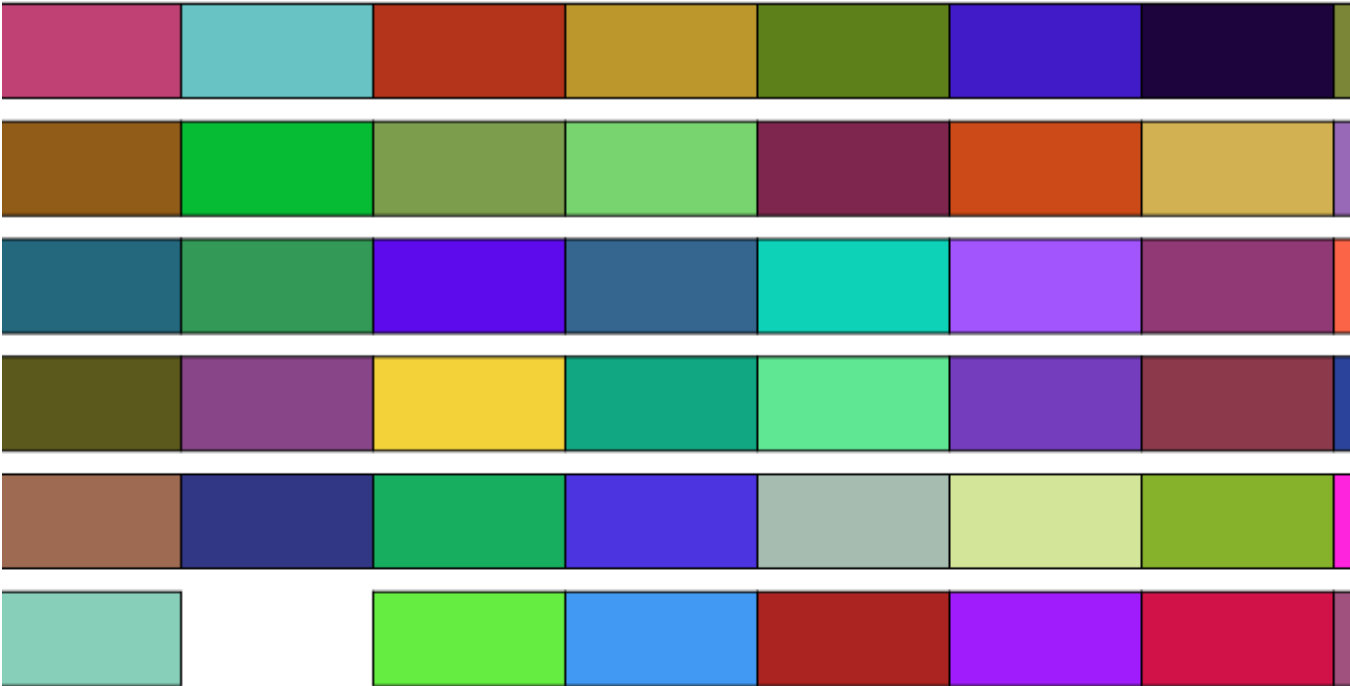
<body>
</body>
</html>

```

I use a draw function to draw my object in my project. here is my draw ball function example. In Breaker I use same function. Beside I used push and pop in my function to fix draw ball in my function.

```
Ball.prototype.drawBall = function (){  
  push();  
  fill(100);  
  translate(this.x, this.y);  
  noStroke();  
  ellipse(0,0,this.width,this.width);  
  pop();  
};
```

This my project. but I think my project can add some awards and score record.



—