# Push Button Counter and Temperature Sensor Display Project

## 1. Project Introduction

This project combines two fundamental embedded systems tasks:  
- A Push Button Counter using Arduino  
- A Temperature Sensor (LM35) Reading displayed on Serial Monitor or LCD  
  
This work is part of the internship requirements at CodTech and will culminate in a certificate of completion.

## 2. Components Required

- Arduino UNO  
- Push Button  
- 10kΩ Resistor (Pull-down)  
- LM35 Temperature Sensor  
- Breadboard and Jumper Wires  
- (Optional) 16x2 LCD Display

## 3. Push Button Counter

### Circuit Description

The push button is connected to digital pin 2 of the Arduino. A pull-down resistor ensures the pin reads LOW when the button is not pressed. On each press, the counter increases and the value is displayed on the Serial Monitor.

### Arduino Code

int buttonPin = 2;  
int count = 0;  
bool lastButtonState = LOW;  
  
void setup() {  
 pinMode(buttonPin, INPUT);  
 Serial.begin(9600);  
}  
  
void loop() {  
 bool currentState = digitalRead(buttonPin);  
 if (currentState == HIGH && lastButtonState == LOW) {  
 count++;  
 Serial.print("Button Pressed Count: ");  
 Serial.println(count);  
 delay(200); // Debounce  
 }  
 lastButtonState = currentState;  
}

## 4. Temperature Sensor Reading

### Circuit Description

The LM35 sensor outputs analog voltage proportional to temperature. It's connected to analog pin A0 of Arduino.

### Arduino Code (Serial Monitor)

int tempPin = A0;  
  
void setup() {  
 Serial.begin(9600);  
}  
  
void loop() {  
 int raw = analogRead(tempPin);  
 float voltage = raw \* (5.0 / 1023.0);  
 float temperatureC = voltage \* 100;  
 Serial.print("Temperature: ");  
 Serial.print(temperatureC);  
 Serial.println(" °C");  
 delay(1000);  
}

## 5. Optional: LCD Display

Instead of using the Serial Monitor, temperature data can be shown on an LCD display. Use the LiquidCrystal library.

## 6. Output Demonstration

Provide images or videos showing the push button count on the Serial Monitor and the temperature value either on Serial Monitor or LCD display.

## 7. Conclusion

This project demonstrates basic sensor interfacing, digital input handling, and data display using Arduino. It fulfills the CodTech internship requirement and provides hands-on experience in embedded system design.