CODE FOR SHOE SENSOR

```
// Pin definitions
const int trigPin = A0;
const int echoPin = A1;
const int buzzerPin = A2; // Make sure your board allows digital write on A2

long duration;
int distance;

void setup() {
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(buzzerPin, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  // Send pulse
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  // Read echo
  duration = pulseIn(echoPin, HIGH);

  // Convert to cm
  distance = duration * 0.034 / 2;
  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println(" cm");

  // Buzzer alert
  if(distance < 20){
    digitalWrite(buzzerPin, HIGH);
  } else {
    digitalWrite(buzzerPin, LOW);
  }

  delay(100);
}
```

EXPLANATION FOR CODE

Step 1: Pin Definitions

trigPin (A0): Sends a pulse to trigger the ultrasonic sensor.

echoPin (A1): Listens for the returning pulse after reflecting off an object.

buzzerPin (A2): Controls a buzzer (turns on/off depending on distance).

Step 2: setup() Function

Sets trigPin and buzzerPin as outputs, echoPin as input.

Initializes Serial communication for debug output.

Step 3: loop() Function

A. Sending the Ultrasonic Pulse

Set trigPin LOW then HIGH (for 10 microseconds), then LOW again.

This sequence sends out a short ultrasonic pulse from the sensor.

B. Reading the Echo

duration = pulseIn(echoPin, HIGH); waits for the echo pin to go HIGH, and measures how long it stays HIGH. This represents the time for the reflected pulse to return.

C. Calculating Distance
distance=duration×0.0342

The speed of sound is about 0.034 cm per microsecond. Dividing by 2 accounts for the pulse traveling to the object and back.

D. Buzzer Alert
If distance < 20, the buzzer is turned ON (digitalWrite(buzzerPin, HIGH);).
Otherwise, the buzzer is OFF.

E. Serial Output and Loop Delay
The measured distance is printed to serial (could be viewed in the Serial Monitor).

Delay of 100 milliseconds before the next measurement.
Key Concepts Checked:
Pin modes: Why inputs and outputs matter; set-up in setup().

Pulse timing: Measuring how long it takes for a sound pulse to return lets us calculate distance.
Condition check: Using if to turn the buzzer on if something is close.

Serial debugging: Helps you see the actual readings in real time.

SCREENSHOT OF THE STIMULATION