Assignment-4

Write code and connections in work for ultrasonic sensor.Whenever distance is lessthan 100cms send "alert" to ibm cloud and display in device recent events.

```
#if ARDUINO >= 100
 #include <Arduino.h>
#else
 #include <WProgram.h>
#endif

#include "Ultrasonic.h"

Ultrasonic::Ultrasonic(uint8_t trigPin, uint8_t echoPin, unsigned long timeOut) {
 trig = trigPin;
 echo = echoPin;
 threePins = trig == echo ? true : false;
 pinMode(trig, OUTPUT);
 pinMode(echo, INPUT);
 timeout = timeOut;
}

unsigned int Ultrasonic::timing() {
 if (threePins)
  pinMode(trig, OUTPUT);

 digitalWrite(trig, LOW);
 delayMicroseconds(2);
 digitalWrite(trig, HIGH);
 delayMicroseconds(10);
 digitalWrite(trig, LOW);

 if (threePins)
  pinMode(trig, INPUT);
```

```
  previousMicros  = micros();
  while(!digitalRead(echo)  && (micros()  - previousMicros)  <= timeout);  //

  previousMicros  = micros();

  while(digitalRead(echo)  && (micros()  - previousMicros)  <= timeout);

  return micros()  - previousMicros;  // duration
}

unsigned  int Ultrasonic::read(uint8_t  und) {
  return timing()  / und / 2;
}


unsigned  int Ultrasonic::distanceRead(uint8_t  und) {
  return read(und);
}

#ifndef  Ultrasonic_h
#define  Ultrasonic_h

#define  CM 28
#define  INC 71
```

```
class Ultrasonic {
  public:
    Ultrasonic(uint8_t sigPin) : Ultrasonic(sigPin, sigPin) {};
    Ultrasonic(uint8_t trigPin, uint8_t echoPin, unsigned long timeOut = 20000UL);
    unsigned int read(uint8_t und = CM);
    unsigned int distanceRead(uint8_t und = CM) __attribute___((deprecated ("This
method is deprecated, use read() instead.")));
    void setTimeout(unsigned long timeOut) {timeout = timeOut;}
    void setMaxDistance(unsigned long dist) {timeout = dist*CM*2;}

  private:
    uint8_t trig;
    uint8_t echo;
    boolean threePins = false;
    unsigned long previousMicros;
    unsigned long timeout;
    unsigned int timing();
};

#endif
{
  "version": 1,
  "author": "Rozen Berg",
  "editor": "wokwi",
  "parts": [
    {
      "type": "wokwi-arduino-uno",
      "id": "uno",
      "top": 259.31,
      "left": 31.06,
      "rotate": 0,
      "hide": false,
      "attrs": {}
    },
    {
      "type": "wokwi-hc-sr04",
      "id": "ultrasonic",
      "top": 86.99,
      "left": 109.89,
      "rotate": 0,
      "hide": false,
      "attrs": { "distance": "100" }
    }
```

```
  ],
  "connections": [
    [ "uno:GND.1", "ultrasonic:GND", "black", [ "v-8", "*", "v8" ] ],
    [ "uno:13", "ultrasonic:ECHO", "green", [] ],
    [ "uno:12", "ultrasonic:TRIG", "purple", [ "*", "v4" ] ],
    [ "uno:5V", "ultrasonic:VCC", "red", [ "v16", "h-96", "*", "v12" ] ]
  ]
}
```

```
#include "Ultrasonic.h"


Ultrasonic ultrasonic(12, 13);
int distance;

void setup() {
  Serial.begin(9600);
}

void loop() {

distance = ultrasonic.read(CM);

Serial.print("Distance in CM: ");

Serial.println(distance);

distance = ultrasonic.read(INC);

Serial.print("Distance in Inches: ");

Serial.println(distance);

delay(1000);
}
```

.

```
  distance  = ultrasonic.read(CM);

  Serial.print("Distance  in  CM:
  "); Serial.println(distance);

  distance  = ultrasonic.read(INC);

  Serial.print("Distance  in  Inches:
  "); Serial.println(distance);

  delay(1000);
}
```

output: