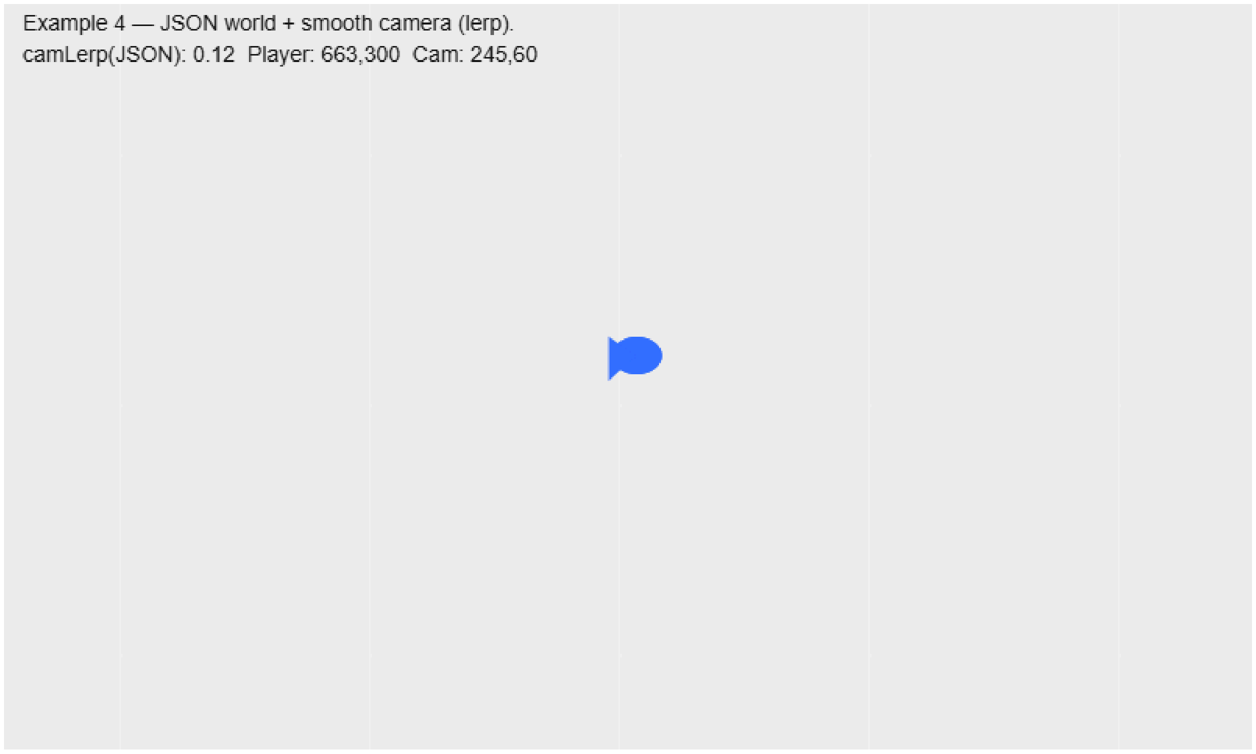
Process & Decision Documentation

Project/Assignment Decisions

One of the most major decisions that was made in the process of developing this project was to use the function createGraphics(). I chose to use this since the previous process that I was doing was not correctly fulfilling my creative vision for this assignment. If you use the “clip()” function in your code and then attempt to fill in the background with the “background” function, it will only cover half of your screen due to the way the createCanvas() function is set up. For thai reason, I made the risky decision to use createGraphics() instead, which ended up working out, but did take a fair bit of work to get it to function properly without breaking.

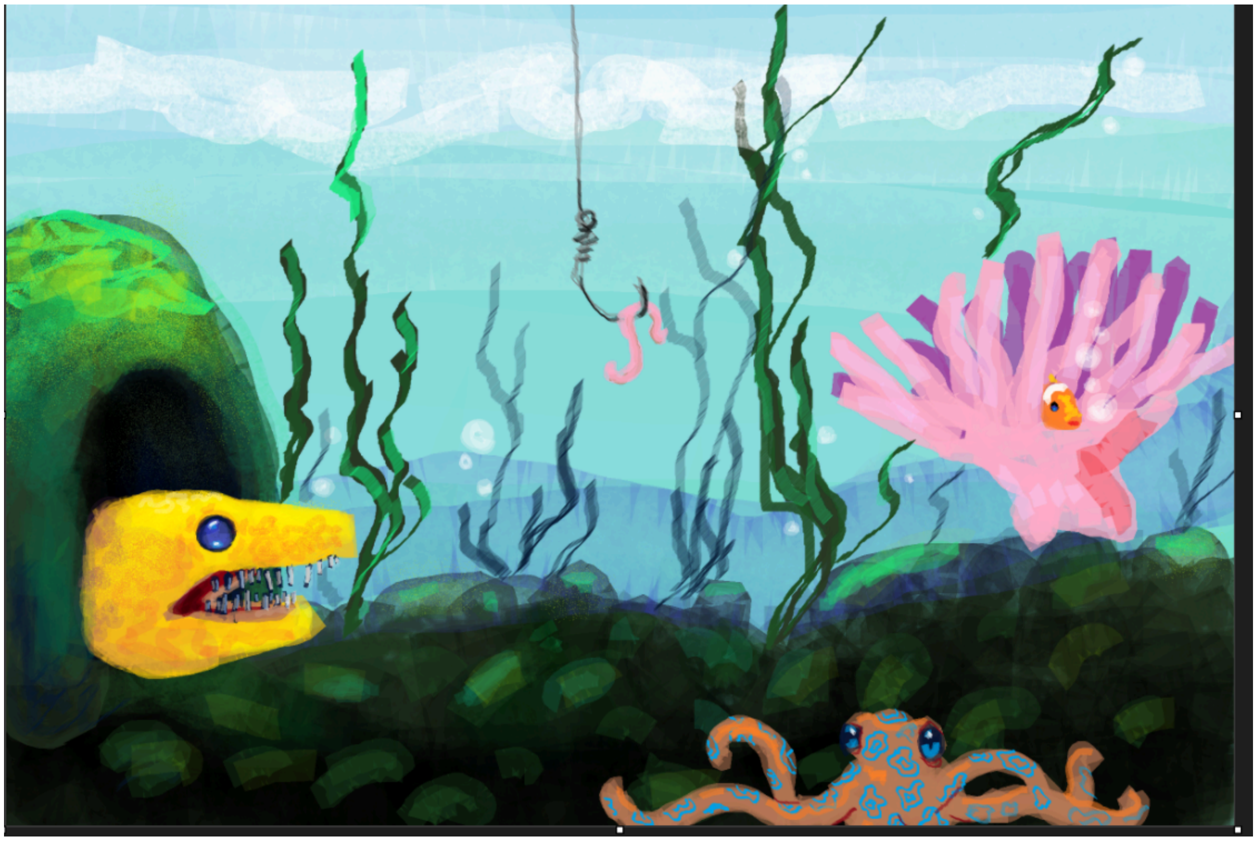
Role-Based Process Evidence

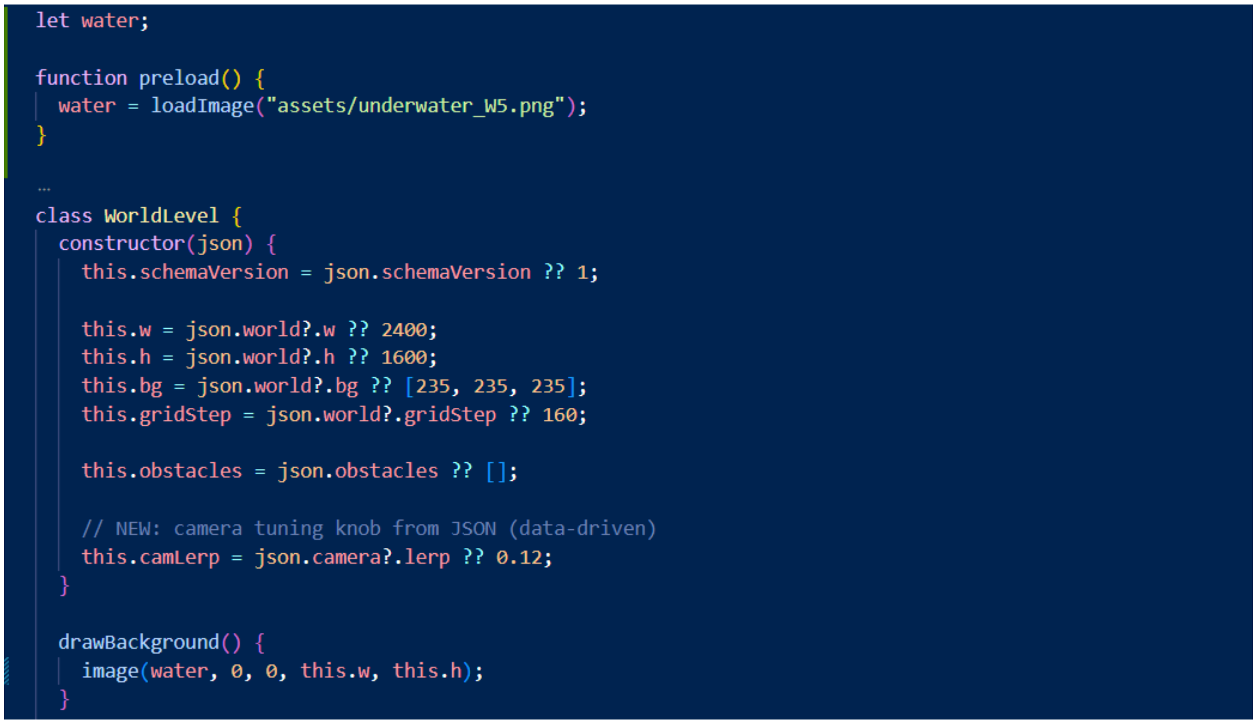


```
Example 4 — JSON world + smooth camera (lerp).
camLerp(JSON): 0.12  Player: 663,300  Cam: 245,60
```

Changing the square from the original example code to a fish

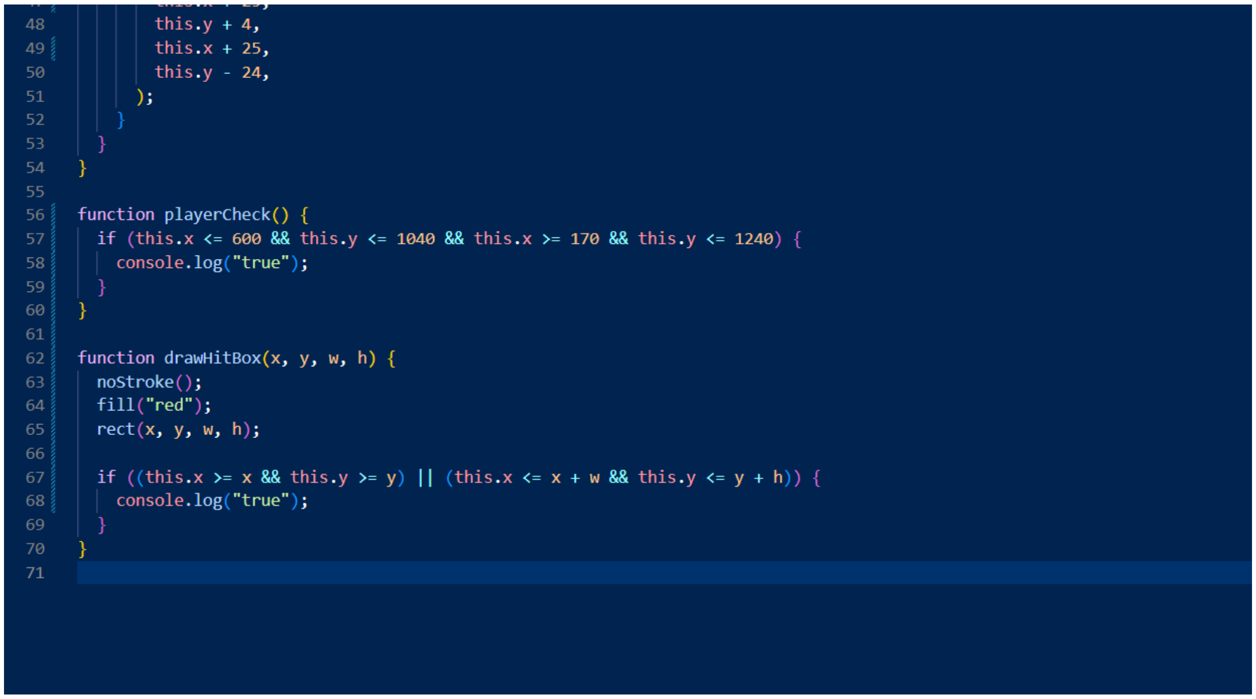
Work-in-progress image of the background, drawn in MS Paint


Finished background drawing made in MS Paint



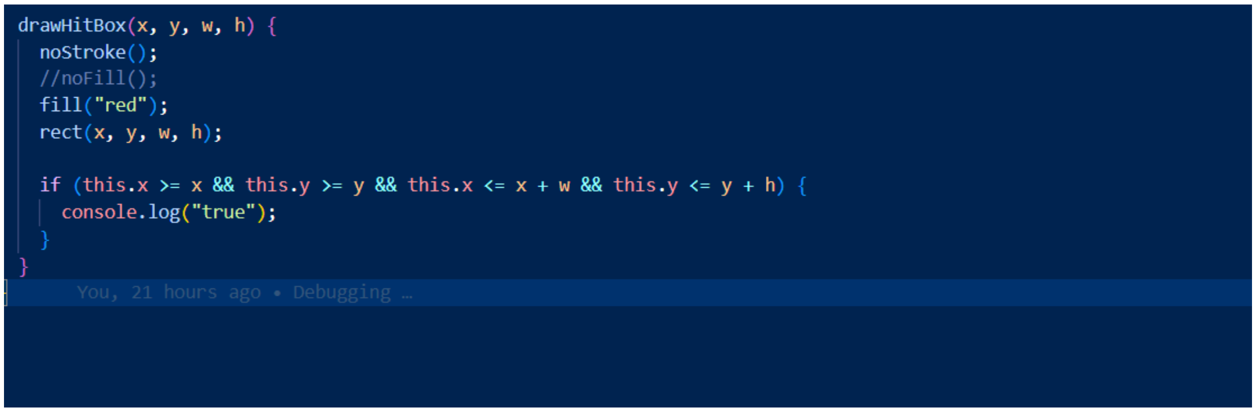
```
let water;

function preload() {
  water = loadImage("assets/underwater_W5.png");
}

...
class WorldLevel {
  constructor(json) {
    this.schemaVersion = json.schemaVersion ?? 1;

    this.w = json.world?.w ?? 2400;
    this.h = json.world?.h ?? 1600;
    this.bg = json.world?.bg ?? [235, 235, 235];
    this.gridStep = json.world?.gridStep ?? 160;

    this.obstacles = json.obstacles ?? [];

    // NEW: camera tuning knob from JSON (data-driven)
    this.camLerp = json.camera?.lerp ?? 0.12;
  }

  drawBackground() {
    image(water, 0, 0, this.w, this.h);
  }
```

Loading in the image and drawing the image in the background.



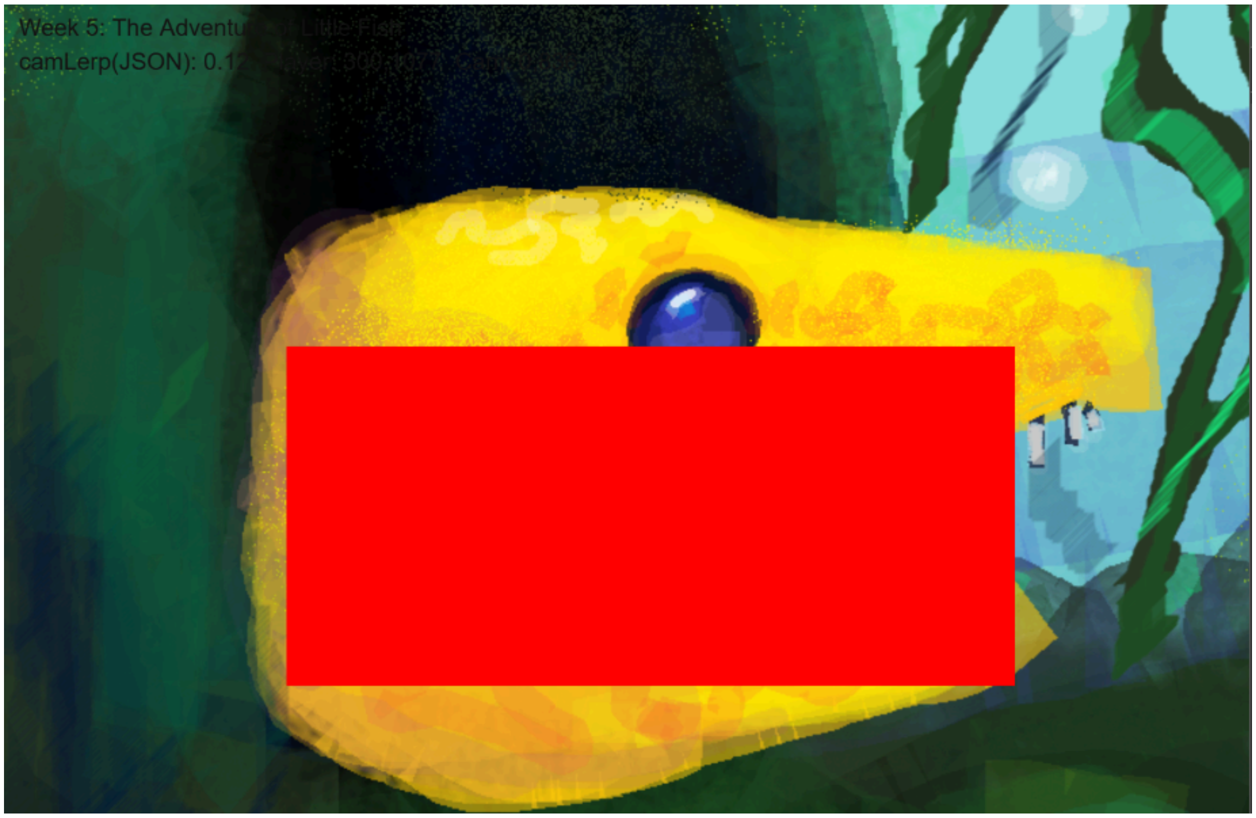
```
        this.y + 4,
        this.x + 25,
        this.y - 24,
      );
    }
  }
}

function playerCheck() {
  if (this.x <= 600 && this.y <= 1040 && this.x >= 170 && this.y <= 1240) {
    console.log("true");
  }
}

function drawHitBox(x, y, w, h) {
  noStroke();
  fill("red");
  rect(x, y, w, h);

  if ((this.x >= x && this.y >= y) || (this.x <= x + w && this.y <= y + h)) {
    console.log("true");
  }
}
```

Drawing hitboxes around each one of the 4 points of interest in the code & filling each hitbox with red for debugging purposes. Using function playerCheck to create hitbox detection



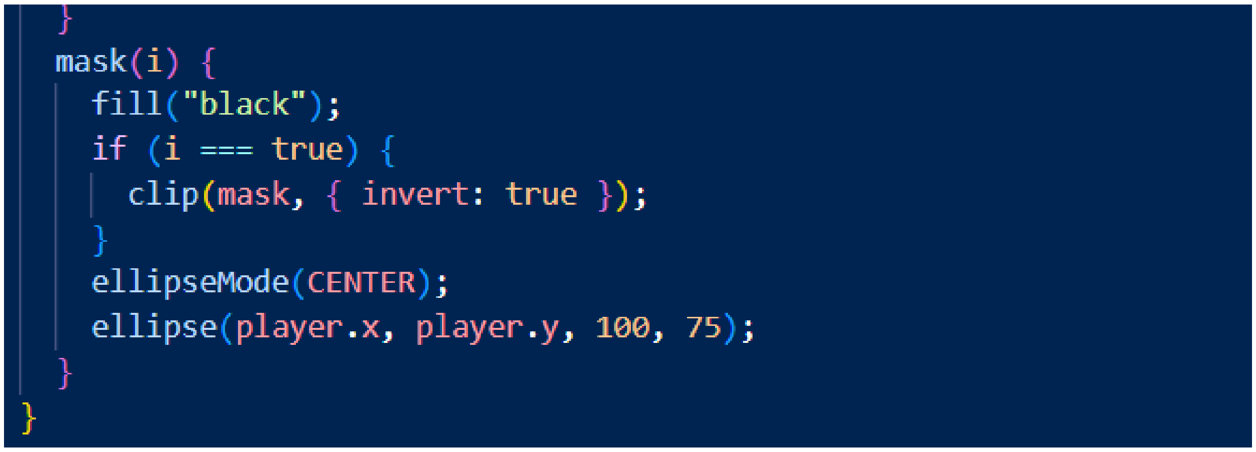
```
drawHitBox(x, y, w, h) {
  noStroke();
  //noFill();
  fill("red");
  rect(x, y, w, h);

  if (this.x >= x && this.y >= y && this.x <= x + w && this.y <= y + h) {
    console.log("true");
  }
}
      You, 21 hours ago • Debugging …
```

Changed the hitbox function so it will automatically convert the length, width, x, and y into a hitbox.


Week 5: The Adventures of Little Fish

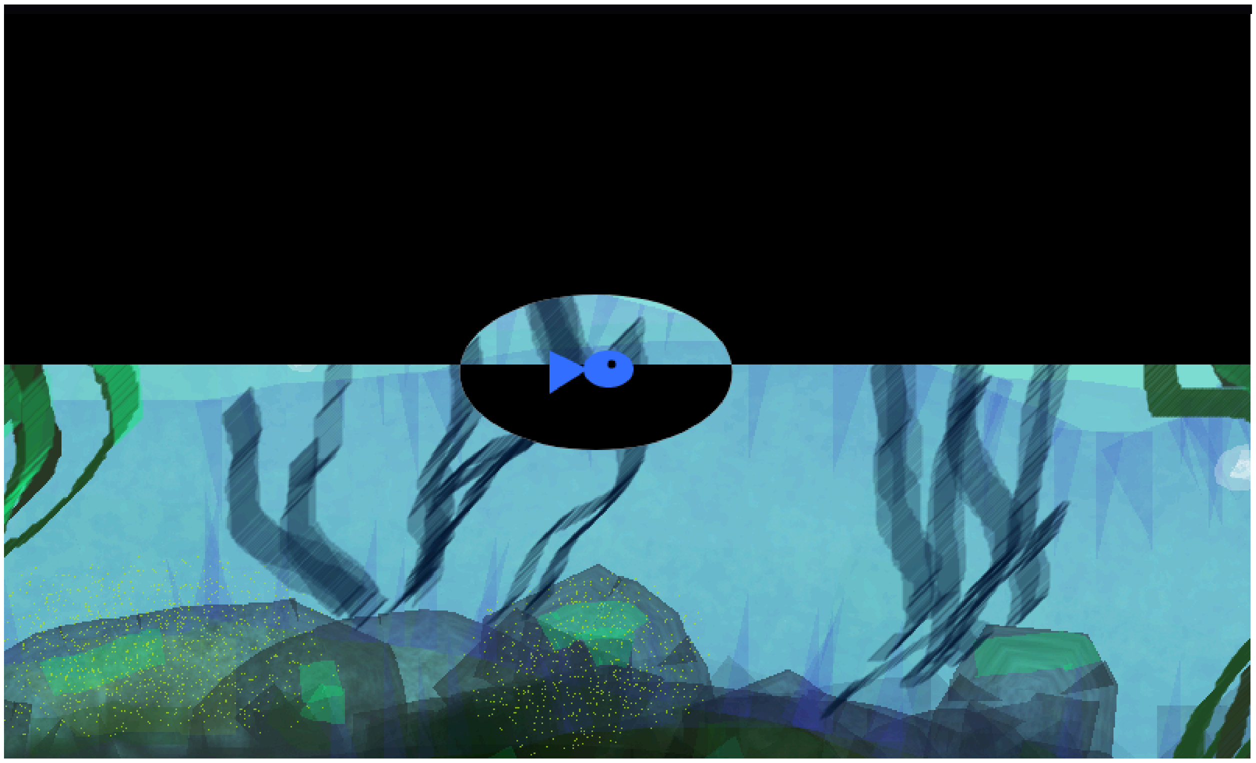
camLerp(JSON): 0.12 Player: 300,1077 Cam

Example of hitbox with red debugging mode active

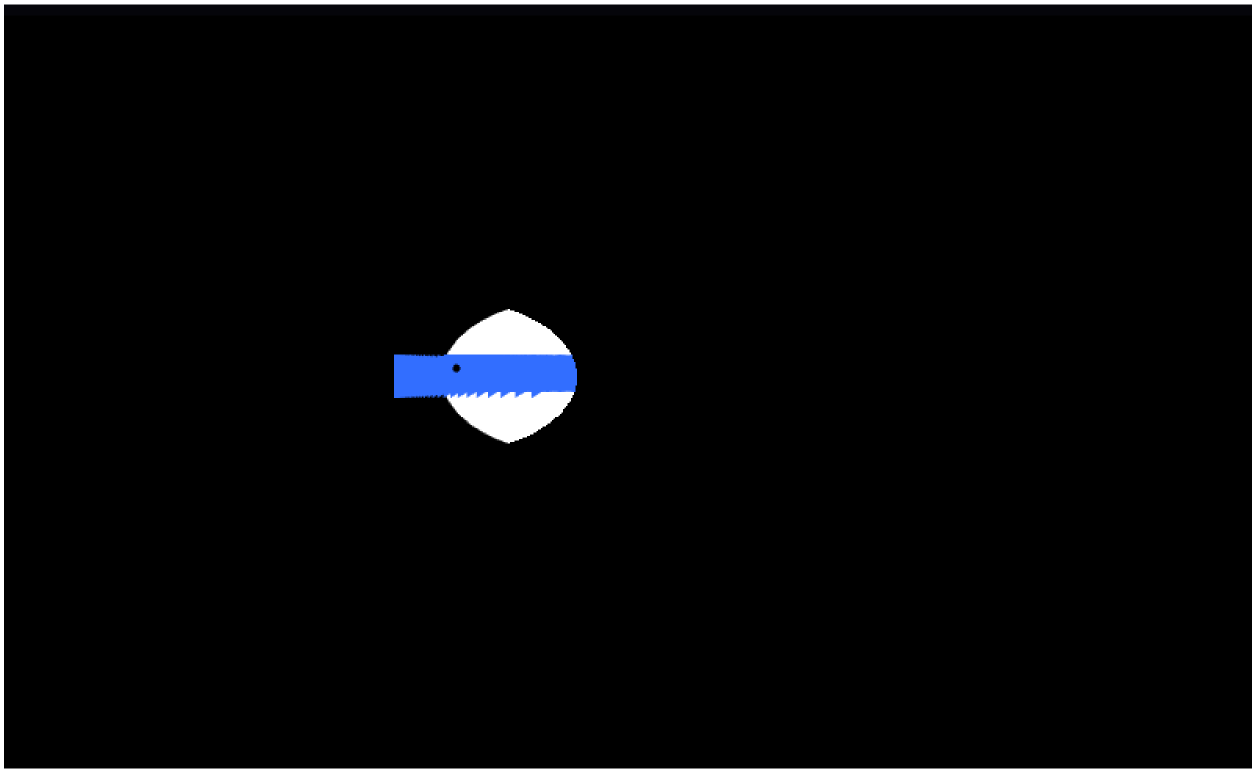


```
  }
  mask(i) {
    fill("black");
    if (i === true) {
      clip(mask, { invert: true });
    }
    ellipseMode(CENTER);
    ellipse(player.x, player.y, 100, 75);
  }
}
```

Learning how to use the clip() function in order to fill in the space with black with a circumference of vision around the player



Example of glitch found due to limitations within the createCanvas function.



Screenshot of error caused when code was uploaded to github.

Clarissa Chamberlain

Roles: Lead programmer, game tester, artist

During this assignment, I was looking to use the concepts we had learned previously in class and in our other assignments, and create an explorational experience for the player where they are tasked with looking around an underwater scene as a fish.

Tools used:

- Visual Studio Code
- GitHub Desktop
- MS Paint

GenAI Documentation

No GenAI was used for this task.