```
// defines pins numbers
const int trigPin = 9;
const int echoPin = 10;
const int buzzer = 11;

long duration;
int distance;


// defines variables
void setup()
{
Serial.begin(9600);
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
pinMode(buzzer, OUTPUT);
pinMode(12,OUTPUT);

}
void loop() {
   long duration, cm;
  digitalWrite(12, LOW);   //Buzzer GND is always low


   //send a signal at ping pin at an interval of 0.002 seconds to check for an object
   digitalWrite(trigPin, LOW);
   delayMicroseconds(2);
   digitalWrite(trigPin, HIGH);
   delayMicroseconds(10);
   digitalWrite(trigPin, LOW);


   duration = pulseIn(echoPin, HIGH);  //check time using pulseIn function

   cm = microsecondsToCentimeters(duration);   //functin call to find distance

  /* Serial.print(cm);
   Serial.print("cm");
   Serial.println();
   delay(100);

   for debugging
   */


   if (cm<30&&cm>21)
                          {analogWrite(buzzer,1000);
                           delay(500);
```

```
                        analogWrite(buzzer,0);
                        delay(500); }    //sound buzzer every second if obstacle
distance is between 20-30cm.


   else if (cm<20&&cm>11) {analogWrite(buzzer,1000);
                         delay(250);
                         analogWrite(buzzer,0);
                         delay(250); }   //sound buzzer every 0.5 seconds if
obstacle distance is between 10-20cm.


   else if (cm<10&&cm>0)  {analogWrite(buzzer,1000);
                         delay(50);
                         analogWrite(buzzer,0);
                         delay(50); }    //sound buzzer every 0.1 seconds if
obstacle distance is between 0-10cm.


   else                    analogWrite(buzzer,0); //do not sound the buzzer
}



//function to return distance in cm from microseconds
long microsecondsToCentimeters(long microseconds) {
   return microseconds / 29 / 2;
}
```