# Lab Week 5A. Using a Class

### Topic so far

* Variables, local and global
* Loops
* Procedures (void) and parameters
* Functions – return a value
* Animation & Events – setup, draw, keyPressed etc

## Learning Objectives

* Understanding and using a Class
* Creating objects (instances of a Class)
* Manipulating object behaviour using its methods
* Using a constructor

## Resources

* Lecture Notes
* Extra tutorials – see moodle
* Example Ball class – download from moodle and take a look. Note the members, constructor, methods and how it’s used from the setup and draw events
* Processing website – reference
* External Objects tutorials
  + <https://docs.oracle.com/javase/tutorial/java/concepts/index.html>
  + <https://processing.org/tutorials/objects/>

Save your code after each exercise

***Using a class***. In the lecture we looked at creating two instances of a class and using its methods to manipulate the instances. We’ll start with a simple class and add methods to improve our program over the next few exercises.

**Pair programming exercise**. The code below is a for a class called Ball. Create a sketch add a blank setup() and draw() event. Add the class below in a new tab in your sketch. Create 3 or 4 ball objects such that they start in the corners of the canvas e.g. (0,0), (width,height) etc and bounce around the screen.

class Ball

{

int x,y;

int speedX=2, speedY=2;

Ball(int x, int y)

{

this.x = x;

this.y =y;

}

void display()

{

ellipse(x,y,10,10);

}

void move()

{

this.x = this.x+this.speedX;

this.y = this.y+this.speedY;

if (this.x<0 || this.x>width)

{ this.speedX=-this.speedX;}

if (this.y<0 || this.y>height)

{ this.speedY=-this.speedY;}

}

void update()

{

this.move();

this.display();

}

}

**Ex1.** Similar to the motorbike race we covered in the webinar, you should write a program to allow three cars to race from the **bottom** of the screen to the **top**. This will need a class Car and images can be found here.

Your Car class should have

* suitable member variables – what do we need know to draw the car in its current position and move it up the screen?
* Constructor – allow position and an image file to be passed in
* Render or display method – draw the car in its current position
* Move method – change the position variable values – use the random function to provide a random value [1..3], random function, e.g.
  + x= x + (int) random(1,3); //random value between 1 & 3.
* Update method – call the move and render methods

Add in game modes to control the draw event and stop them moving once they have finished. Use the text command to state which car won the race.

//constants and variables : At top of code

final int RACING=0;

final int FINISH=1;

int gameMode = RACING;

**Ex7.**  Amend your code so that the number of wins for each bike is displayed on the screen. Hint, the code below would display “blue:0” at position 10,10.

int winYellow = 0;

text(“YellowCar:”+winYellow,10,10);

**Extension exercises**, modify your code to allow one of the cars to be controlled by the user – moving that particular car by pressing a specific key (or a mouse click). One key press should move the bike a set value (try 15 pixels). You can also add a start screen.

Look at and experiment with the example at <https://processing.org/examples/objects.html> add another instance of the MRect class.