Step by Step Circuit for Ultrasonic Sensor:

\* Ensure that the Arduino is switched off before starting to connect the components

1) On the bread board, place the Ultrasonic sensor such that the sensor faces you and the pins are placed horizontally on the board



2) From the Arduino controller connect a wire from GND to negative of the bread board on any one side

2a) Connect another wire from 5v to the positive of the bread board



3) Form the hole next to VCC of the ultrasonic sensor, connect a wire to the positive slot

3a) From the hole next to GND of the ultrasonic sensor, connect a wire to the negative slot



4) From the trig Pin on the Ultrasonic sensor connect a wire to PIN 11

4a) From the echo Pin on the Ultrasonic sensor connect a wire to PIN 12



Our Ultrasonic circuit is complete, next add an active sensor to the circuit.

5) Place a resistor ( ) on the breadboard , one end to E slot and one to F slot

5a) Place the buzzers positive side to the resistors other side and negative to some slot



6) From the G slot next to resistor, connect a wire to a positive slot that is free

6a) from the Other end of the buzzer connect a wire to a negative slot that is free

The circuit is now complete

* Ensure there is no loose wires or loosely kept pins from the buzzer or the sensor.
* Next step is to code for the Arduino , connect, upload and enjoy.

// defines pins numbers

const int trigPin = 11;

const int echoPin = 12;

const int buzpin = 9;

// defines variables

long duration;

int distance;

void setup() {

pinMode(trigPin, OUTPUT);

pinMode(buzpin, OUTPUT);// Sets the buzpin as an Output

pinMode(echoPin, INPUT); // Sets the echoPin as an Input

Serial.begin(9600); // Starts the serial communication

}

void loop() {

// Clears the trigPin

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds

duration = pulseIn(echoPin, HIGH);

// Calculating the distance

distance= duration\*0.034/2;

// Prints the distance on the Serial Monitor

Serial.print("Distance: ");

Serial.println(distance);

 Serial.print("Duration: ");

Serial.println(duration);

 if (distance < 100)

 {

 digitalWrite(buzpin, HIGH);

 } else {

 digitalWrite(buzpin, LOW);

 }

}