1) 언어: HTML CSS JAVAscript

2) 개발환경: Visual Studio Code

3)

<Html>

<!DOCTYPE html>

<html lang="en">

  <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"

    />

    <link rel="stylesheet" href="style.css" />

    <title>Aim practice</title>

  </head>

  <body>

    <canvas id="canvas" width="800" height="600"></canvas>

    <script src="main.js"></script>

  </body>

</html>

<css>

#canvas {

  background-color: black;

  padding: 0;

  margin: auto;

  display: block;

  width: 800px;

  height: 600px;

  position: absolute;

  top: 0;

  bottom: 0;

  left: 0;

  right: 0;

}

JS

Setup() = 이미지 설정

function setup() {

  canvas = document.getElementById("canvas");

  ctx = canvas.getContext("2d");

  width = 800;

  height = 600;

  mainmenu = true;

  backgroundImg = new Image();

  backgroundImg.src = "res/background.png";

  targetImg = new Image();

  targetImg.src = "res/target.png";

  menutextImg = new Image();

  menutextImg.src = "res/menutext.png";

  startbutton\_darkImg = new Image();

  startbutton\_darkImg.src = "res/startbutton\_dark.png";

  startbutton\_lightImg = new Image();

  startbutton\_lightImg.src = "res/startbutton\_light.png";

  gameovertextImg = new Image();

  gameovertextImg.src = "res/gameovertext.png";

  retrybutton\_darkImg = new Image();

  retrybutton\_darkImg.src = "res/retrybutton\_dark.png";

  retrybutton\_lightImg = new Image();

  retrybutton\_lightImg.src = "res/retrybutton\_light.png";

}

Setupgame() = 게임 초기화 설정 (마우스 좌표, 점수, 시작시간, 남은시간)

mousePos: 마우스 좌표

mouse1: 마우스 클릭 여부

score: 점수

starttime: 시작시간

timeleft: 남은시간

gameover: 게임오버

function setupGame() {

  mousePos = { x: 0, y: 0 };

  mouse1 = false;

  score = 0;

  startTime = Math.round(new Date().getTime() / 1000);

  timeleft = 30;

  gameover = false;

  target.setupTarget();

}

Target.setuptarget(): 타겟 이미지의 좌표를 정가운데로 설정

This.radius: 타겟의 반지름

target = {

  setupTarget: function () {

    this.radius = 30;

    this.x = width / 2;

    this.y = height / 2;

  },

Target.draw: 타겟 이미지 생성

draw: function () {

    ctx.drawImage(

      targetImg,

      this.x - this.radius,

      this.y - this.radius

    );

Target.Update: 타겟 이미지 좌표범위 안을 클릭하면 hit, 클릭 못하면 miss

update: function () {

    x = mousePos.x - this.x;

    y = mousePos.y - this.y;

    if (x \*\* 2 + y \*\* 2 <= this.radius \*\* 2) {

      //hit

      if (mouse1) {

        this.hit();

      }

    } else {

      //miss

    }

  },

Target.hit: 타겟을 클릭하면 다음 좌표를 랜덤으로 설정, 점수 1점 획득

hit: function () {

    this.x = Math.floor(Math.random() \* 700 + this.radius);

    this.y = Math.floor(Math.random() \* 400 + 100);

    score += 1;

  },

Mainmenuscreen: 메인메뉴화면

function mainmenuscreen() {

  ctx.drawImage(backgroundImg, 0, 0);

  ctx.drawImage(menutextImg, 300 - 107, 150);

  if (

    mousePos.x > width / 2 - 100 &&

    mousePos.x < width / 2 + 100 &&

    mousePos.y > height / 2 + 50 &&

    mousePos.y < height / 2 + 100

  ) {

    ctx.drawImage(

      startbutton\_lightImg,

      width / 2 - 100,

      height / 2 + 50

    );

    console.log;

    if (mouse1) {

      mainmenu = false;

      setupGame();

    }

  } else {

    ctx.drawImage(

      startbutton\_darkImg,

      width / 2 - 100,

      height / 2 + 50

    );

  }

}

Render: 게임 화면

function render() {

  ctx.drawImage(backgroundImg, 0, 0);

  target.draw();

  ctx.fillStyle = "black";

  ctx.font = "20px Arial";

  ctx.fillText("score: " + score, 10, 30);

  ctx.fillText("time: " + timeleft, 600, 30);

}

Gamevoerscreen(): 게임오버화면

function gameoverscreen() {

  ctx.drawImage(backgroundImg, 0, 0);

  ctx.drawImage(gameovertextImg, 300 - 114, 150);

  ctx.textAlign = "center";

  ctx.fillStyle = "white";

  ctx.font = "40px Arial";

  ctx.fillText("score: " + score, width / 2, height / 2 + 10);

  if (

    mousePos.x > width / 2 - 100 &&

    mousePos.x < width / 2 + 100 &&

    mousePos.y > height / 2 + 50 &&

    mousePos.y < height / 2 + 100

  ) {

    ctx.drawImage(

      retrybutton\_lightImg,

      width / 2 - 100,

      height / 2 + 50

    );

    if (mouse1) setupGame();

  } else {

    ctx.drawImage(

      retrybutton\_darkImg,

      width / 2 - 100,

      height / 2 + 50

    );

  }

}

Update: mainmenu가 true일 때, 메인메뉴 출력, false라면 게임화면 출력

Timeleft<0이면, 게임오버화면 출력

function update() {

  if (mainmenu) {

    mainmenuscreen();

  } else {

    if (timeleft >= 0) {

      // gameplay

      target.update();

      timeleft =

        startTime - Math.round(new Date().getTime() / 1000) + 30;

      render();

    } else {

      gameoverscreen();

    }

  }

  mouse1 = false;

}

getMousepos: 실시간 마우스 좌표 반환

function getMousePos(canvas, evt) {

  var rect = canvas.getBoundingClientRect();

  return {

    x: evt.clientX - rect.left,

    y: evt.clientY - rect.top,

  };

}

Addeventlistner: 마우스 움직일 때마다 좌표 설정, 마우스 클릭했을 때 mouse1값 true

window.addEventListener(

  "mousemove",

  function (evt) {

    mousePos = getMousePos(canvas, evt);

  },

  false

);

window.addEventListener("mousedown", function (evt) {

  mouse1 = true;

});

Setinterval: 1초에 240번 update

setInterval(function () {

  update();

}, 1000 / 240);