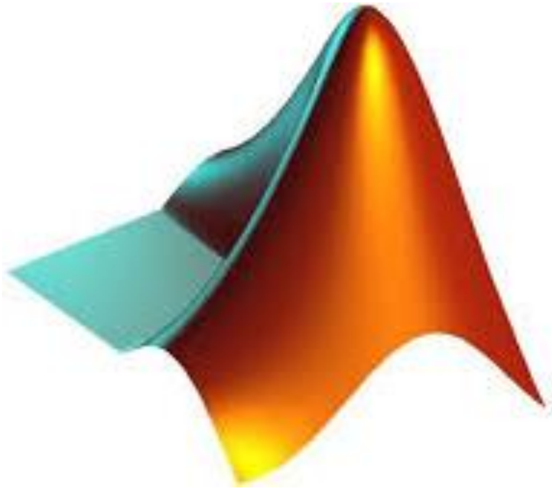
carenuty

Getting started With ThingSpeak & MATLAB

By Cynthia Kipruto

Overview

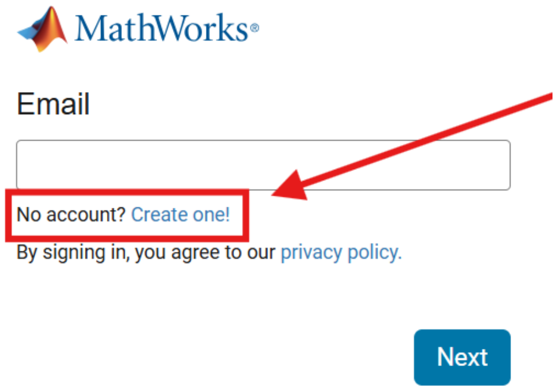
- Introduction to IoT data logging with ThingSpeak.
- Setting up ThingSpeak account, channels, and fields.
- Connecting sensors to log data.
- Visualizing data with graphs and MATLAB.



Creating a ThingSpeak Account

Steps:

- Go to *https://thingspeak.mathworks.com/*
- Click on Get started for free and Create an Account



MathWorks®

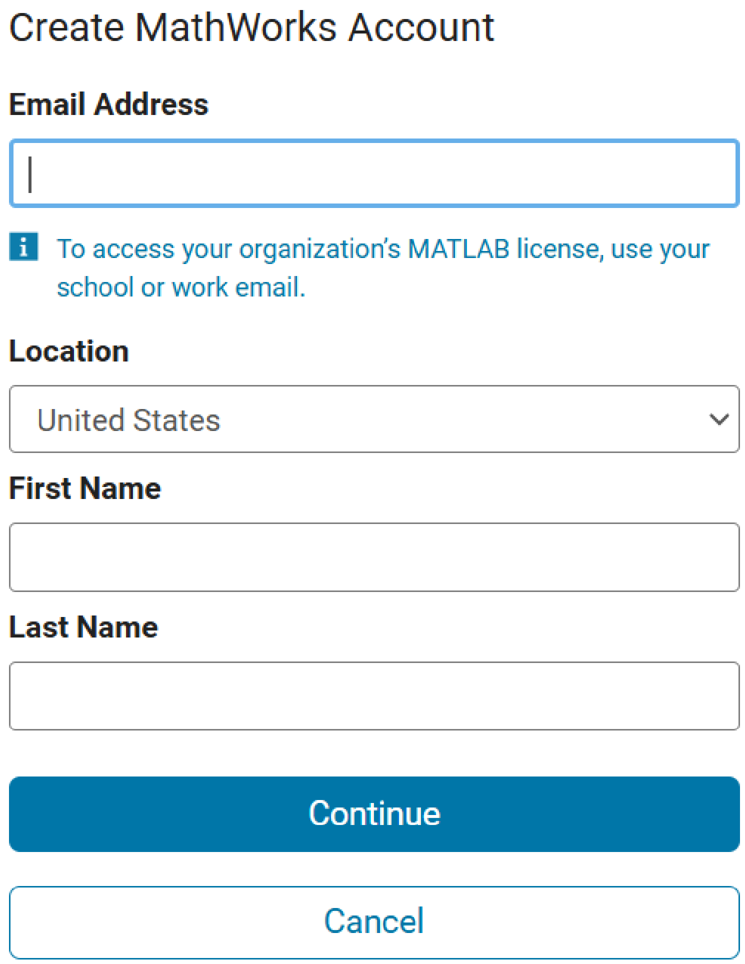
Email

No account? Create one!

By signing in, you agree to our privacy policy.

- Fill in your details (name, email, password).
- Confirm your email to activate the account

And set up your password.



Create MathWorks Account

Email Address

To access your organization's MATLAB license, use your school or work email.

Location

United States

First Name

Last Name

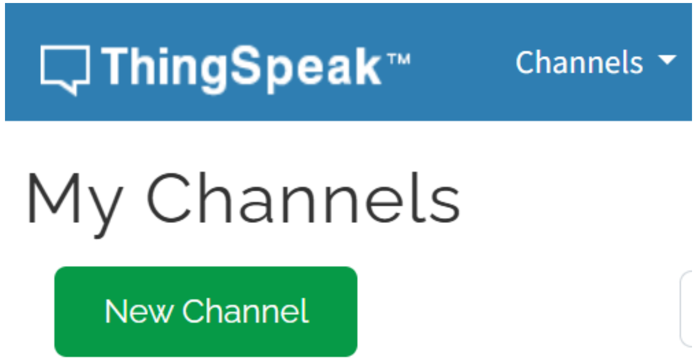
Creating a Channel

What is a Channel?

- A channel is where data from your sensor gets stored.

Steps:

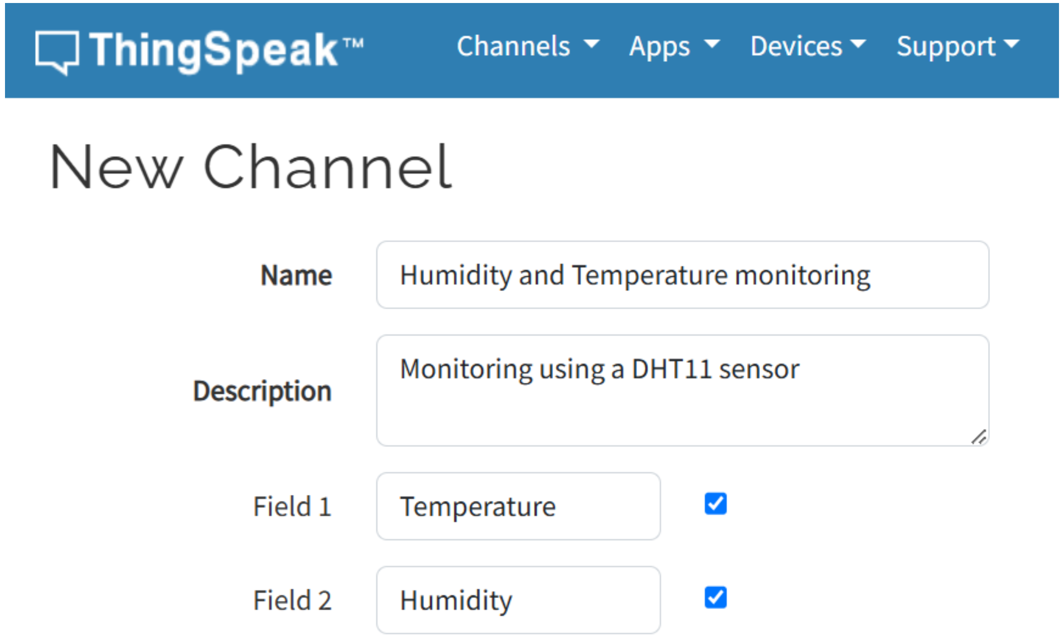
1. Navigate to Channels > My Channels > New Channel.



2. Provide a name and description.

3. Create relevant Fields (e.g., Temperature, Humidity).

6. Click Save Channel.



New Channel

Name: Humidity and Temperature monitoring

Description: Monitoring using a DHT11 sensor

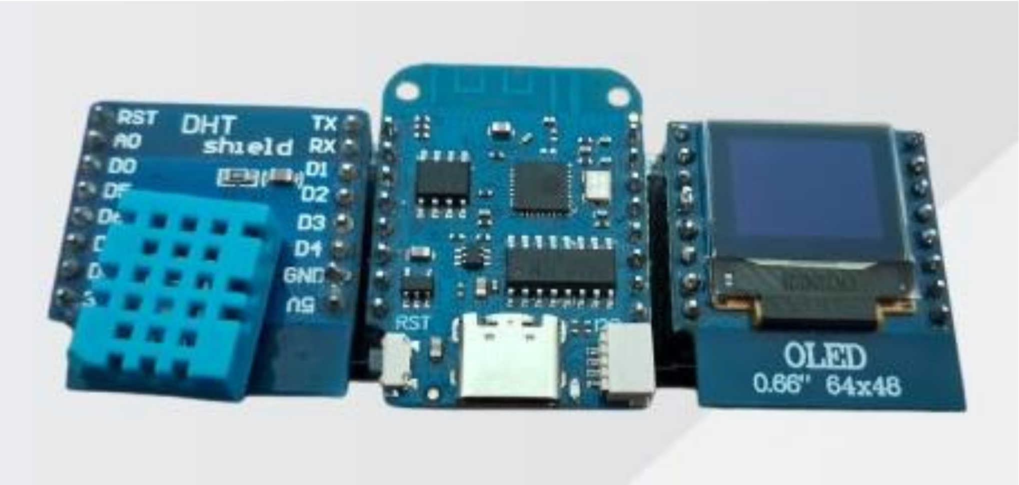
Field 1: Temperature ☑

Field 2: Humidity ☑

Connecting a Sensor to ThingSpeak

Hardware Requirements:

- Microcontroller (e.g., D1 Mini).
- Sensor (e.g., DHT11 for Temperature and Humidity).
- Display (Optional)
- Internet connection (WiFi).



Connect Sensor:

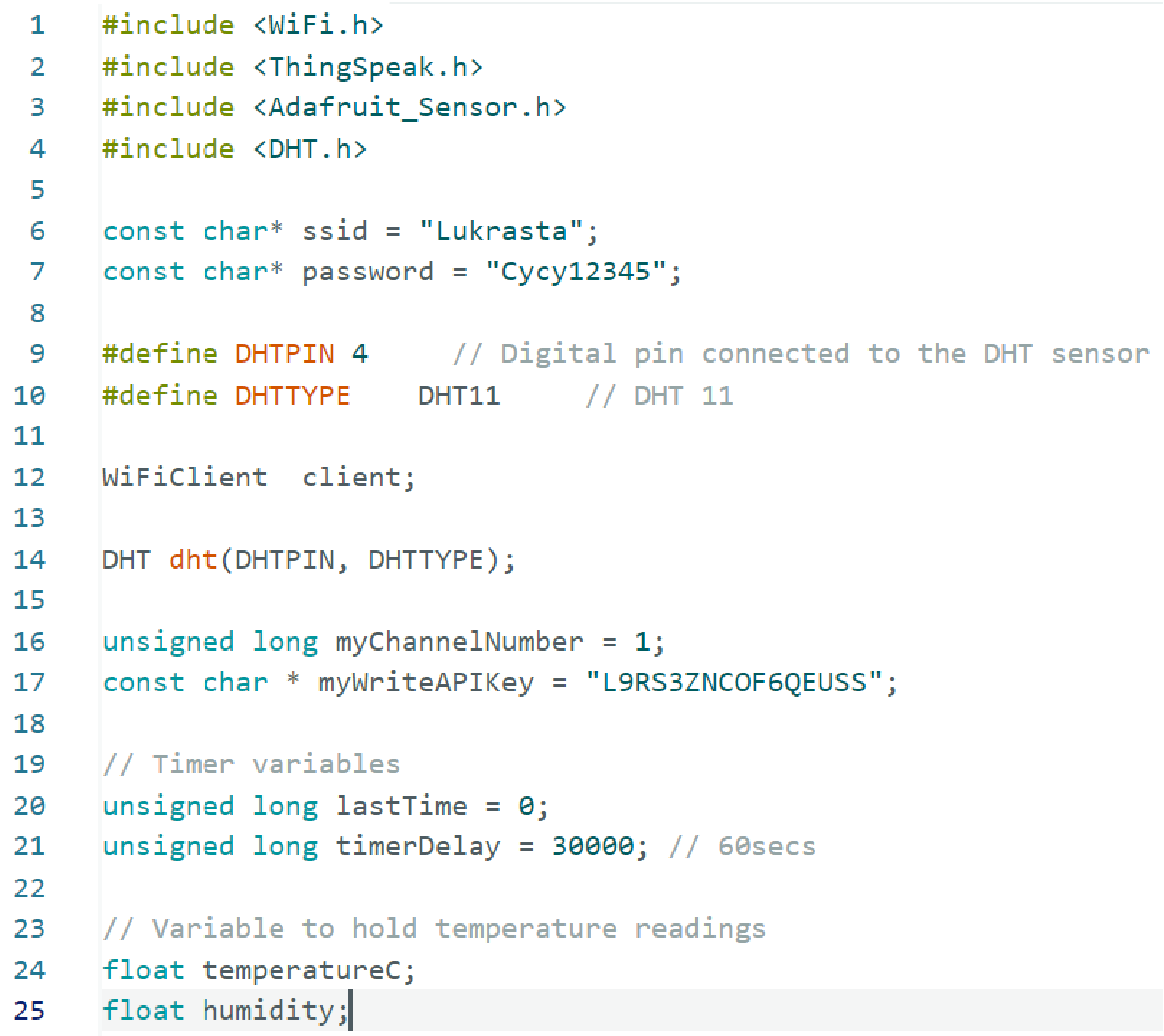
1. Program the microcontroller to send data to ThingSpeak via Arduino IDE.

2. Use the **Write API Key** from your channel.

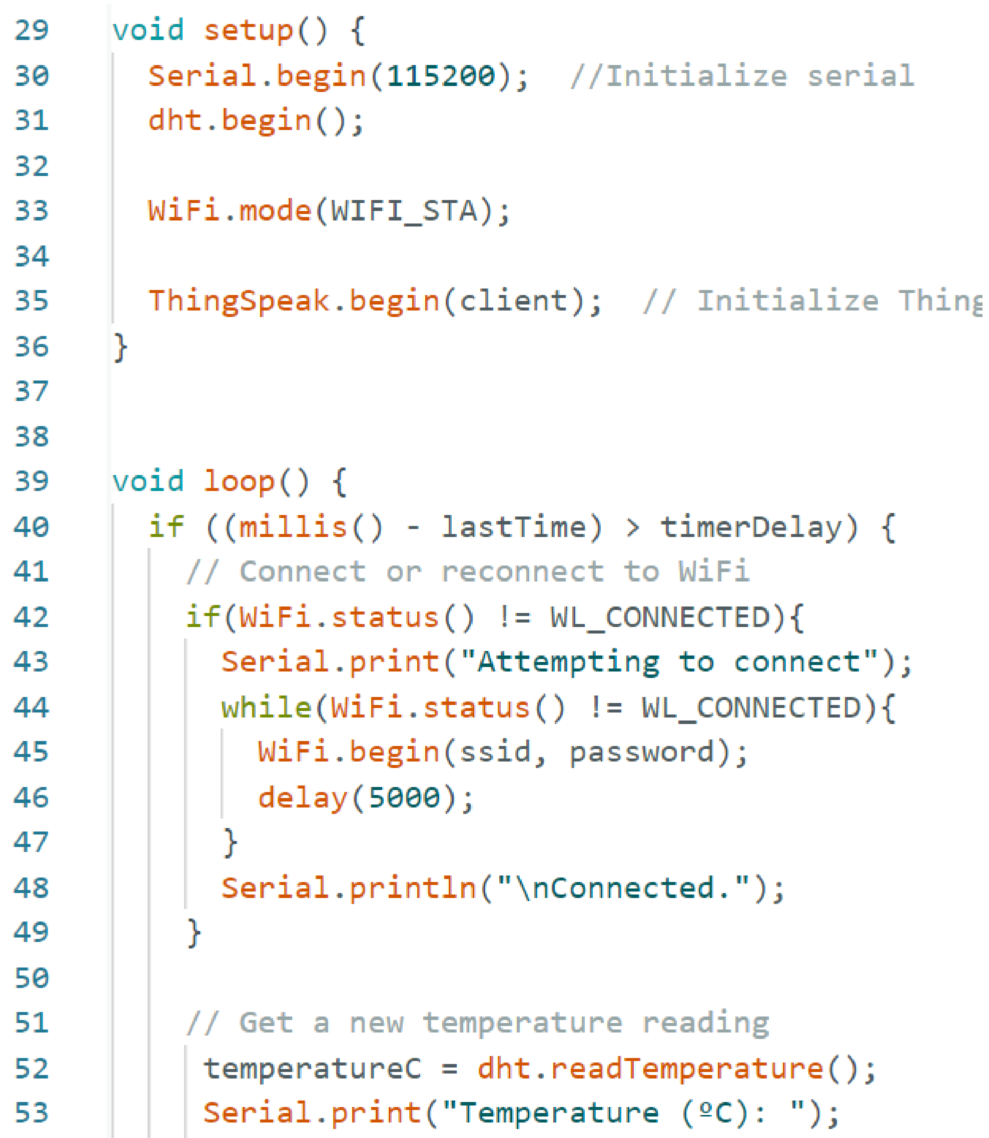
3. Install the necessary libraries

3. Upload the code to the microcontroller.

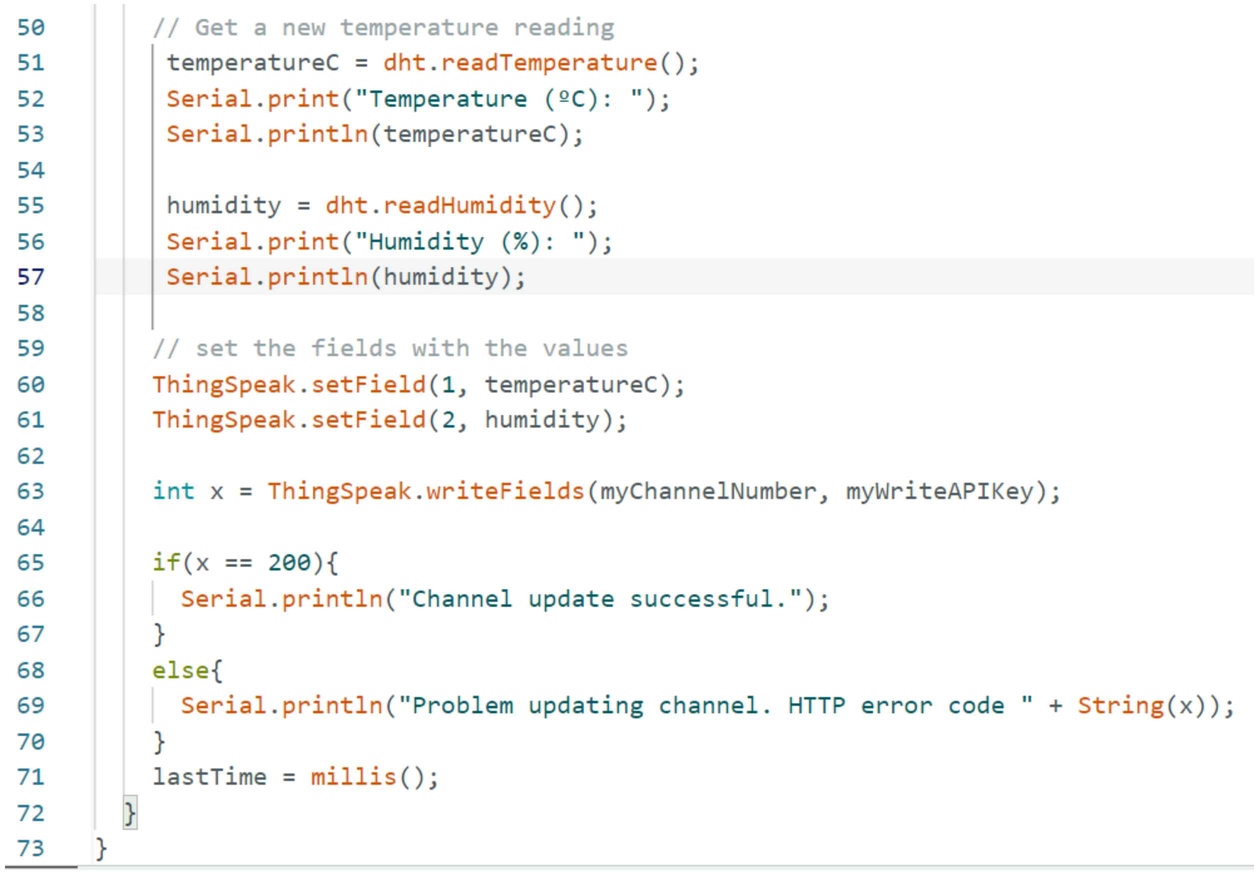
4. Verify if data is being sent successfully.



```
#include <WiFi.h>
#include <ThingSpeak.h>
#include <Adafruit_Sensor.h>
#include <DHT.h>

const char* ssid = "Lukrasta";
const char* password = "Cycy12345";

#define DHTPIN 4     // Digital pin connected to the DHT sensor
#define DHTTYPE    DHT11     // DHT 11

WiFiClient  client;

DHT dht(DHTPIN, DHTTYPE);

unsigned long myChannelNumber = 1;
const char * myWriteAPIKey = "L9RS3ZNCOF6QEUSS";

// Timer variables
unsigned long lastTime = 0;
unsigned long timerDelay = 30000; // 60secs

// Variable to hold temperature readings
float temperatureC;
float humidity;
```

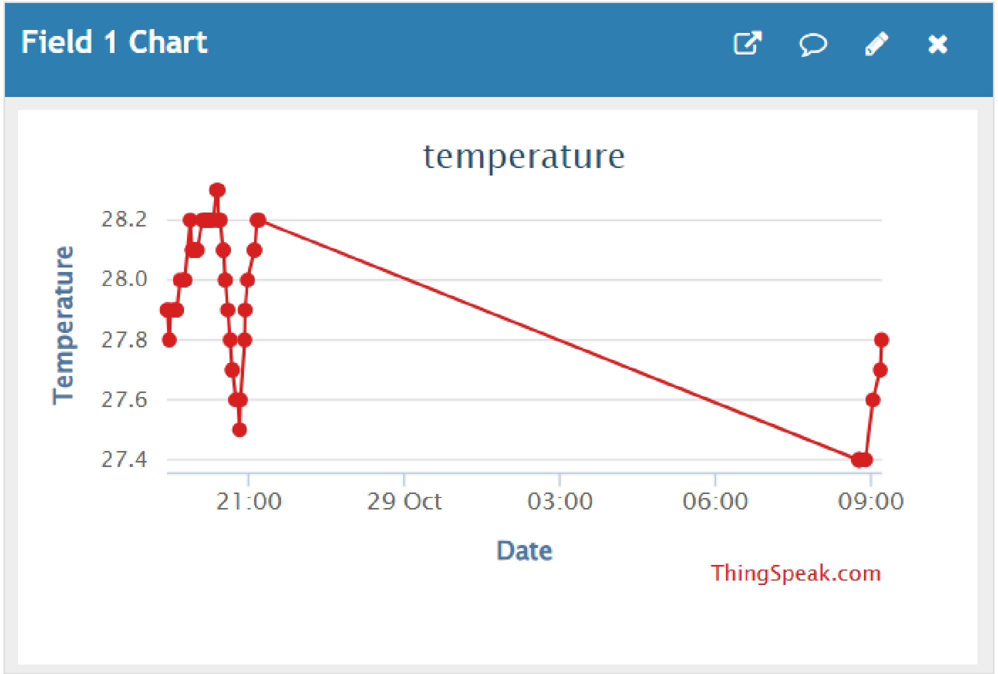


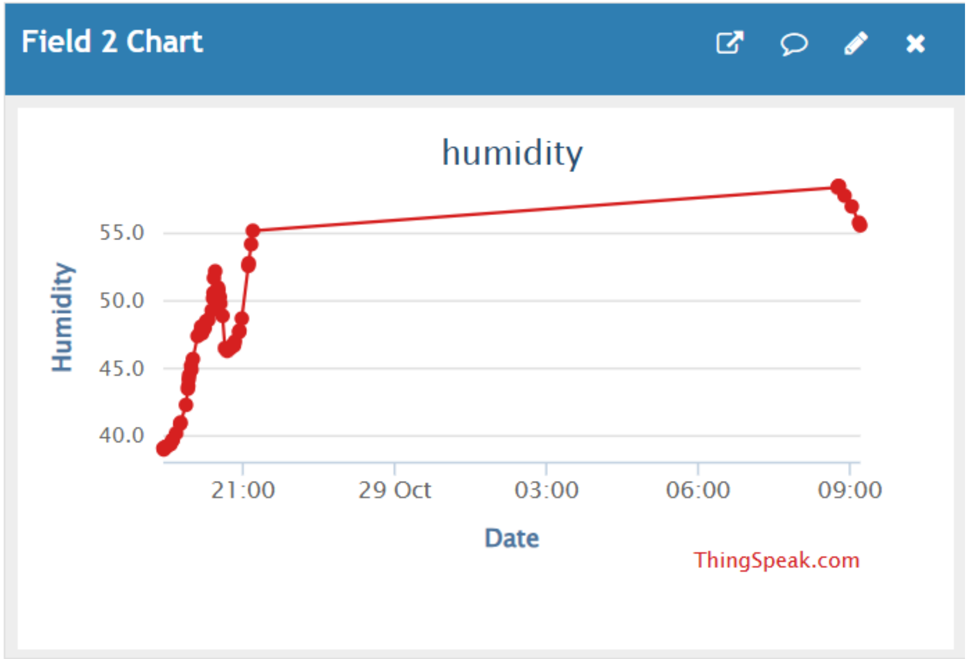
```
void setup() {
  Serial.begin(115200);  //Initialize serial
  dht.begin();

  WiFi.mode(WIFI_STA);

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


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.");
    }

    // Get a new temperature reading
    temperatureC = dht.readTemperature();
    Serial.print("Temperature (ºC): ");
```



```
    // Get a new temperature reading
    temperatureC = dht.readTemperature();
    Serial.print("Temperature (ºC): ");
    Serial.println(temperatureC);

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

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

    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();
  }
}
```

carenuity


Field 1 Chart
temperature
Temperature
28.2
28.0
27.8
27.6
27.4
21:00
29 Oct
03:00
06:00
09:00
Date
ThingSpeak.com

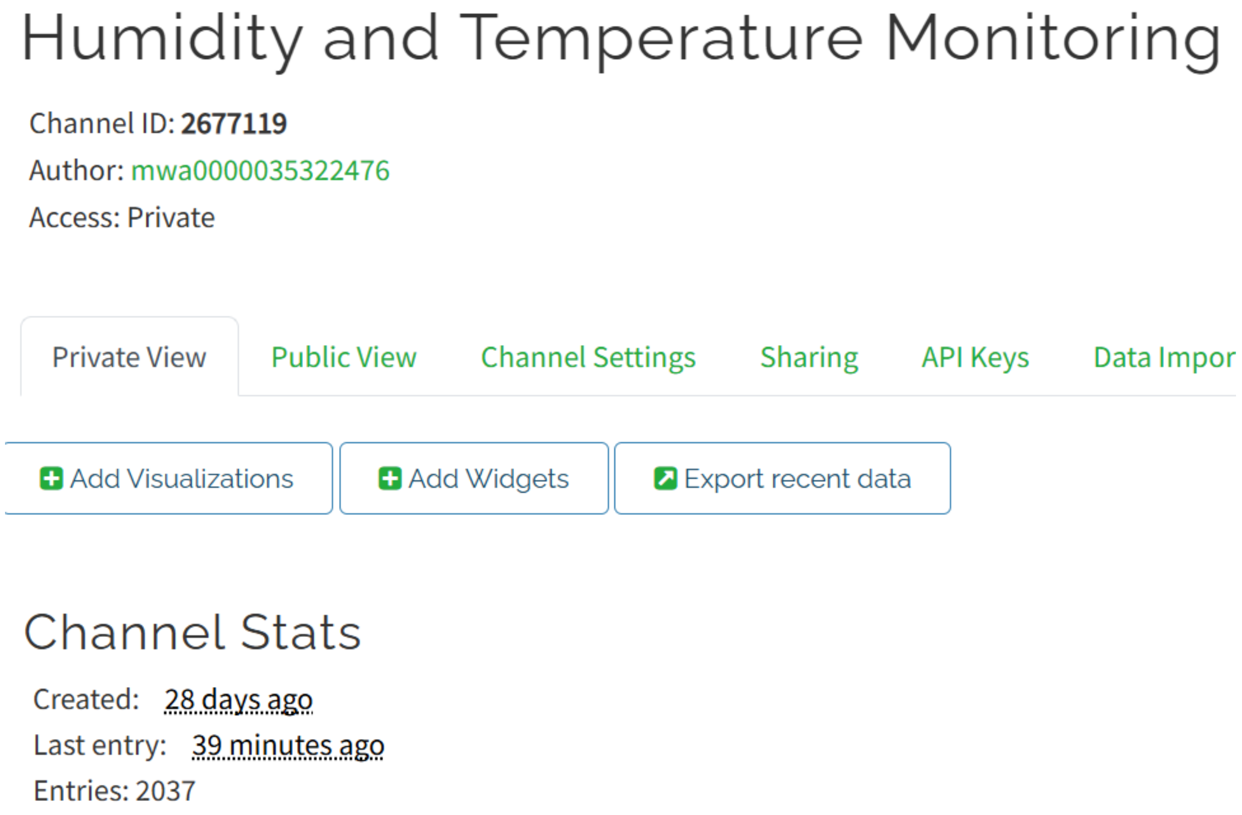

Field 2 Chart
humidity
Humidity
55.0
50.0
45.0
40.0
21:00
29 Oct
03:00
06:00
09:00
Date
ThingSpeak.com

Visualizing Data on ThingSpeak

carenuty

How to visualize data:

1. Open your channel.
2. Go to Private View
3. Scroll to see the generated graphs.
4. Customize the time range or graph style.



Humidity and Temperature Monitoring

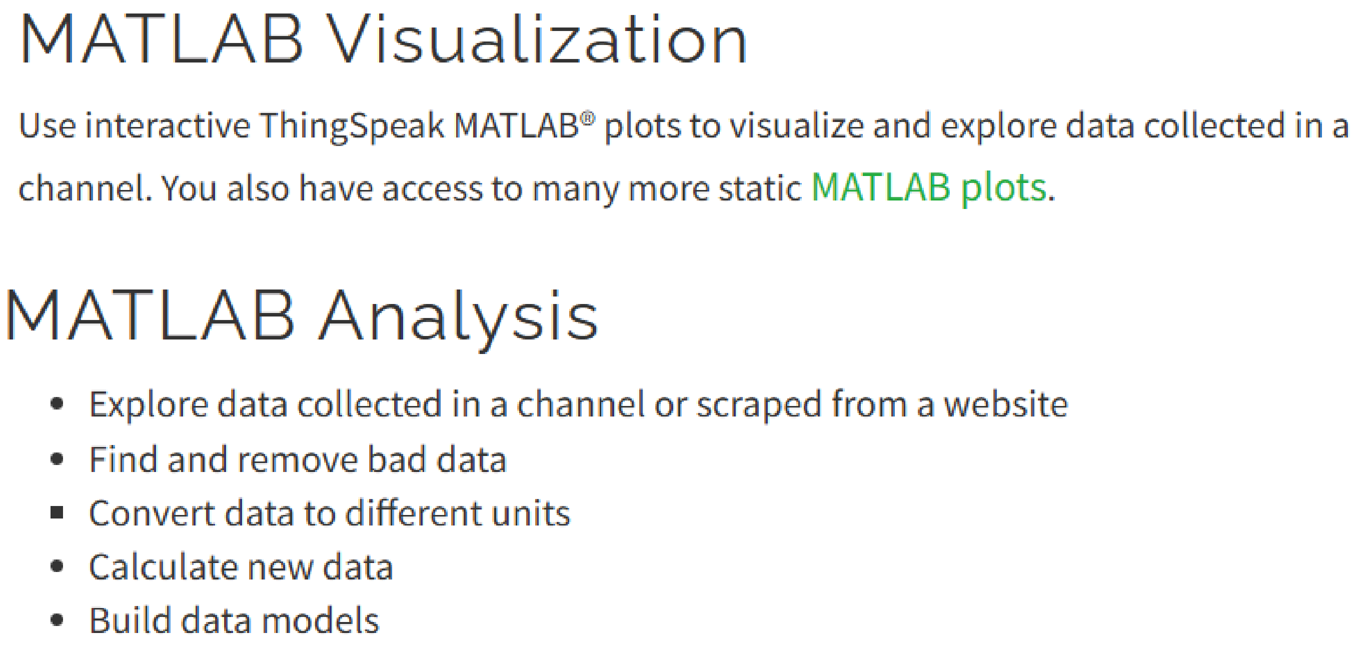
Channel ID: **2677119**
Author: mwa0000035322476
Access: Private

Channel Stats

Created: 28 days ago
Last entry: 39 minutes ago
Entries: 2037

MATLAB in ThingSpeak

- MATLAB Analysis: Perform operations on data.
- MATLAB Visualization: Create customized plots.



MATLAB Visualization

Use interactive ThingSpeak MATLAB® plots to visualize and explore data collected in a channel. You also have access to many more static MATLAB plots.

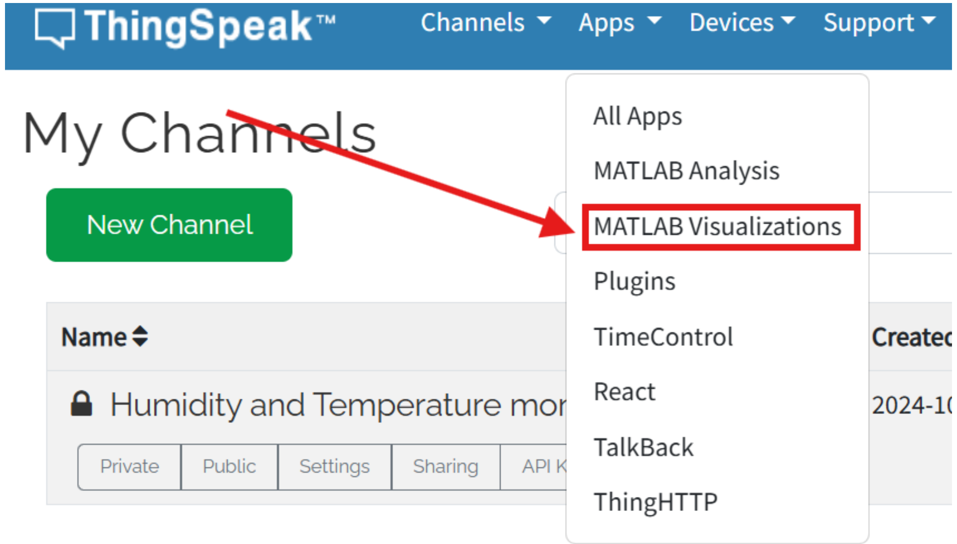
MATLAB Analysis

- Explore data collected in a channel or scraped from a website
- Find and remove bad data
- Convert data to different units
- Calculate new data
- Build data models

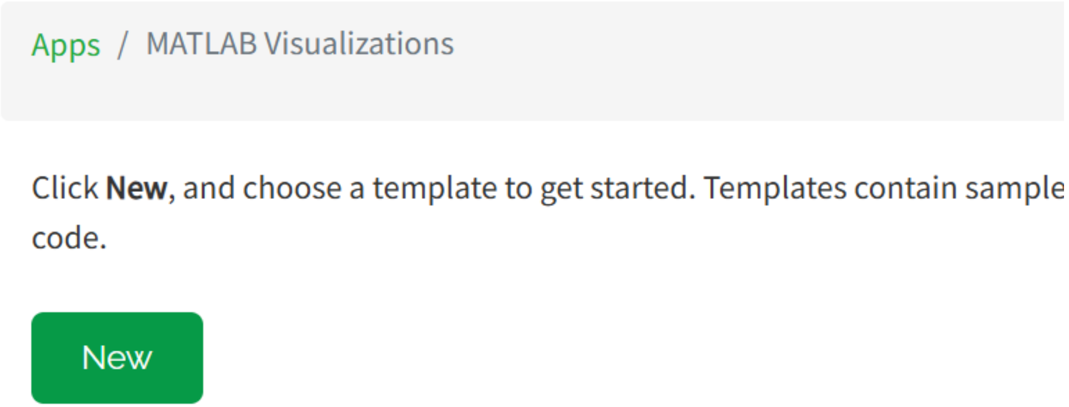
Writing a MATLAB Script to Visualize Data

Steps :

Go to Apps > MATLAB Visualization.

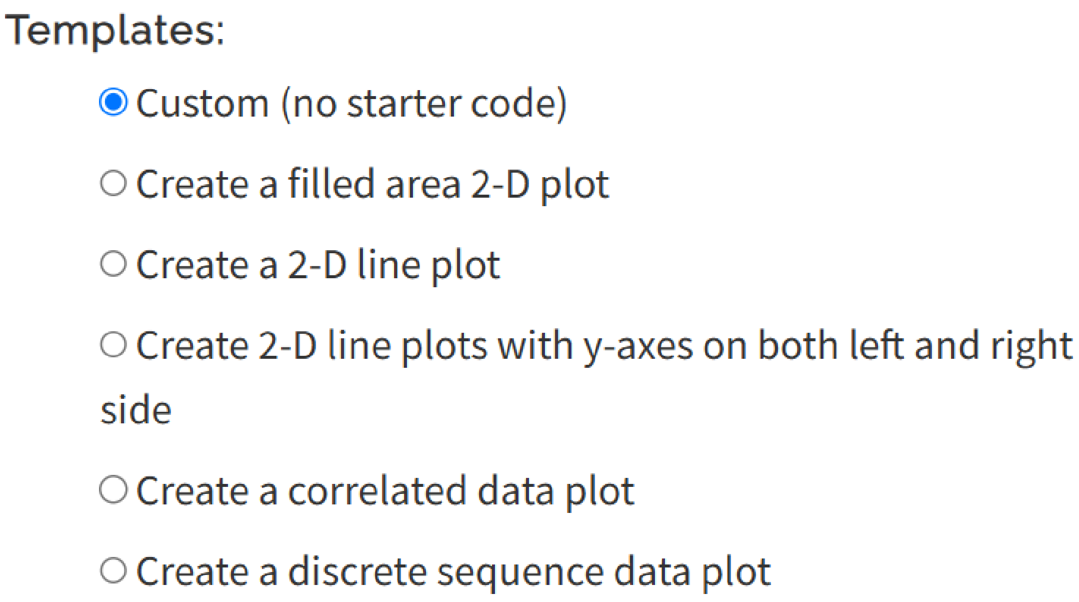


Click **New** to create a script.



Click **New**, and choose a template to get started. Templates contain sample code.

- Select Custom (no starter code)\



Templates:

● Custom (no starter code)

○ Create a filled area 2-D plot

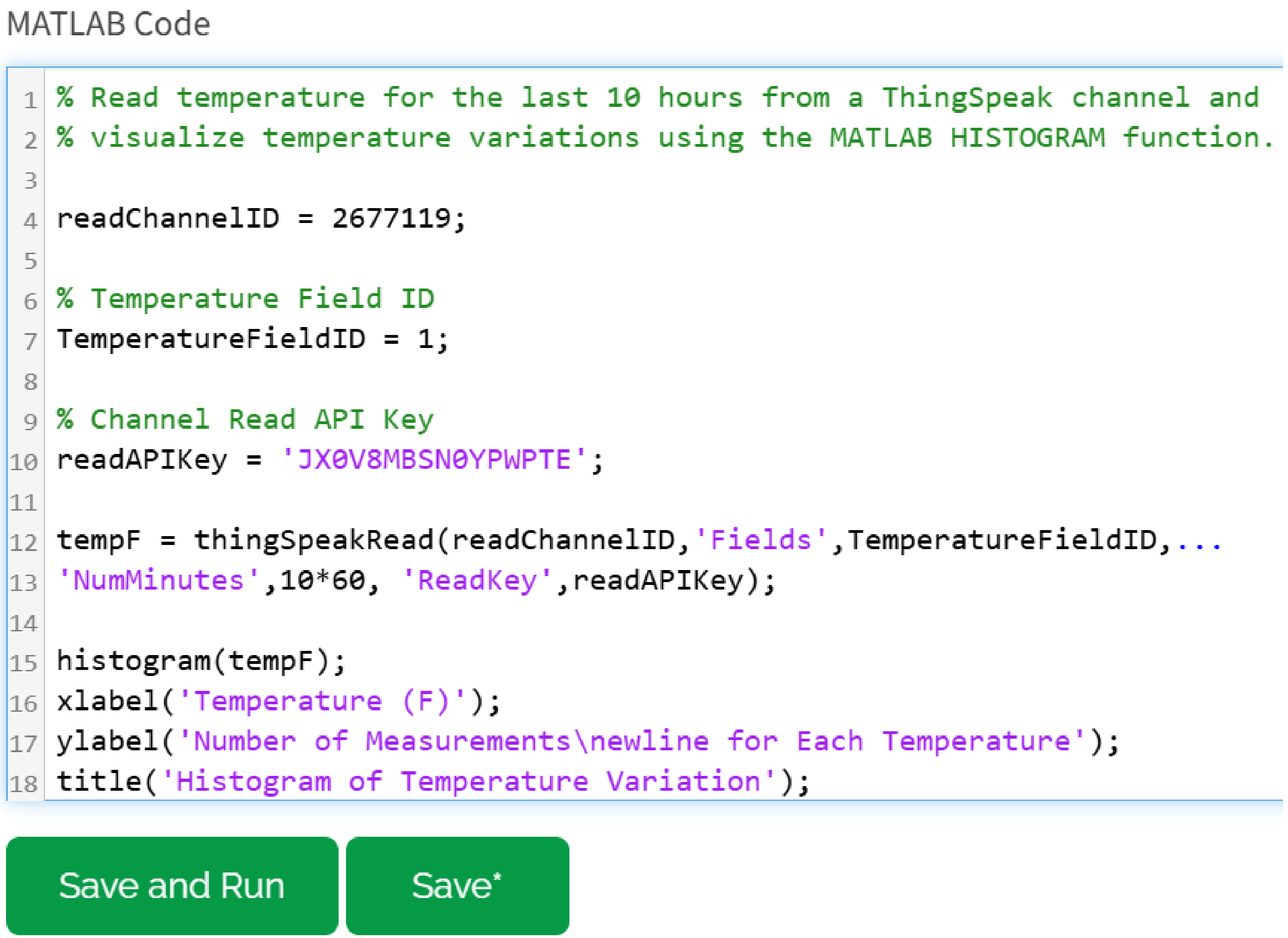
○ Create a 2-D line plot

○ Create 2-D line plots with y-axes on both left and right side

○ Create a correlated data plot

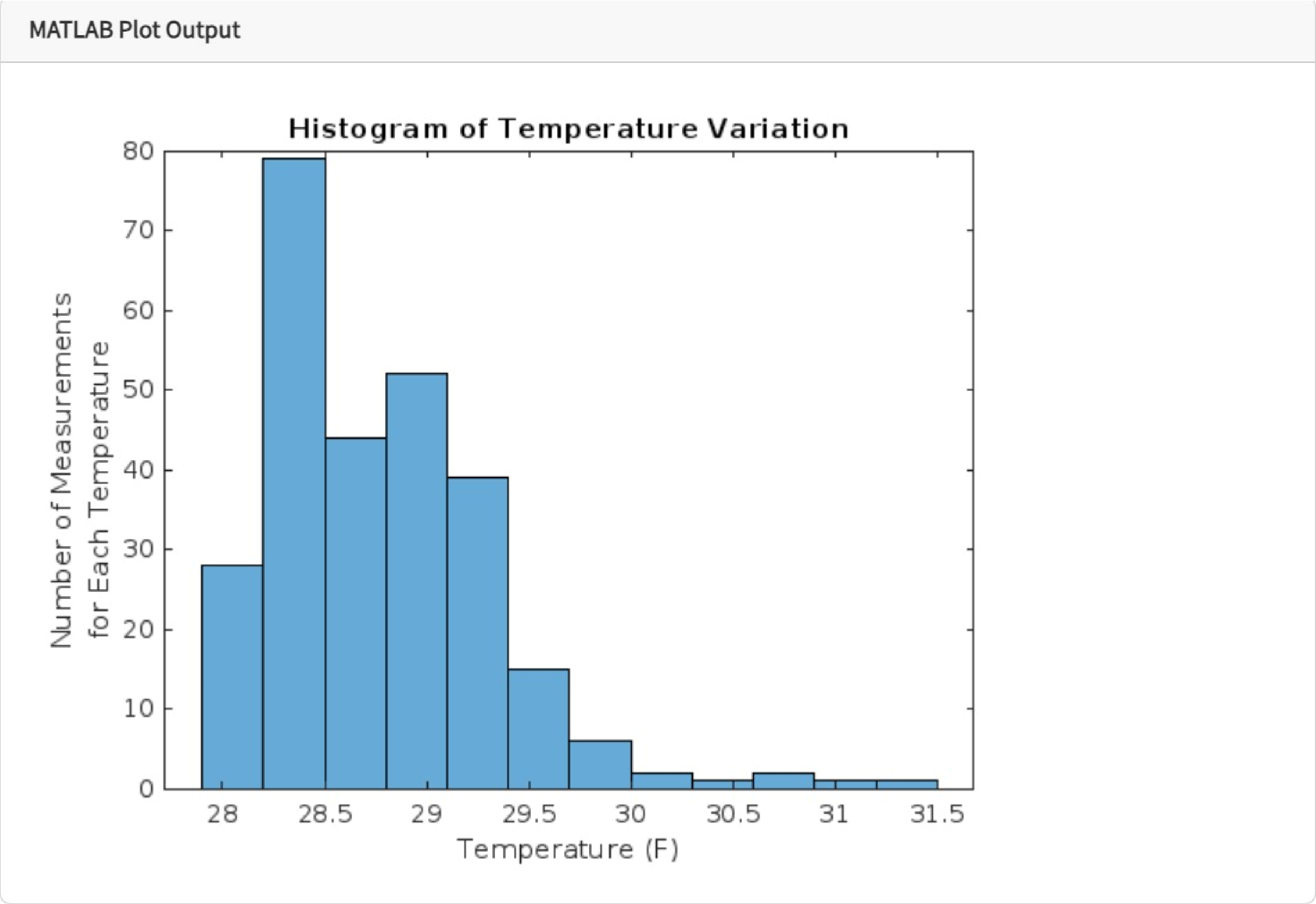
○ Create a discrete sequence data plot

- Click create and write your MATLAB code (e.g., read and plot data), Save and Run.



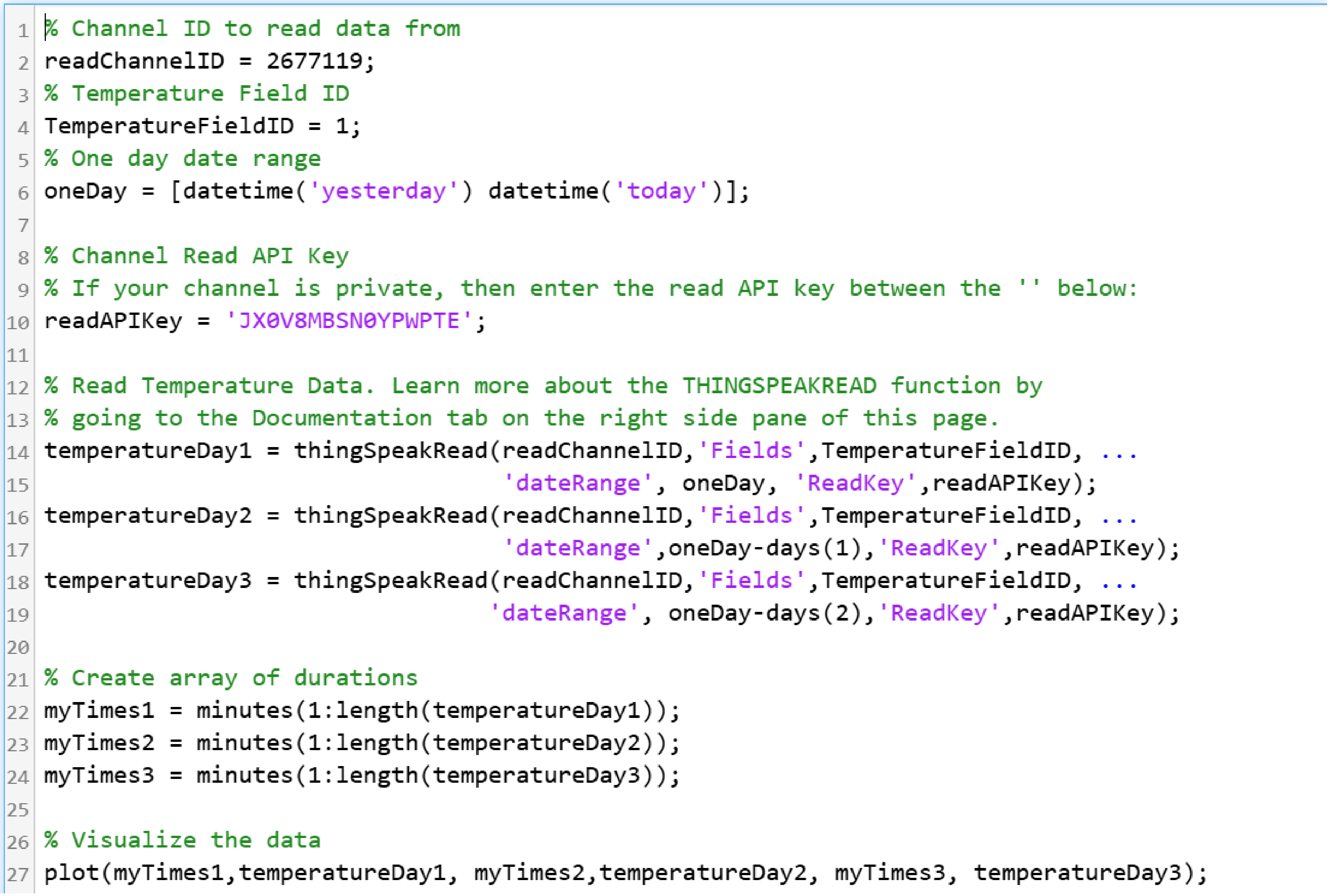
MATLAB Code

```
% Read temperature for the last 10 hours from a ThingSpeak channel and
% visualize temperature variations using the MATLAB HISTOGRAM function.

readChannelID = 2677119;

% Temperature Field ID
TemperatureFieldID = 1;

% Channel Read API Key
readAPIKey = 'JX0V8MBSN0YPWPTE';

tempF = thingSpeakRead(readChannelID,'Fields',TemperatureFieldID,...
'NumMinutes',10*60, 'ReadKey',readAPIKey);

histogram(tempF);
xlabel('Temperature (F)');
ylabel('Number of Measurements\newline for Each Temperature');
title('Histogram of Temperature Variation');
```

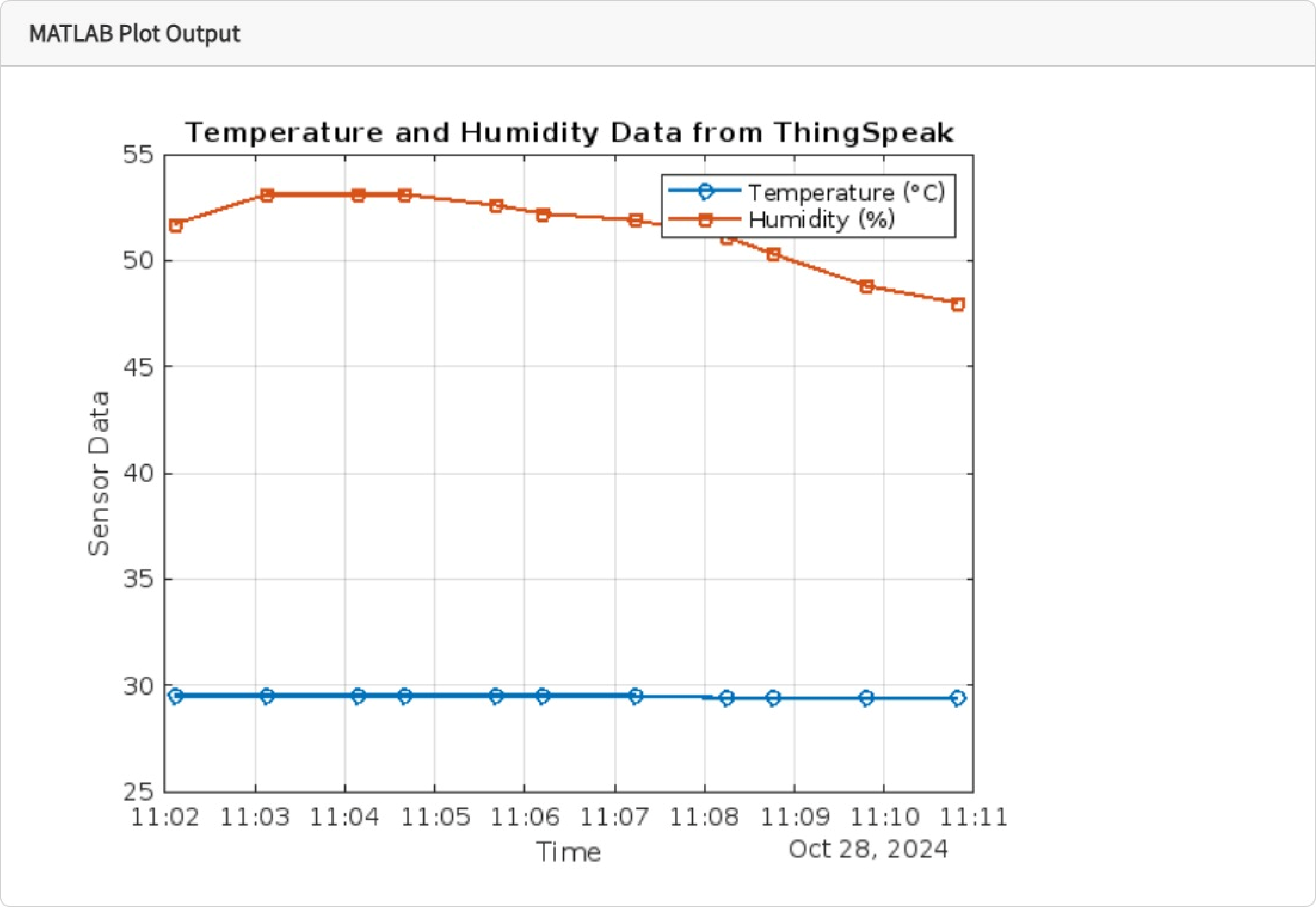


MATLAB Plot Output

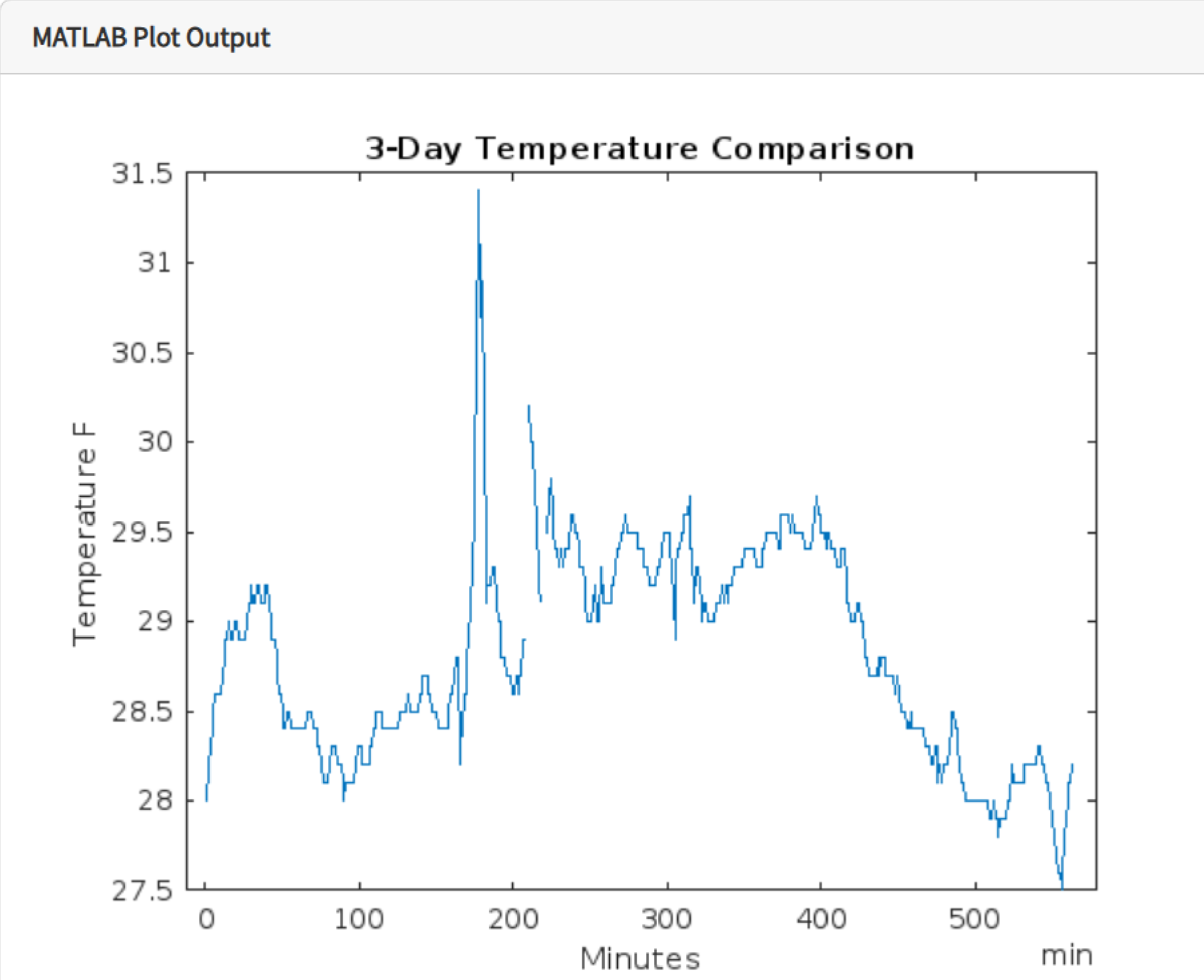
MATLAB Code



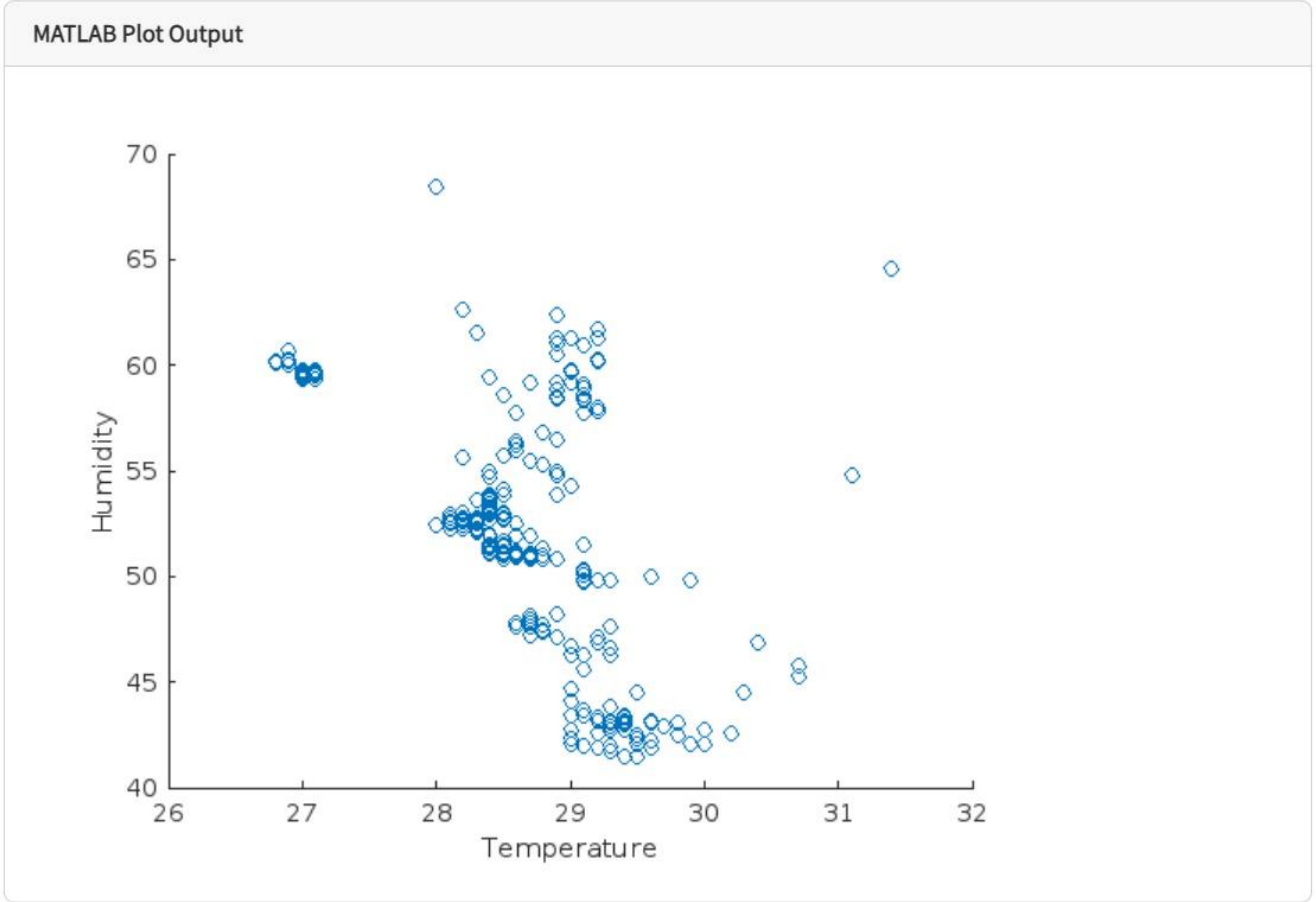
```matlab
% Channel ID to read data from
readChannelID = 2677119;
% Temperature Field ID
TemperatureFieldID = 1;
% One day date range
oneDay = [datetime('yesterday') datetime('today')];

% Channel Read API Key
% If your channel is private, then enter the read API key between the '' below:
readAPIKey = 'JX0V8MBSN0YPWPTE';

% Read Temperature Data. Learn more about the THINGSPEAKREAD function by
% going to the Documentation tab on the right side pane of this page.
temperatureDay1 = thingSpeakRead(readChannelID,'Fields',TemperatureFieldID, ...
                                 'dateRange', oneDay, 'ReadKey',readAPIKey);
temperatureDay2 = thingSpeakRead(readChannelID,'Fields',TemperatureFieldID, ...
                                 'dateRange',oneDay-days(1),'ReadKey',readAPIKey);
temperatureDay3 = thingSpeakRead(readChannelID,'Fields',TemperatureFieldID, ...
                                'dateRange', oneDay-days(2),'ReadKey',readAPIKey);

% Create array of durations
myTimes1 = minutes(1:length(temperatureDay1));
myTimes2 = minutes(1:length(temperatureDay2));
myTimes3 = minutes(1:length(temperatureDay3));

% Visualize the data
plot(myTimes1,temperatureDay1, myTimes2,temperatureDay2, myTimes3, temperatureDay3);
```



MATLAB Plot Output

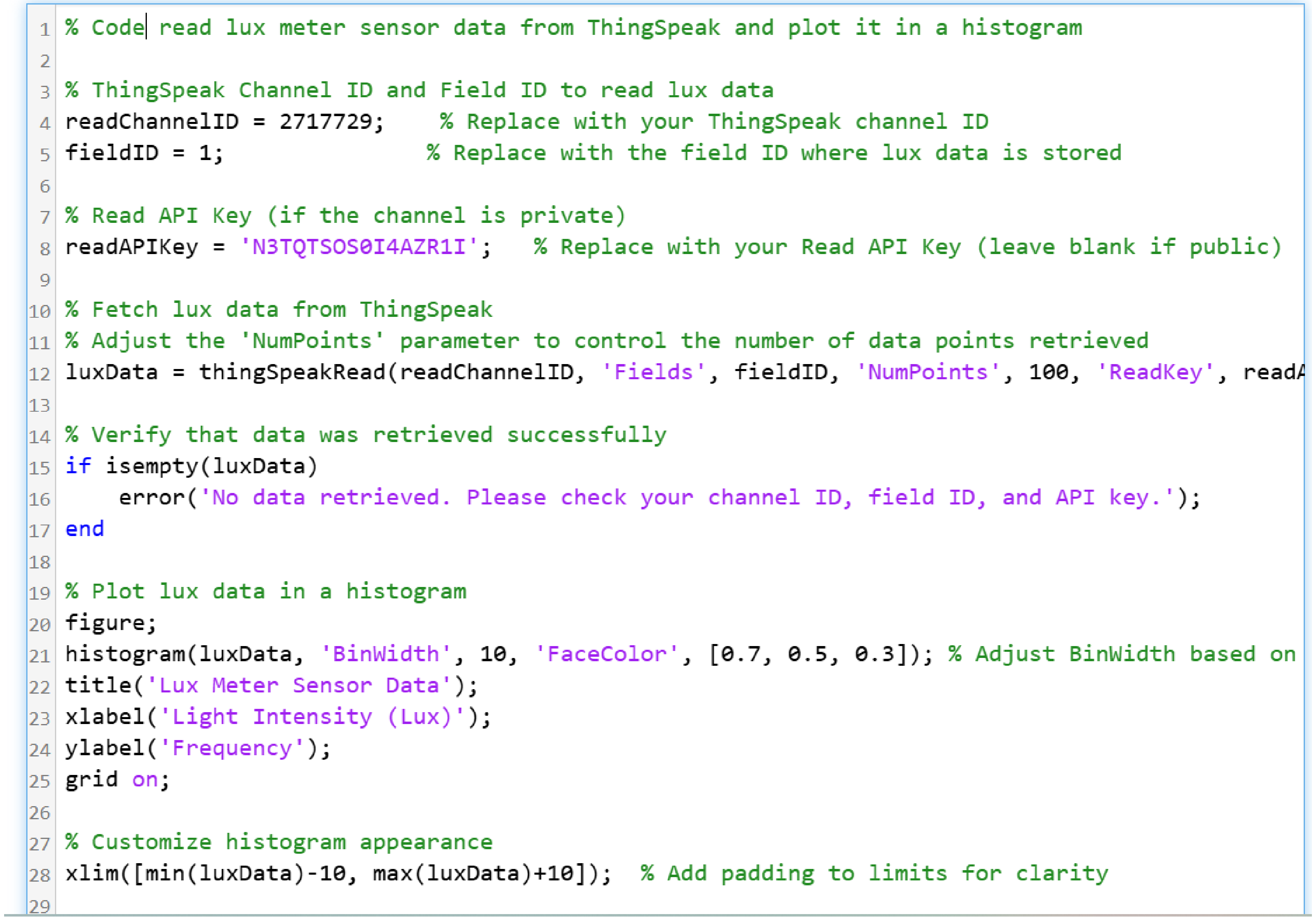


MATLAB Plot Output

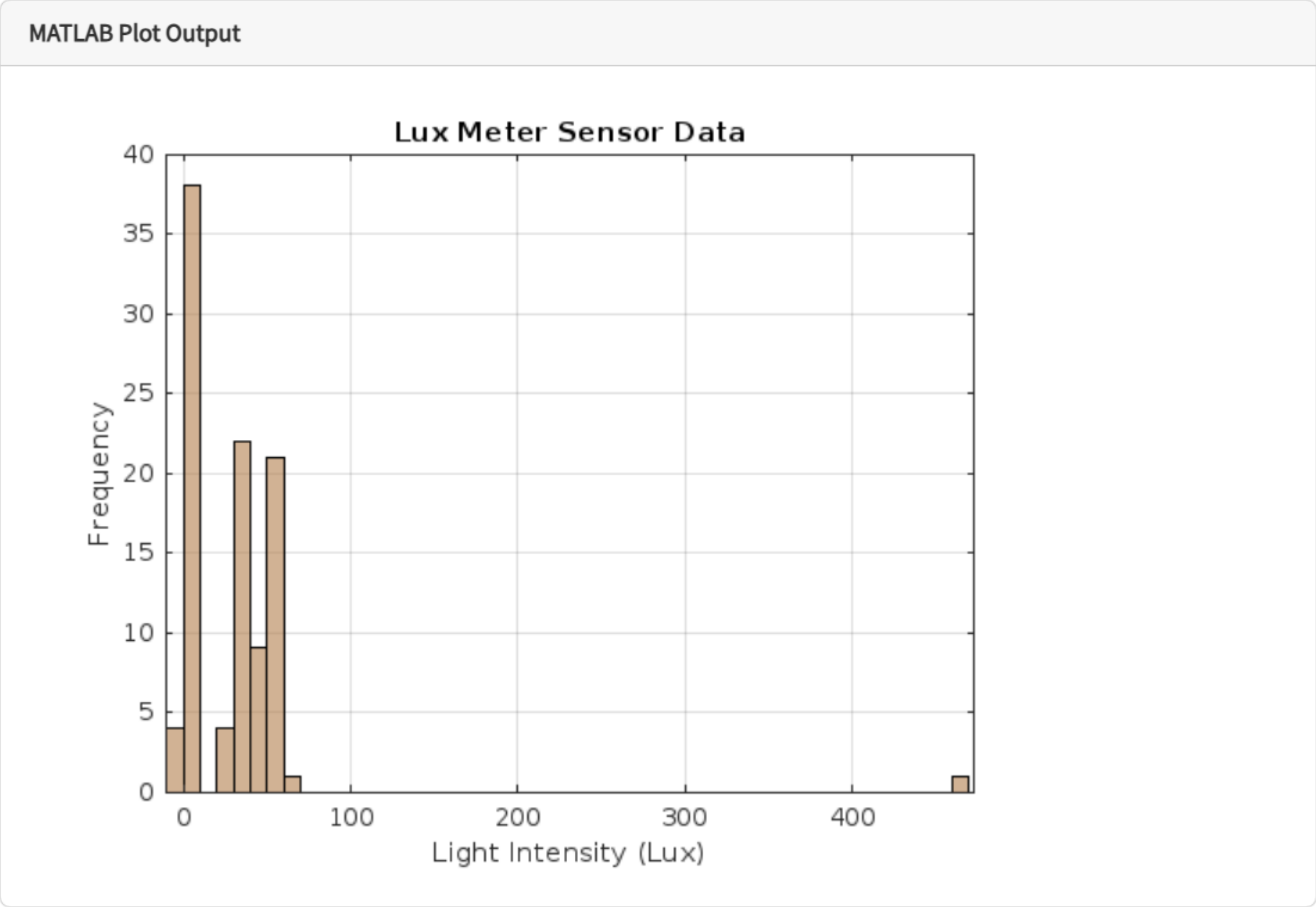


MATLAB Plot Output

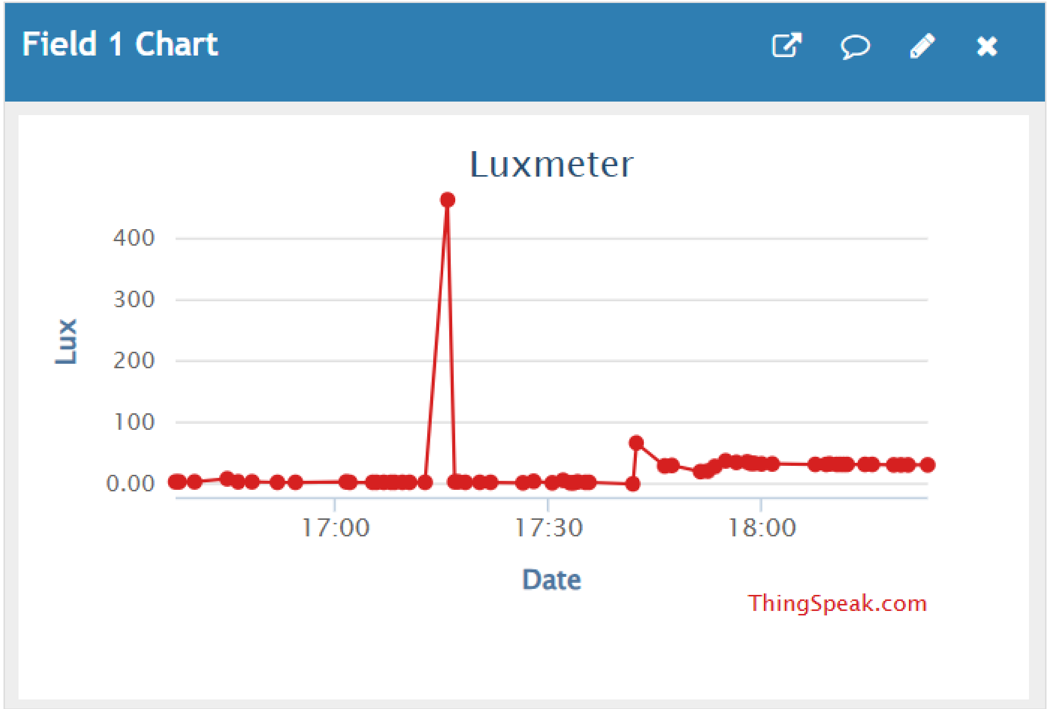
carenuty

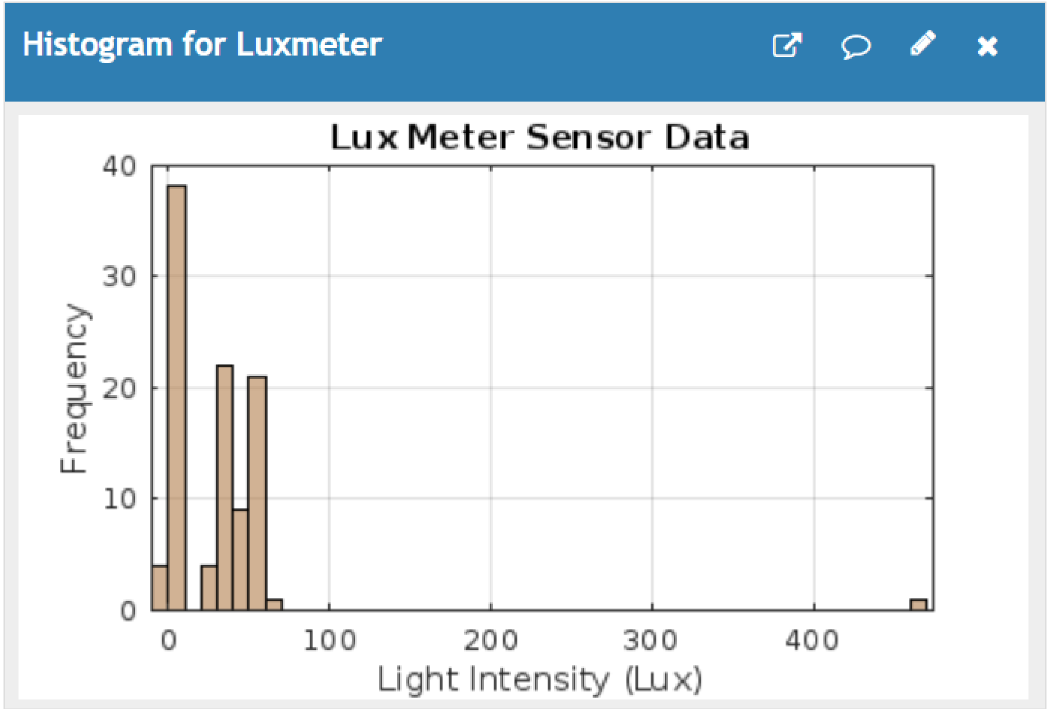


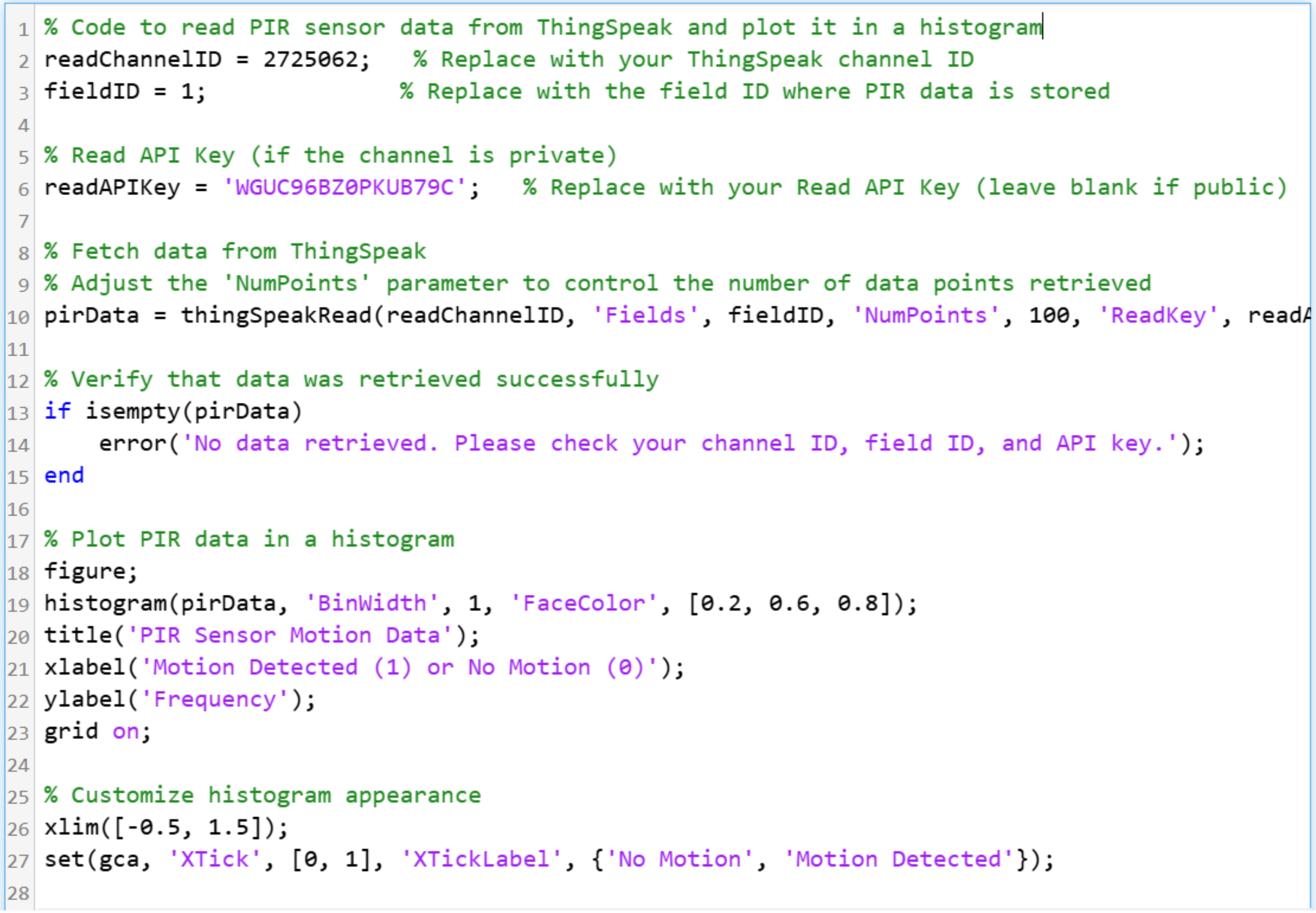
```
% Code read lux meter sensor data from ThingSpeak and plot it in a histogram

% ThingSpeak Channel ID and Field ID to read lux data
readChannelID = 2717729;    % Replace with your ThingSpeak channel ID
fieldID = 1;               % Replace with the field ID where lux data is stored

% Read API Key (if the channel is private)
readAPIKey = 'N3TQTSOS0I4AZR1I';   % Replace with your Read API Key (leave blank if public)

% Fetch lux data from ThingSpeak
% Adjust the 'NumPoints' parameter to control the number of data points retrieved
luxData = thingSpeakRead(readChannelID, 'Fields', fieldID, 'NumPoints', 100, 'ReadKey', readA

% Verify that data was retrieved successfully
if isempty(luxData)
    error('No data retrieved. Please check your channel ID, field ID, and API key.');
end

% Plot lux data in a histogram
figure;
histogram(luxData, 'BinWidth', 10, 'FaceColor', [0.7, 0.5, 0.3]); % Adjust BinWidth based on
title('Lux Meter Sensor Data');
xlabel('Light Intensity (Lux)');
ylabel('Frequency');
grid on;

% Customize histogram appearance
xlim([min(luxData)-10, max(luxData)+10]);  % Add padding to limits for clarity

```



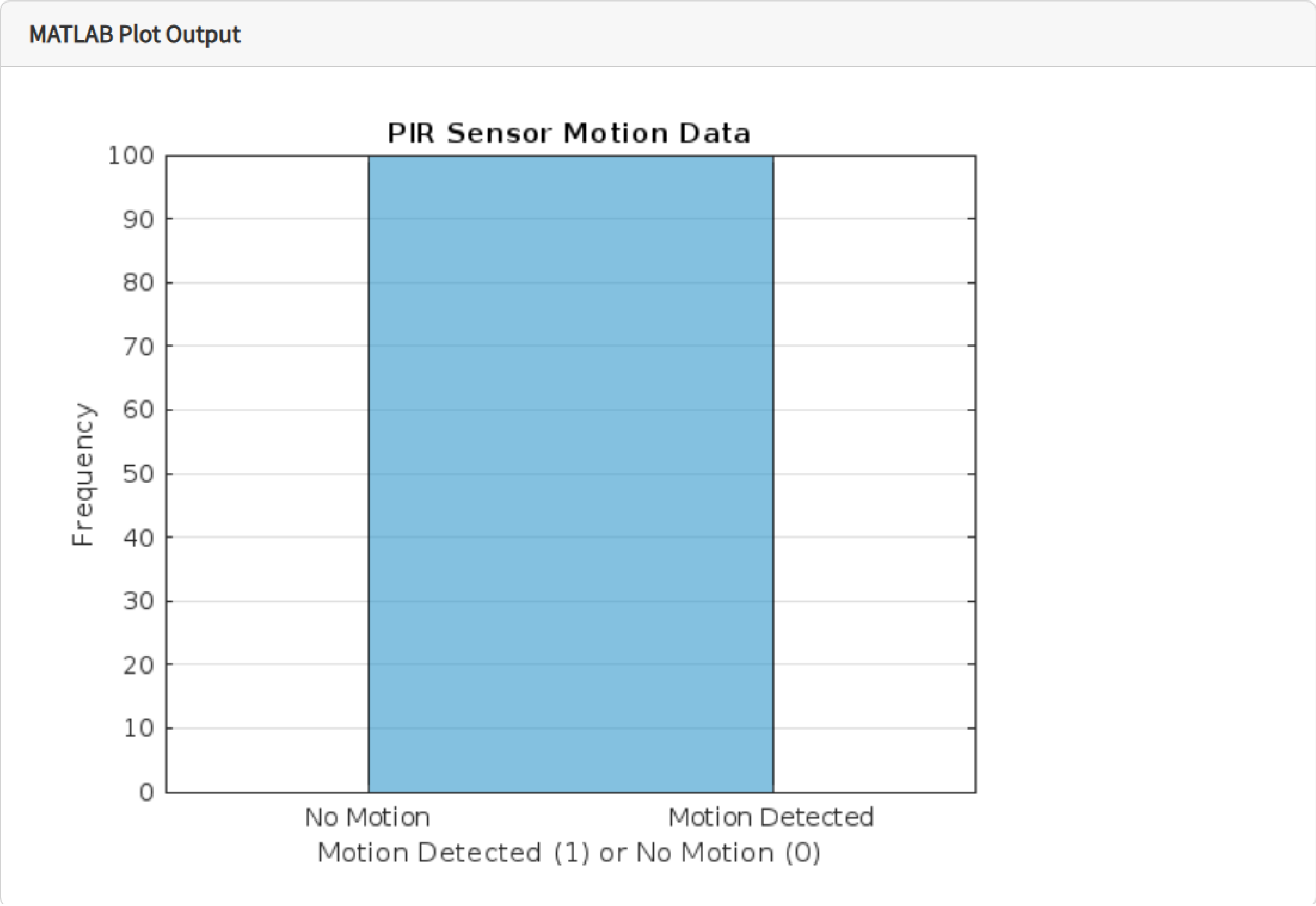
MATLAB Plot Output


Field 1 Chart
Luxmeter
Lux
400
300
200
100
0.00
17:00
17:30
18:00
Date
ThingSpeak.com

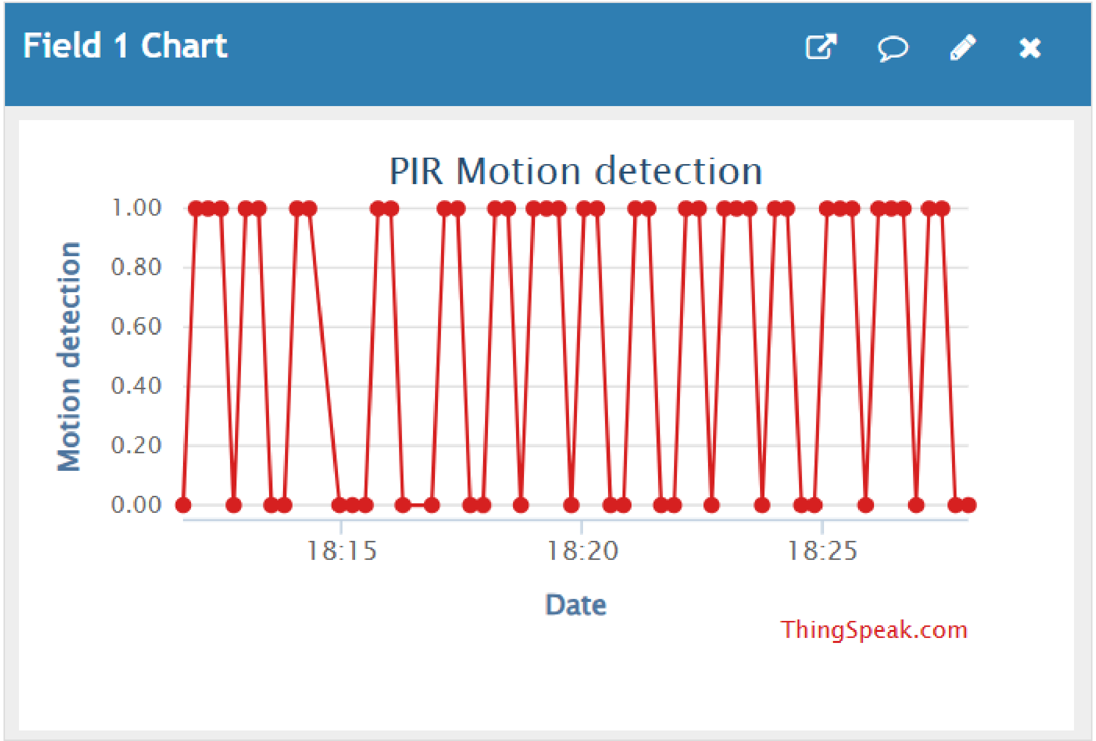

Histogram for Luxmeter
Lux Meter Sensor Data
Frequency
40
30
20
10
0
0
100
200
300
400
Light Intensity (Lux)

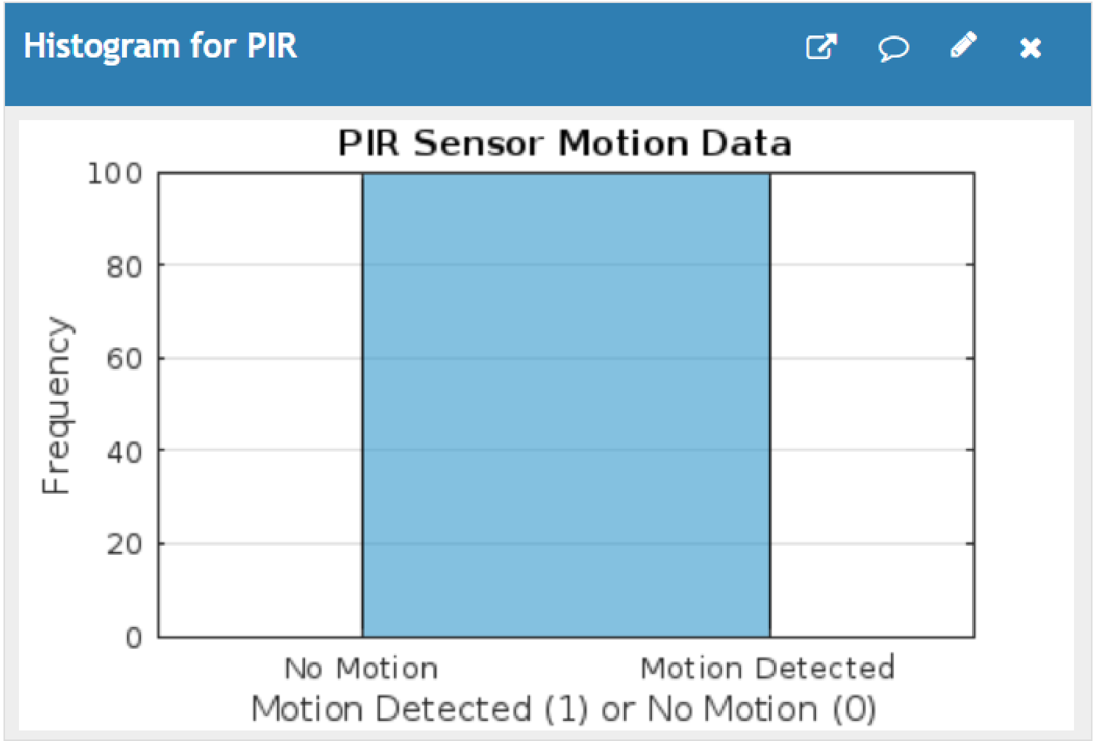


```
% Code to read PIR sensor data from ThingSpeak and plot it in a histogram
readChannelID = 2725062;   % Replace with your ThingSpeak channel ID
fieldID = 1;              % Replace with the field ID where PIR data is stored

% Read API Key (if the channel is private)
readAPIKey = 'WGUC96BZ0PKUB79C';   % Replace with your Read API Key (leave blank if public)

% Fetch data from ThingSpeak
% Adjust the 'NumPoints' parameter to control the number of data points retrieved
pirData = thingSpeakRead(readChannelID, 'Fields', fieldID, 'NumPoints', 100, 'ReadKey', readA

% Verify that data was retrieved successfully
if isempty(pirData)
    error('No data retrieved. Please check your channel ID, field ID, and API key.');
end

% Plot PIR data in a histogram
figure;
histogram(pirData, 'BinWidth', 1, 'FaceColor', [0.2, 0.6, 0.8]);
title('PIR Sensor Motion Data');
xlabel('Motion Detected (1) or No Motion (0)');
ylabel('Frequency');
grid on;

% Customize histogram appearance
xlim([-0.5, 1.5]);
set(gca, 'XTick', [0, 1], 'XTickLabel', {'No Motion', 'Motion Detected'});
```



MATLAB Plot Output


Field 1 Chart
PIR Motion detection
Motion detection
1.00
0.80
0.60
0.40
0.20
0.00
18:15
18:20
18:25
Date
ThingSpeak.com


Histogram for PIR
PIR Sensor Motion Data
Frequency
100
80
60
40
20
0
No Motion
Motion Detected
Motion Detected (1) or No Motion (0)

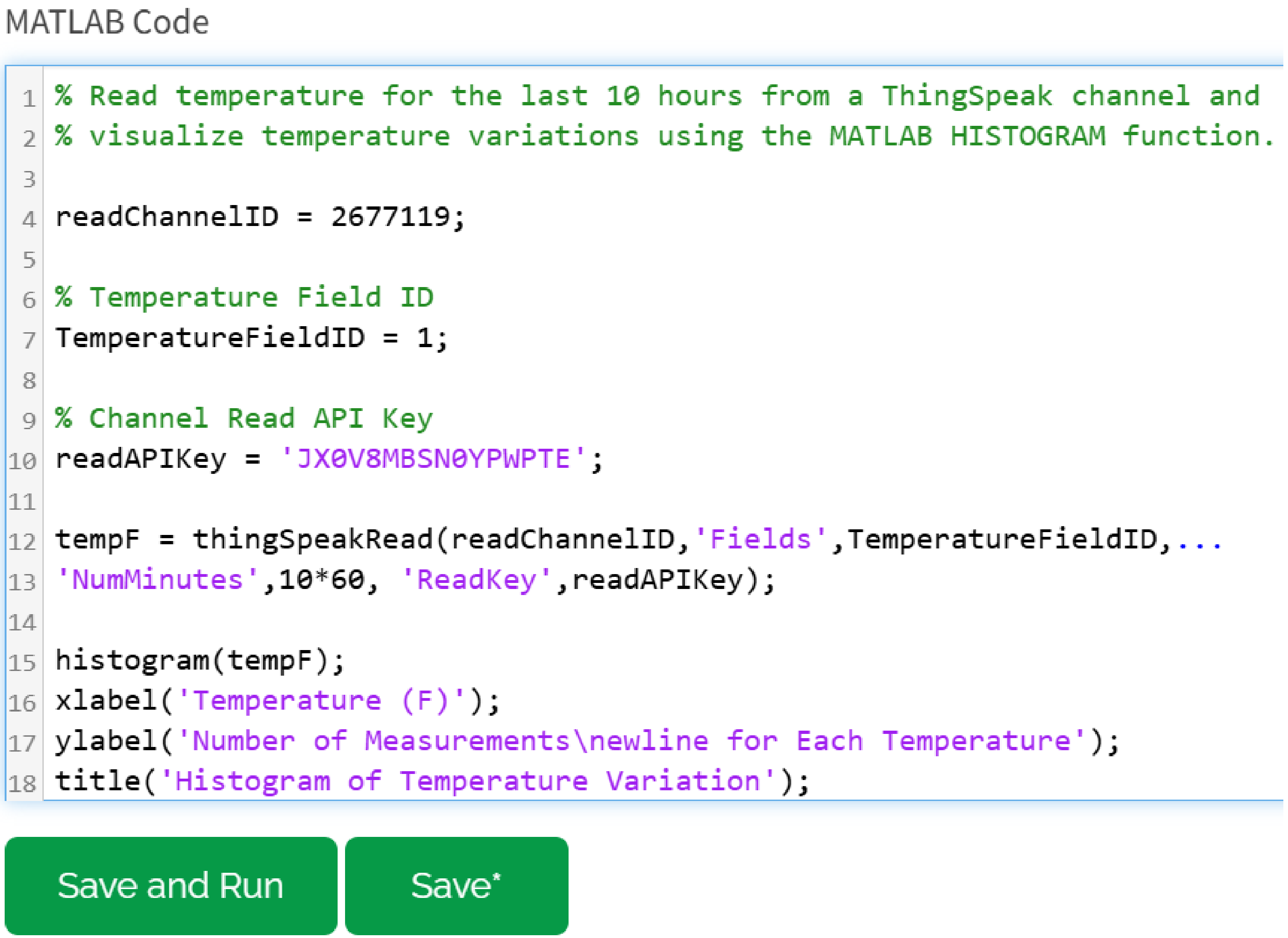
CONCLUSION

carenuty

- MATLAB offers a robust environment for data visualization, enabling the creation of a wide variety of graphs and plots, such as histograms, scatter plots, 3D surfaces, and heatmaps, which provide clear and detailed insights.

- While ThingSpeak's Field graphs are effective for basic real-time data monitoring, they are limited in customization and depth. By importing ThingSpeak data into MATLAB through channels, I am able to enhance visualizations to suit the dataset better, resulting in more tailored and informative views.

- Click create and write your MATLAB code (e.g., read and plot data), Save and Run.



MATLAB Code

```matlab
% Read temperature for the last 10 hours from a ThingSpeak channel and
% visualize temperature variations using the MATLAB HISTOGRAM function.

readChannelID = 2677119;

% Temperature Field ID
TemperatureFieldID = 1;

% Channel Read API Key
readAPIKey = 'JX0V8MBSN0YPWPTE';

tempF = thingSpeakRead(readChannelID,'Fields',TemperatureFieldID,...
'NumMinutes',10*60, 'ReadKey',readAPIKey);

histogram(tempF);
xlabel('Temperature (F)');
ylabel('Number of Measurements\newline for Each Temperature');
title('Histogram of Temperature Variation');
```

NEXT STEPS

carenuty

- Next, I will be exploring MATLAB’s analytical capabilities, allowing me to perform deeper data analysis on the collected information.
- This shift to MATLAB analysis promises to unlock further insights, turning raw data into actionable information with advanced functions and calculations.
- This workflow, from data collection on ThingSpeak to enhanced visualization and analysis in MATLAB, showcases a powerful and complementary use of both platforms.

carenuity

THE END