```
Ball balls = new Ball ();
Line lines = new Line();
Surface surfaces = new Surface();


import processing.serial.*;
float[] datalist;


Serial myPort;
String incomingData = null;
float rawSensorData1 = 0;
float rawSensorData2 = 0;
float rawSensorData3 = 0;

int x1 = 200;
int y1 = 200;
int x2 = 300;
int y2 = 300;
int x3 = 300;
int y3 = 300;


float angleRotate = 0;

void setup() {
  size(1200, 1200);
  String portName = Serial.list()[5];
  myPort = new Serial(this, portName, 9600);
  myPort.bufferUntil('\n');


  background (0,0,0);
  textSize(15);
  fill(255,255,255);
  text ("Sitexx“, 790, 1025);
  textAlign (RIGHT, BOTTOM);

  float angle1 = radians(90);
  translate(130, 240);
  rotate(angle1);
  text("Site 1", 0, 0);
  circle(20,-3,6.3);
  circle(13,-8,6.3);
  circle(20,-12,6.3);

  float angle2 = radians(0);
  translate(350, 0);
  rotate(angle2);
  text("Site 2  //", 0, 0);

  float angle3 = radians(0);
  translate(350, 0);
  rotate(angle3);
  text("Site 3", 0, 0);
  rect(17,-13,10,10);
}


void draw() {
  lines.update();
  balls.update();
  surfaces.update();
  rec();
}


void serialEvent(Serial myPort) {
  incomingData = myPort.readString();
  datalist =float(split(trim(incomingData), ','));
  println (datalist);

  rawSensorData1=int(datalist[0]);
  rawSensorData2 =int(datalist[1]);
  rawSensorData3=int(datalist[2]);

  myPort.clear();
}

void keyPressed() {
  if (key == 's' || key == 'S')
    saveFrame("output/image####.jpg");}


===================================
class Ball {
  Ball() {
  }

  // functions
  void update() {
    if (x1>=width-210) {
      x1= 185 + int(datalist[0]/4);
      y1= y1 + 30;
    } else {
      x1= x1+int(rawSensorData1/26)+9;
    }

    if (y1>=height-210) {
      y1=200;

      background (0);
      fill(255);
      textSize(15);
      text ("b, 09.02.2022", 1023, 1029);
      textAlign (RIGHT, BOTTOM);
      float angle1 = radians(90);
      translate(130, 240);
      rotate(angle1);
      text("Site 1", 0, 0);

        circle(20,-3,5);
        circle(13,-8,5);
        circle(20,-12,5);

      float angle2 = radians(0);
      translate(350, 0);
      rotate(angle2);
      text("Site 2  //", 0, 0);

      float angle3 = radians(0);
      translate(350, 0);
      rotate(angle3);
      text("Site 3", 0, 0);
      rect(17, -13, 10, 10);
    }

    stroke(255, rawSensorData1*1.7);
    fill(255, rawSensorData1*1.7);
    circle(x1, y1, rawSensorData1/100);
  }
}
=================================
class Line {

  Line(){

  }


void update(){
  if (x2>=width-210){
    x2= 185 + int(rawSensorData2/8);
    y2= y2 + 25;}
  else{
    x2= x2+int(rawSensorData2/15);}

  if (y2>=height-210){
    y2=200;



    background (0);
    fill(255);
    textSize(15);
    text (", 09.02.2022",1023,1029);
    textAlign (RIGHT,BOTTOM);
    float angle1 = radians(90);
    translate(130, 240);
    rotate(angle1);
    text("Site 1", 0, 0);
      circle(20,-3,5);
  circle(13,-8,5);
  circle(20,-12,5);

    float angle2 = radians(0);
    translate(350, 0);
    rotate(angle2);
    text("Site 2  //", 0, 0);

    float angle3 = radians(0);
    translate(350, 0);
    rotate(angle3);
    text("Site 3", 0, 0);
    rect(17,-13,10,10);
  }
    //stroke(0, rawSensorData2*1.7);
   // fill(0, rawSensorData2*1.5);
    circle(x2, y2, rawSensorData2/200);
}
}
==================================
class Surface {

  Surface(){

  }


void update(){
  if (x3>=width-210){
    x3= 185 + int(rawSensorData3/8);
    y3= y3 + 35;}

  else{
    x3= x3+int(rawSensorData3/8)+5;}

  if (y3>=height-210){
    y3=200;



    background (0);
    fill(255);
    textSize(15);
    text ("b, 09.02.2022",1023,1029);
    textAlign (RIGHT,BOTTOM);
    float angle1 = radians(90);
    translate(130, 240);
    rotate(angle1);
    text("Site 1", 0, 0);
      circle(20,-3,5);
  circle(13,-8,5);
  circle(20,-12,5);

    float angle2 = radians(0);
    translate(350, 0);
    rotate(angle2);
    text("Site 2  //", 0, 0);

    float angle3 = radians(0);
    translate(350, 0);
    rotate(angle3);
    text("Site 3", 0, 0);
    rect(17,-13,10,10);
  }
  fill(220,rawSensorData3/1.11);
  beginShape();
  vertex(x1+rawSensorData3/8, y3+rawSensorData3/12);
  bezierVertex(10+x3, 1+y3, 20+x3, rawSensorData3/8+y3, x3,
rawSensorData3/12+y3);
  endShape();
}
}
====================================
final String sketchname = getClass().getName();

import com.hamoid.*;
VideoExport videoExport;
```

```
void rec() {
  if (frameCount == 1) {
    videoExport = new VideoExport(this, "../"+sketchname+".mp4");
    videoExport.setFrameRate(30);
    videoExport.startMovie();
  }
  videoExport.saveFrame();
}
```