```
ellipseAPP
// IMA NYU Shanghai
// Interaction Lab
// For receiving multiple values from Arduino to Processing

/*
* Based on the readStringUntil() example by Tom Igoe
* https://processing.org/reference/libraries/serial/Serial_readStringUntil_.html
*/

import processing.serial.*;

String myString = null;
Serial myPort;


int NUM_OF_VALUES = 2; /** YOU MUST CHANGE THIS ACCORDING TO YOUR
PROJECT **/
int[] sensorValues; /** this array stores values from Arduino **/


void setup() {
 size(500, 500);
 background(0);
 setupSerial();
}


void draw() {
 updateSerial();
 printArray(sensorValues);

 // use the values like this!
 // sensorValues[0]

 // add your code
 background(0);
 fill(255);
 ellipse(map(sensorValues[0],0,1023,50,400),map(sensorValues[1],0,1023,50,400),100,100);

 //
}



void setupSerial() {
 printArray(Serial.list());
 myPort = new Serial(this, Serial.list()[ 1 ], 9600);
 // WARNING!
 // You will definitely get an error here.
 // Change the PORT_INDEX to 0 and try running it again.
 // And then, check the list of the ports,
```

```
  // find the port "/dev/cu.usbmodem----" or "/dev/tty.usbmodem----"
  // and replace PORT_INDEX above with the index number of the port.

  myPort.clear();
  // Throw out the first reading,
  // in case we started reading in the middle of a string from the sender.
  myString = myPort.readStringUntil( 10 );  // 10 = '\n' Linefeed in ASCII
  myString = null;

  sensorValues = new int[NUM_OF_VALUES];
}



void updateSerial() {
 while (myPort.available() > 0) {
   myString = myPort.readStringUntil( 10 ); // 10 = '\n' Linefeed in ASCII
   if (myString != null) {
     String[] serialInArray = split(trim(myString), ",");
     if (serialInArray.length == NUM_OF_VALUES) {
       for (int i=0; i<serialInArray.length; i++) {
         sensorValues[i] = int(serialInArray[i]);
       }
     }
   }
 }
}
```