```
// -----Connections-----
// 1.Connect both GND and V of both Sensors to GND and 3.3v respectively.
// 2.Connect Data pin of DHT22 sensor to PIN4 on ESP 32.
// 3.Connect SCL or SCK pin of BME380 to PIN 23(GPIO23) on ESP32.
// 4.Connect SDA or SDI pin of BME380 to PIN 21(GPIO21) on ESP32.

#include <WiFi.h>
#include "ThingSpeak.h"
#include <Adafruit_BME280.h>
#include <Adafruit_Sensor.h>

#include "DHT.h"

#define DHTTYPE DHT22 // DHT 22

#define SEALEVELPRESSURE_HPA (1013.25)

const char *ssid = "Ajay";         // your network SSID (name)
const char *password = "test1235"; // your network password

WiFiClient client;

unsigned long myChannelNumber = 4;
const char *myWriteAPIKey = "J03E7XJOB6I18SL2";

// Timer variables
unsigned long lastTime = 0;
unsigned long timerDelay = 1000;

// Variable to hold temperature readings
float temperatureC;
float humidity;
float pressure;
float altitude;

// Create a sensor object
Adafruit_BME280 bme;

// DHT Sensor
uint8_t DHTPin = 4;

DHT dht(DHTPin, DHTTYPE);

void initBME()
{
  if (!bme.begin(0x76))
  {
    Serial.println("Could not find a valid BME280 sensor, check wiring!");
    while (1)
      ;
  }
}

void setup()
{
  Serial.begin(115200); // Initialize serial
  initBME();

  pinMode(DHTPin, INPUT);
  dht.begin();

  WiFi.mode(WIFI_STA);

  ThingSpeak.begin(client); // Initialize ThingSpeak
}
```

```
void loop()
{
  if ((millis() - lastTime) > timerDelay)
  {

    // Connect or reconnect to WiFi
    if (WiFi.status() != WL_CONNECTED)
    {
      Serial.print("Attempting to connect");
      while (WiFi.status() != WL_CONNECTED)
      {
        WiFi.begin(ssid, password);
        delay(5000);
      }
      Serial.println("\nConnected.");
    }

    temperatureC = bme.readTemperature();
    Serial.print("Temperature (ºC): ");
    Serial.println(temperatureC);

    pressure = bme.readPressure() / 100.0F;
    Serial.print("Pressure (hPa): ");
    Serial.println(pressure);

    altitude = bme.readAltitude(SEALEVELPRESSURE_HPA);
    Serial.print("Approx. Altitude = ");
    Serial.println(altitude);

    humidity = dht.readHumidity();
    Serial.print("Humidity (%): ");
    Serial.println(humidity);

    // set the fields with the values
    ThingSpeak.setField(1, temperatureC);
    ThingSpeak.setField(2, humidity);
    ThingSpeak.setField(3, pressure);
    ThingSpeak.setField(4, altitude);

    int x = ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);

    if (x == 200)
    {
      Serial.println("Channel update successful.");
    }
    else
    {
      Serial.println("Problem updating channel. HTTP error code " +
String(x));
    }
    lastTime = millis();
  }
}
```