Gabby Melamed

Paddle Game Abstraction and All Code Artifact

ball.js:

```
class Ball {
  //constructs values to pass into the balls
constructor(x, y, dx, dy){
   this.loc = createVector(x, y);
   this.vel = createVector(dx, dy);
   this.acc = createVector(0,0);
   this.clr = color(random(255), random(255), random(255));
  }

//runs all of the ball functions
  run(){
    this.render();
    this.checkedges();
    this.update();
  }

//makes each ball bounce when it reaches any edge of the canvas
//changes x velocity to the opposite sign when it hits the left or right
//changes y velocity to the opposite sign when it hits the top or bottom
  checkedges(){
    if(this.loc.x < 0){
      this.vel.x = -this.vel.x;
    }
    if(this.loc.x > width){
      this.vel.x = -this.vel.x;
    }
    if(this.loc.y < 0){
      this.vel.y = -this.vel.y;
    }
    if(this.loc.y > height){
      this.vel.y = -this.vel.y;
    }
  }

//deletes a ball out of the array everytime it hits the paddle coming down only
//increases the score by 1 when the ball hits the paddle
update(){

//makes the velocity have a limit so the balls don't get too fast
    this.vel.limit(7);
    this.vel.add(this.acc);
    this.loc.add(this.vel);

    //if the balls are colliding the paddle at the top, they will disappear
    //if they collide at the bottom, the balls should disappear and more balls will appear
    for(var i = balls.length - 1; i >= 0; i--) {
      if(balls[i].isColliding() && this.vel.y > 0) {
          balls.splice(i, 1);
          score = score + 1;
        }else if(balls[i].isColliding() && this.vel.y < 0) {
          loadBalls(20);
          lives = lives - 1;
        }
}
//keeps velocity the same for the easiest level
```

```
   if(gameLevel === 'easy') {
     this.vel.x = this.vel.x * 1;
     this.vel.y = this.vel.y * 1;
   }

   //makes the balls faster than easy level

   if(gameLevel === 'medium') {
     this.vel.x = this.vel.x * 1.1;
     this.vel.y = this.vel.y * 1.1;
   }

   //makes the balls faster than medium level

   if(gameLevel === 'hard') {
     this.vel.x = this.vel.x * 1.2;
     this.vel.y = this.vel.y * 1.2;
   }
  }

//makes the ball know when it hits the paddle
   isColliding() {
    if(this.loc.x + 13 > paddle.loc.x &&
         this.loc.x - 13 < paddle.loc.x + paddle.w &&
         this.loc.y + 13 > paddle.loc.y &&
         this.loc.y - 13 < paddle.loc.y + paddle.h)
         {
           return true;
         } else {
           return false;
```

```
       }
      }

//creates the ball
  render(){
    fill(this.clr);
    ellipse (this.loc.x, this.loc.y, 26, 26);
  }
}//  +++++++++++++++++++++++++++++++++++  End Ball Class
```

button.js:

```
class Button {


  //constructs values to pass into the buttons
  constructor(x, y, msg){
    this.loc = createVector(x, y);
    this.msg = msg;
    this.clr = color(random(255), random(255), random(255));
  }

  //displays the buttons on the screen
  run() {
    this.render();
  }

  //creates the button with the words in it
  render() {
    fill(this.clr);
    rect(this.loc.x, this.loc.y, 100, 100)
    fill(255, 0, 0);
    textSize(20);
    text(this.msg, this.loc.x, this.loc.y);
  }
}
```

Index.html:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Paddle Game</title>
    <script src="libraries/p5.js" type="text/javascript"></script>
    <script src="libraries/p5.dom.js" type="text/javascript"></script>
    <script src="libraries/p5.sound.js" type="text/javascript"></script>
    <script src="sketch.js" type="text/javascript"></script>
    <script src="ball.js" type="text/javascript"></script>
    <script src="paddle.js" type="text/javascript"></script>
    <script src="button.js" type="text/javascript"></script>
    <style> body {padding: 0; margin: 0;} canvas {vertical-align: top;} </style>
  </head>


  <body>
  </body>
</html>
```

paddle.js:

```
class Paddle {

  //passes in the values of the paddles
  constructor(x, y, w, h) {
    this.loc = createVector(x, y);
    this.w = w;
    this.h = h;
    //this.loc2 = createVector(w, h);
    this.clr = color(random(255), random(255), random(255));
  }

//shows the paddle on the screen and allows movement
  run() {
    this.render();
    this.update();
  }

//creates the paddle
  render() {
    fill(this.clr);
    rect(this.loc.x, this.loc.y, this.w, this.h);
  }

//allows the mouse to move the x location of the paddle
  update() {
    var paddleMouseLoc = createVector(mouseX, 700);
    this.loc = p5.Vector.lerp(this.loc, paddleMouseLoc, 0.09);
  }
}
```

sketch.js:

```
var paddle;
var balls = [];
var gameState = 1;
var score = 0;
var gameLevel;
var lives = 5;
var buttonEasy;
var buttonMedium;
var buttonHard;
var buttonPlayAgain;
var buttonEndGame;
var buttonInstructions;
var buttonBack;
var numBallsEasy = 5;
var numBallsMedium = 10;
var numBallsHard = 15;

//create canvas and background
function setup() {
  var cnv = createCanvas(800, 800);
  cnv.position((windowWidth-width)/2, 30);
  background(5, 5, 5);
  fill(200, 30, 150);
}

//The draw function is called automatically @ 30 fps
//associates a game screen with a gameState number
//calls a game screen when the gameState = the associated number
function draw() {
  background(5, 5, 5, 50);
  if (gameState === 1){
    startGame();
  } else if(gameState === 2){
    gameMode();
  } else if(gameState === 3){
    endGame();
  }else if(gameState === 4){
    instructions();
  }
}

//start screen of game
//displays buttons with different levels the player can choose from
function startGame() {
  lives = 5;
  clear();
  background(5, 5, 5);
  textSize(90);
  fill(255, 255, 255);
  text('The Paddle Game', 30, 200);

  //displays the buttons that have .run in makeButtons
  makeButtons();

  //sets game level to the level of the button the player pressed on
  //changes the gameState to gameState = 2 which calls the gameMode function
  if(mouseIsPressed) {
    if(mouseX > 120 &&
      mouseX < 220 &&
      mouseY > 600 &&
      mouseY < 700) {
        gameLevel = 'easy';
        gameState = 2;
      }else if(mouseX > 270 &&
      mouseX < 370 &&
      mouseY > 600 &&
      mouseY < 700) {
        gameLevel = 'medium';
        gameState = 2;
      }else if(mouseX > 420 &&
      mouseX < 520 &&
      mouseY > 600 &&
      mouseY < 700) {
        gameLevel = 'hard';
        gameState = 2;
      }

the//when the player presses on the instructions button, the game screen changes to
instructions screen
      else if(mouseX > 570 &&
      mouseX < 670 &&
      mouseY > 600 &&
      mouseY < 700) {
        gameState = 4;
      }
    }

    //loads a different amount of balls depending on the game level that is selected
    if(gameLevel === 'easy'){
      loadBalls(numBallsEasy);
    }
    if(gameLevel==='medium'){
      loadBalls(numBallsMedium);
    }
    if(gameLevel==='hard'){
      loadBalls(numBallsHard);
    }
}

//instruction screen
//displays the game instructions

function instructions() {
  clear();
  background(5, 5, 5);
  fill(255,192,203);
  textSize(30);
  text('INSTRUCTIONS', 290, 50);
  textSize(18);
  text('Welcome to The Paddle Game! Your score and lives are displayed in the top left
corner.',
10, 100);
  text('The goal of this game is to collect all of the balls on the screen.', 10, 130);
  text('To collect the balls, move the paddle so that that ball(s) land on the paddle.', 10,
160);
  text('When the ball(s) hit the top of the paddle, they will disappear.', 10, 190);
  text('But, do not let the balls hit the bottom of the paddle.', 10, 220);
  text('Randomely, if they do, the remaining balls on the screen will disappear and more
balls
will appear.', 10, 250);
  text('You will win the game when you have collected all of the balls on the screen', 10,
280);
```

```
  text('Everytime new balls appear, you will lose a life.', 10, 310);
  text('You will start with 5 lives and once your lives = 0, you lose the game.', 10, 340);
  text('After you have won or lost the game, you will have the option to quit or play
again.',
10,
370);
  text('Good luck!', 50, 400);

  //displays the back button
  //when the back button is pressed, the gameState = 1 again the screen changes back
to the
start screen
  buttonBack.run();
  if(mouseIsPressed &&
    mouseX > 570 &&
    mouseX < 670 &&
    mouseY > 450 &&
    mouseY < 550) {
      gameState = 1;
    }
}

//creates all of the buttons
//only runs the buttons that appear in startGame
function makeButtons() {
  buttonEasy = new Button(120, 600, 'EASY');
  buttonMedium = new Button(270, 600, 'MEDIUM');
  buttonHard = new Button(420, 600, 'HARD');
  buttonPlayAgain = new Button(500, 110, 'PLAY AGAIN?');
  buttonEndGame = new Button(200, 110, 'QUIT?');
  buttonInstructions = new Button(570, 600, 'INSTRUCTIONS');
  buttonBack = new Button(570, 450, 'BACK');
  buttonEasy.run();
  buttonMedium.run();
  buttonHard.run();
  buttonInstructions.run();

}

function gameMode(){

    //displays a score in the game screen
    fill(255, 0, 0);
    textSize(35);
    text('score:' + score, 30, 30);
    text('lives:' + lives, 30, 70);

    //drops the balls down from the top and displays the paddle
    runBalls();

    //when all of the lives have been used, screen switches to gameState = 3

    if(lives === 0) {
      gameState = 3;
    }

3   //when all of the balls have been collected for a level, screen switches to gameState =
    if (score === balls.length + score) {
      gameState = 3;
    }
  }
```

```
//the end screen of the game
function endGame() {
  clear();
  background(5, 5, 5);

  //displays buttonPlayAgain and buttonEndGame
  buttonPlayAgain.run();
  buttonEndGame.run();

  //if player collected all of the balls, 'YOU WIN' is displayed on the screen
  if(score === balls.length + score)  {
   textSize(100);
   fill(255, 10, 10);
   text('YOU WIN!!!', 130, 500);
  }

  //if player runs out of lives, 'YOU LOST' is displayed on the screen
  if(lives === 0) {
   fill(255, 0, 0);
   textSize(100);
   text('YOU LOST!', 120, 400);
  }

   //if player presses buttonPlayAgain, gameState is reset to one and the start screen
will
appear
   if(mouseIsPressed &&
    mouseX > 500 &&
    mouseX < 600 &&
        mouseY > 110 &&
        mouseY < 210) {
        score = 0;
        gameState = 1;
        balls = [ ];
        gameLevel = '';
        }

        //if player presses buttonEndGame, everything on the screen is erased and the
screen
turns white
      if(mouseIsPressed &&
        mouseX > 200 &&
        mouseX < 300 &&
        mouseY > 110 &&
        mouseY < 210) {
          remove();
        }
}

//creates the paddle and the array of balls
function loadBalls(x) {
  paddle = new Paddle(250, 700, 300, 25);
  for(var i = 0; i < x; i++){
    balls[i] = new Ball(random(0, 800), random(0, 100), random(1, 5), random(1, 5));
  }
}

//displays the paddle and the array of balls
function runBalls(x) {
  paddle.run();
  for(var i = 0; i < balls.length; i++){
    balls[i].run();
  }
}
```