Comments For Assignment 5

Well done Lester! Good commenting and nice work with using in-built variables like width and height.

So like you pointed out in your comments in google classroom, the paddles do not move, the scores are not incremented, the second ball also does not move at all, and lastly you still have some global variables at the start of your which can be thinned out from your classes.

General comments.

* Normally, it's good practice to have each class in a separate file (or tab) named exactly as the class. Eg. Your Ball, Paddle and Game classes will be in their own tabs(or files) named Ball, Paddle and Game respectively. Although separating classes into their own files is not obligatory, you have a better chance at spotting errors and understanding how these classes integrate into your final program.
* Also, you don’t need to have global variables for the same instance variables of your class objects. Instance variables are tied to your class objects. So each time you create a new object (using a constructor and parameters), the object inherently knows its instance variables. Eg. Your paddle class has instance variables x,y, w, h for the paddle positions, width and height. So once you call the Paddle class constructor with these lines:

leftPaddle= new Paddle(width-width, height-height, width/50, height/3); rightPaddle= new Paddle(width-(width/50), height-((height/2)+(height/6)), width/50, height/3);

LeftPaddle, rightPaddle are newly created objects and inherently have their respective

x,y,w,h instance variables assigned the parameters you pass into their individual

constructors. Thus you don’t need a global variable, LeftPaddleX, or RightPaddleX,

Because you have access to these through your objects. So if i want my leftPaddle’s x,y

Co-ordinates, I simply use the the name of the paddle object and the specific instance variable : Eg. leftPaddle**.**x, leftPaddle**.**y Similarly rightPaddle**.**w, rightPaddle**.**h for the width and height of the rightPaddle. NB: The dot is necessary. Syntax rule: objectReference**.**variable

* Similar the dot notation, allows you to call **methods** on your objects which you do very well. Eg.leftPaddle.display(); But this can also be applied to method definitions of a class containing Object parameters from a different class. Eg. void checkWall(Game game) is a method in your Ball Class. However since it has a Game object Parameter, (Game game), it can access methods and instance variables of your Game class using the dot-notation.So within your definition of checkWall method in your ball class, you can call game.incrementLeftScore(); or some other method using the game parameter.

Specifics.

* With these ideas, you have flexibility of adding instance variables to your classes rather than global. Your global variables should only be from your class constructor.
* Since you have to include two balls, you want these to have separate x and y velocities so you can clearly see them, Else you could have 2 balls but they move directly on top of each other so you see them only as one. You can use the random function to assign a random velocity to each ball’s velocity instance variable in your ball class. You can look online to see ways to do this. Try out this link if you get stuck. <http://processingjs.org/reference/random_/>

Here are my edits and suggestions. You can make changes and re-submit.

<https://docs.google.com/document/d/1DyQUaIvVw8_x7qnT9KxVrRib7_LU0HEckh0_m5gPWoo/edit?usp=sharing>