Connecting NodeMcu with thingspeak :-

Setting up thingspeak :

- Before programming the Node mcu the creditionals and the data must be initialized in Thingspeak website.
- Visit the thingspeak website from the browser.
- By clicking the profile set up the Mathworks account that is to be used with thingspeak.
- After setting up a page for creating new channels is prompted .
- From here we can access the previously created channels as well as we can also create a new channel to use with new projects.
- Click new channel to create a new channel.
- In the next step input fields for data required to the new channel will be opened.
- Enter the choosed channel name and fields required for the sensors or the inputs configured to node mcu.
- The field name corresponds to the sensor name or input data field name that is assigned in the Nodemcu from which the data is collected.
- We can create upto 8 fields in one channel.
- Now save the channel then the main dashboard for the channel appears.
- The data for each field is visualized as a chart under the channel stats.
- Using the edit icon rename the chart name with the name of input used or the sensor used.
- Go to API keys and copy the write API key which is further required to importing the collected data to the channel.
- The API key acts as identification address to this channel from which it can the channel can be accessed.
- Now the setup is completed in thingspeak and the further steps need to be carried out in NodeMcu programming.

Setting up Node mcu for thingspeak :

- The Arduino IDE is used for compiling and uploading the Arduino code.
- The thingspeak provides a separate library for accessing thingspeak from NodeMcu and other Arduino boards.
- Under Sketch – Include Library – manage library search for thingspeak and install the library
- Connect the required sensors to Node mcu , whose data to be synced with thingspeak.
- Import the library in the sketch and provide the API keys and channel Id.
- Use WifiClient class to request and publish data to the Thingspeak cloud.
- Under Thingspeak.writeField method pass Channel number , field number, data and Write API key as arguments.
- This function writes the required data to the thingspeak cloud.
- The value returned by the function is stored under a variable.
- The variable stores the output response from the thingspeak server which says whether the data in the field is updated or not.
- Then compile the code and upload it the Node Mcu board.
- Now on resetting the board the sensor data is updated in the thingspeak cloud and the data is visualized in the corresponding field chart in the main dashboard.
- The data can also be exported as spreadsheet using the export recent data option in the dashboard.

This is the procedure to connect Node mcu board with the thingspeak cloud and visualize the data .

PROGRAM for NODE MCU TO PUBLISH DATA TO THINGSPEAK :-

```
#include <ESP8266WiFi.h>
#include "ThingSpeak.h"

const char* ssid = "SSID";
const char* password = "Password";
const int temp = A0;

WiFiClient  client;

unsigned long myChannelNumber = X;
const char * myWriteAPIKey = "XXXXXXXXXXXXXXX";

float temperatureC;

void setup() {
  Serial.begin(9600);
  Delay(1000);
  pinMode(temp,INPUT);
  WiFi.mode(WIFI_STA);
  ThingSpeak.begin(client);
}

void loop() {
    WiFi.begin(ssid, password);
    Delay(5000);
    Serial.print(“wifi connected”);
    temperatureC = analogRead(temp);
    Serial.print("Temperature (ºC): ");
    Serial.println(temperatureC);

    int x = ThingSpeak.writeField(myChannelNumber, 1, temperatureC,
myWriteAPIKey);

    if(x == 200){
      Serial.println("Channel update successful.");
    }
    else{
      Serial.println(“error updating channel”);
    }
}
```