Personal Stuff\Sophomore Tech Example\DuckHunt.html

```
<!DOCTYPE html>
<html lang="en" id="html">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Cursor Position Tracker</title>
  <!--
    Everett Wilcox
    4/28/2023
    Duck Hunt Game
  -->
  <style>
    #gun {
      position: absolute;
      top: 90vh;
      left: 50vw;
      z-index: 3;
    }

    #man {
      position: absolute;
      top: 88vh;
      left: 49vw;
    }

    #cursor {
      position: fixed;
      transform: translate(-50%, -50%);
      width: -50%;
      z-index: 10;
      pointer-events: none;
    }

    body {
      position: fixed;
      width: 100vw;
      height: 100vh;
    }

    html {
      cursor: none;
      background-color: rgba(100, 148, 237, 0.226);
    }

    #grounds {
      padding: 0px;
      position: absolute;
      top: 50vh;
      z-index: 2;
      display: flex;
      align-items: row;
      transform: translateX(-20px);
      pointer-events: none;
    }


    #clouds {
      padding: 0px;
      position: absolute;
      top: -3vh;
      z-index: 2;
      display: flex;
      align-items: row;
      pointer-events: none;
    }

    .bird {
      position: absolute;
      top: 0;
      left: 0;
      width: 100px;
      height: 75px;
      z-index: 5;
      background-image: url('images/vgul.gif');
      background-size: cover;
    }
  </style>
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.4/jquery.min.js"></script>
  <script>
    $(document).ready(function () {
      const body = document.querySelector('html');
      var mouseX;
      var mouseY;
      var gunPosX = (90 / 100) * window.innerWidth;
      var gunPosY = (50 / 100) * window.innerHeight;

      // Listen for mousemove events on the body element
      body.addEventListener('mousemove', e => {
        // Get the x and y coordinates of the mouse cursor relative to the viewport
        mouseX = e.clientX;
        mouseY = e.clientY;

        // Log the coordinates to the console
        document.getElementById("cursor-position").innerHTML = (`Mouse position: ${mouseX}, ${mouseY}`);

        var susX = (mouseX - gunPosX);
        var susY = (gunPosY - mouseY);
        var hyp = Math.sqrt(Math.pow(susY, 2) + Math.pow(susX, 2));
        var thea = (susY / hyp);

        var ang = (Math.asin(thea) * 180 / Math.PI);

        $("#cursor").css({ "left": mouseX + "px", "top": mouseY + "px" });

      });


      // Get the HTML element with ID "gun"
      const gun = document.getElementById('gun');

      // Listen for mousemove events on the document
      document.addEventListener('mousemove', function (event) {

        // Get the position and center of the gun
        const gunRect = gun.getBoundingClientRect();
        const gunCenterX = gunRect.left + gunRect.width / 2;
        const gunCenterY = gunRect.top + gunRect.height / 2;

        // Get the angle between the gun and the mouse pointer
        const dx = event.clientX - gunCenterX;
        const dy = event.clientY - gunCenterY;
        const angle = Math.atan2(dy, dx) * 180 / Math.PI;

        // Set the image source based on the angle
        let imgUrl = "";
        if (angle <= -90 || angle >= 90) {
          imgUrl = "images/gun2.png";
        } else {
          imgUrl = "images/gun.png";
        }

        // Set the image source and rotate the gun to face the mouse pointer
        gun.setAttribute("src", imgUrl);
        gun.style.transform = `rotate(${angle}deg)`;
      });



      // Get the container element and all the images within it
      var container = document.getElementById("clouds");
      var images = container.getElementsByTagName("img");

      // Get the width of the first image and the total number of images
      var imageWidth = images[0].width;
      var totalImages = images.length;

      // Set the initial position, speed, and direction of the moving background
      var currentPosition = 0;
      var speed = 2;

      // Move the background in a loop using the requestAnimationFrame method
      function moveBackground() {
        currentPosition -= speed;
        container.style.left = currentPosition + "px";

        // If the background has moved one image width to the left, move the first image to the end of the container
        if (Math.abs(currentPosition) >= imageWidth) {
          currentPosition = 0;
          container.appendChild(images[0]);
          container.style.left = currentPosition + "px";
        }

        window.requestAnimationFrame(moveBackground);
      }

      // Start moving the background
      moveBackground();

      // Get the bird container element and set the bird speed
      var birdContainer = document.getElementById("bird-container");
      var birdSpeed = 2;

      // Function to create a new bird element with a random vertical position and add it to the bird container
      function createBird() {
        var bird = document.createElement("div");
        bird.classList.add("bird");
        bird.style.top = Math.floor(Math.random() * 500) + "px";
        birdContainer.appendChild(bird);

        // Call the function again after a delay to create a new bird
        setTimeout(createBird, 2000);
      }


      // This function moves the birds to the right side of the screen.
      function moveBirds() {
        // Get all the birds in the bird container
        var birds = birdContainer.querySelectorAll(".bird");

        // Loop through each bird and move them to the right by birdSpeed pixels
        for (var i = 0; i < birds.length; i++) {
          var currentPosition = parseInt(birds[i].style.left) || 0;
          var newPosition = currentPosition + birdSpeed;
          birds[i].style.left = newPosition + "px";

          // If a bird has moved past the right edge of the bird container, remove it from the DOM
          if (newPosition >= birdContainer.offsetWidth) {
            birds[i].remove();
          }

          // Check if the birds are clicked
          $(".bird").click(function() {
            $(this).remove();
          })
        }
        // Call the moveBirds function again on the next animation frame
        requestAnimationFrame(moveBirds);
      }
      // Call the createBird function to start creating birds, and then start the animation loop with moveBirds
      createBird();
      requestAnimationFrame(moveBirds);
    });

```

```
  </script>
</head>

<body ondragstart="return false;" ondrop="return false;" oncontextmenu="return false;" id="body">
  <div id="cursor-position"></div>
  <img id="cursor" src="images/cursor.png" alt="cursor">
  <div><img id="gun" src="images/gun.png" alt="gun"></div>
  <div id="bird-container"></div>
  <section id="grounds">
    <p><img src="images/hill1.png" alt="The Ground"></p>
    <p><img src="images/hill2.png" alt="The Ground"></p>
    <p><img src="images/hill1.png" alt="The Ground"></p>
    <p><img src="images/hill2.png" alt="The Ground"></p>
    <p><img src="images/hill1.png" alt="The Ground"></p>
  </section>
  <section id="clouds">
    <p><img id="cloud" src="images/emptySky.png" alt="clouds"></p>
    <p><img id="cloud" src="images/cloud1.png" alt="clouds"></p>
    <p><img id="cloud" src="images/emptySky.png" alt="clouds"></p>
    <p><img id="cloud" src="images/cloud2.png" alt="clouds"></p>
    <p><img id="cloud" src="images/emptySky.png" alt="clouds"></p>
    <p><img id="cloud" src="images/cloud3.png" alt="clouds"></p>
  </section>
</body>

</html>
```