//This code was written to be easy to understand.  
//Code efficiency was not considered.  
//Modify this code as you see fit.  
//This code will output data to the Arduino serial monitor.  
//Type commands into the Arduino serial monitor to control the DO circuit.  
//This code was written in the Arduino 1.6.5 IDE  
//An Arduino UNO was used to test this code.

#include <SoftwareSerial.h>
define rx 2
#define tx 3

//we have to include the SoftwareSerial library, or else we can't use it
//define what pin rx is going to be
//define what pin tx is going to be
SoftwareSerial myserial(rx, tx);

void setup() {
  Serial.begin(9600);
  myserial.begin(9600);
  inputstring.reserve(10);
  sensorstring.reserve(30);
}

void serialEvent() {
  inputstring = Serial.readStringUntil(13);
  input_string_complete = true;
}

String inputstring = "";
String sensorstring = "";
boolean input_string_complete = false;
boolean sensor_string_complete = false;

void loop() {  //here we go...
  if (input_string_complete){
    myserial.print(inputstring);
    myserial.print(\r\n);
    inputstring = "";
    input_string_complete = false;
  }

  if (myserial.available() > 0) {
    char inchar = (char)myserial.read();
    sensorstring += inchar;
    if (inchar == (\r)) {
      sensor_string_complete = true;
    }
  }

  if (sensor_string_complete== true) {
    Serial.println(sensorstring);
    if (isdigit(sensorstring[0])) {
      DO = sensorstring.toFloat();
      if (DO >= 6.0) {
        Serial.println("high");
      }
      if (DO <= 5.99) {
        Serial.println("low");
      }
    }
    sensorstring = "";
    sensor_string_complete = false;
  }
  }
}

//if a string from the PC has been received in its entirety
//send that string to the Atlas Scientific product
//add a <CR> to the end of the string
//clear the string
//reset the flag used to tell if we have received a completed string from the PC

//if a string from the Hardware serial port 0 receives a char
//read the string until we see a <CR>
//set the flag used to tell if we have received a completed string from the PC

//if a string from the Hardware serial port 0 receives a char
//we go...

//if the Hardware serial port 0 receives a char
//set the char to the char we just received
//if the first character in the string is a digit
//convert the string to a floating point number so it can be evaluated by the Arduino
//print "high" this is demonstrating that the Arduino is evaluating the DO
//as a number and not as a string
//if the DO is less than or equal to 5.99
//print "low" this is demonstrating that the Arduino is evaluating the DO
//as a number and not as a string
//clear the string
//reset the flag used to tell if we have received a completed string from the