

Project 16: Capstone Project

Author : Nilesh Tiwari cs17b022@iittp.ac.in

It is a temperature sensor setup. It employs machine learning and anomaly methods to maintain and detect temperature changes and sends an alert message over email(using Mailgun email services) about the problem.

Hardware required

- The Bolt Wifi module
- 3 female to male wire
- Temperature Sensor: LM35 sensor
- USB cable

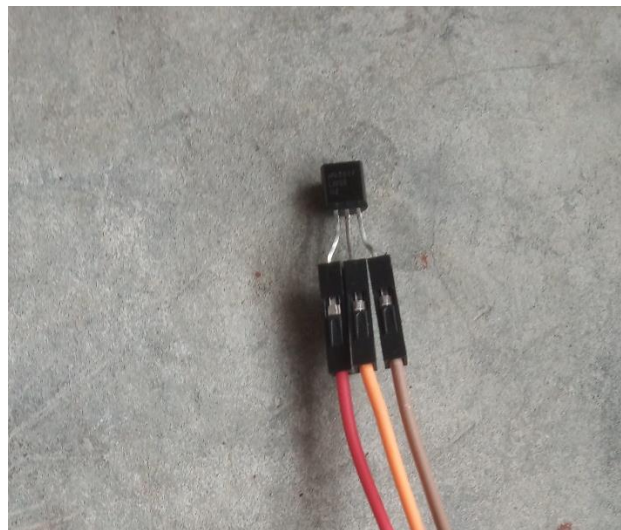
The project progression in order of the objectives of the Capstone project are as follows.

A. Build the circuit for temperature monitoring system, using the Bolt and LM35 sensor.

Connections

Step 1: Hold the sensor in a manner such that you can read LM35 written on it.

Step 2: In this position, identify the pins of the sensor as VCC, Output and Gnd from your left to right.

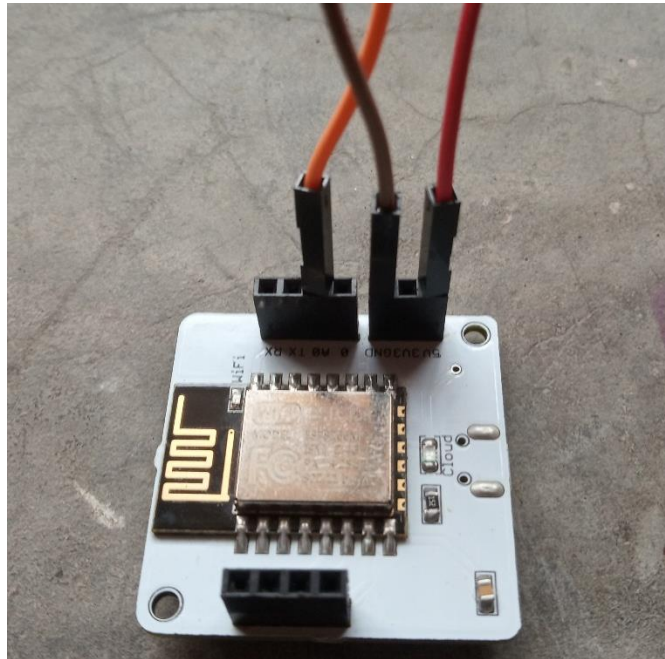


In the above image, VCC is connected to the red wire, Output is connected to the orange wire and Gnd is connected to the brown wire.

Step 3: Using male to female wire connect the 3 pins of the LM35 to the Bolt Wifi Module as follows:

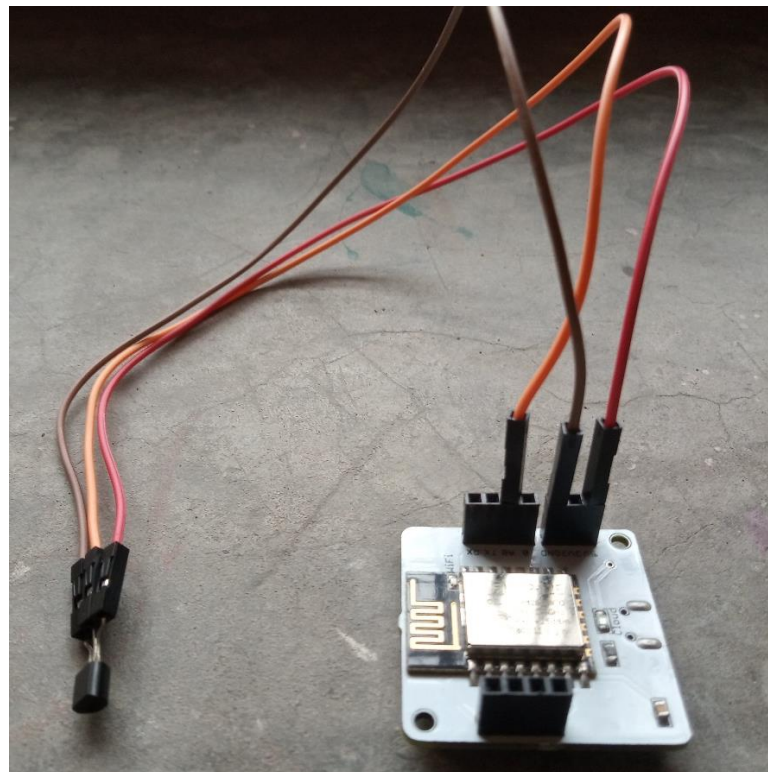
1. VCC pin of the LM35 connects to 5v of the Bolt Wifi module.

2. Output pin of the LM35 connects to A0 (Analog input pin) of the Bolt Wifi module.
3. Gnd pin of the LM35 connects to the Gnd.

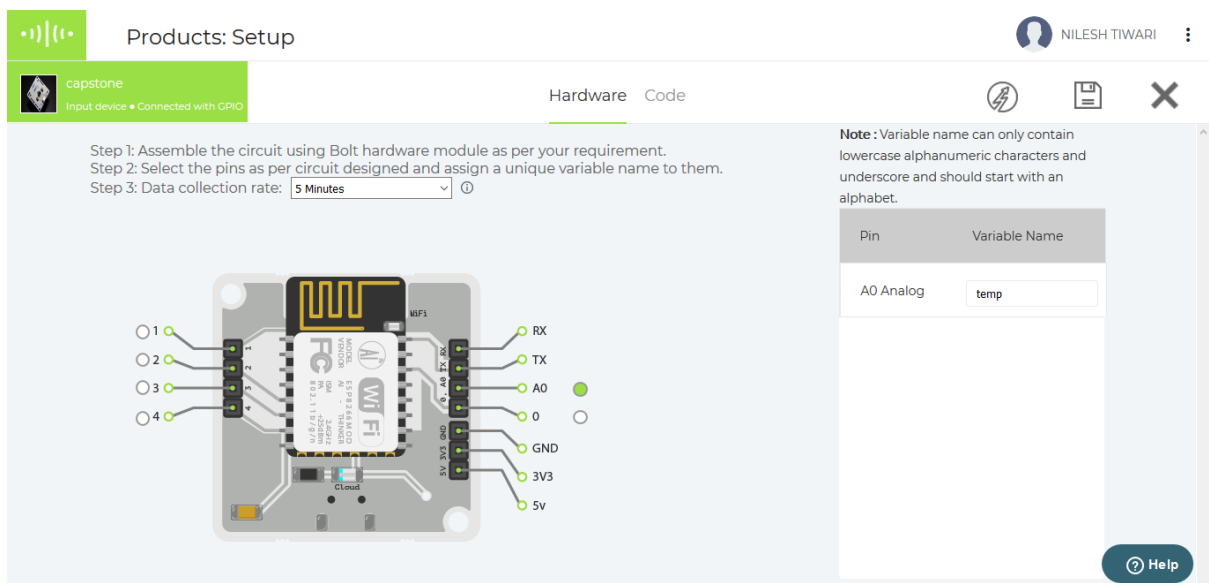
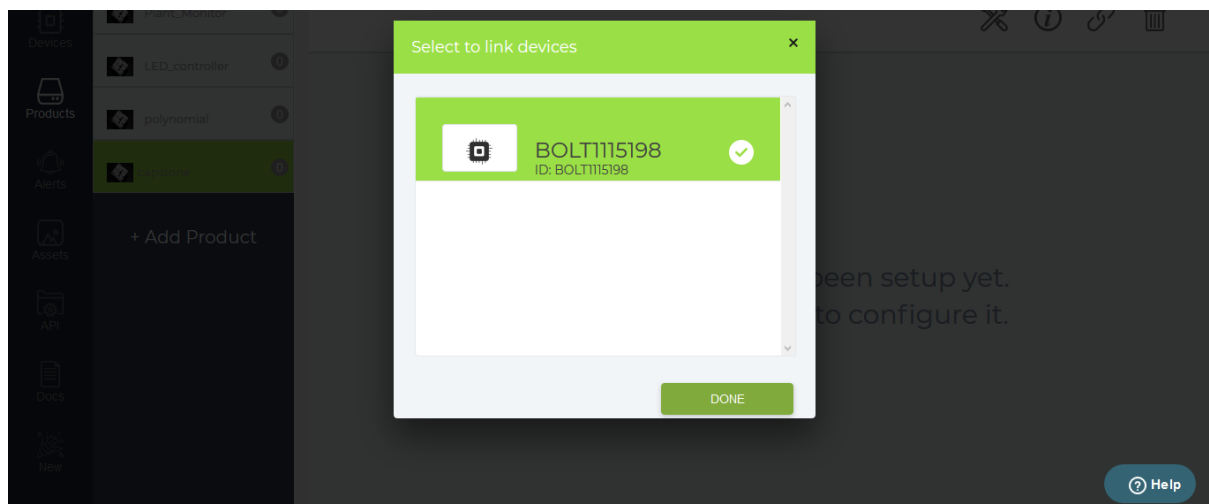
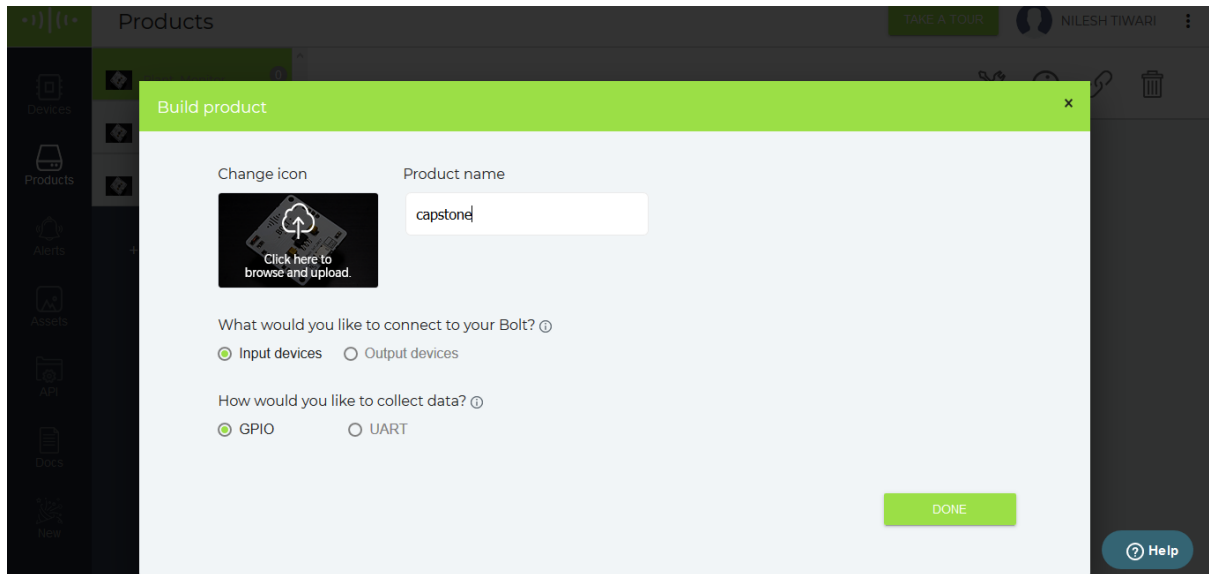


Circuit Connection

The final circuit should look like the image below:



B. Create a product on the Bolt Cloud, to monitor the data from the LM35, and link it to your Bolt.



C. Write the product code, required to run the polynomial regression algorithm on the data sent by the Bolt.

Device • Connected with CPIC

Write your code in the code window below.

polyfit

js

```
1 /*start typing your code here */
2 setChartLibrary('google-chart');
3 setChartTitle('Polynomial Regression');
4 setChartType('predictionGraph');
5 setAxisName('time_stamp','temp');
6 mul(0.0977);
7 plotChart('time_stamp','temp');
```

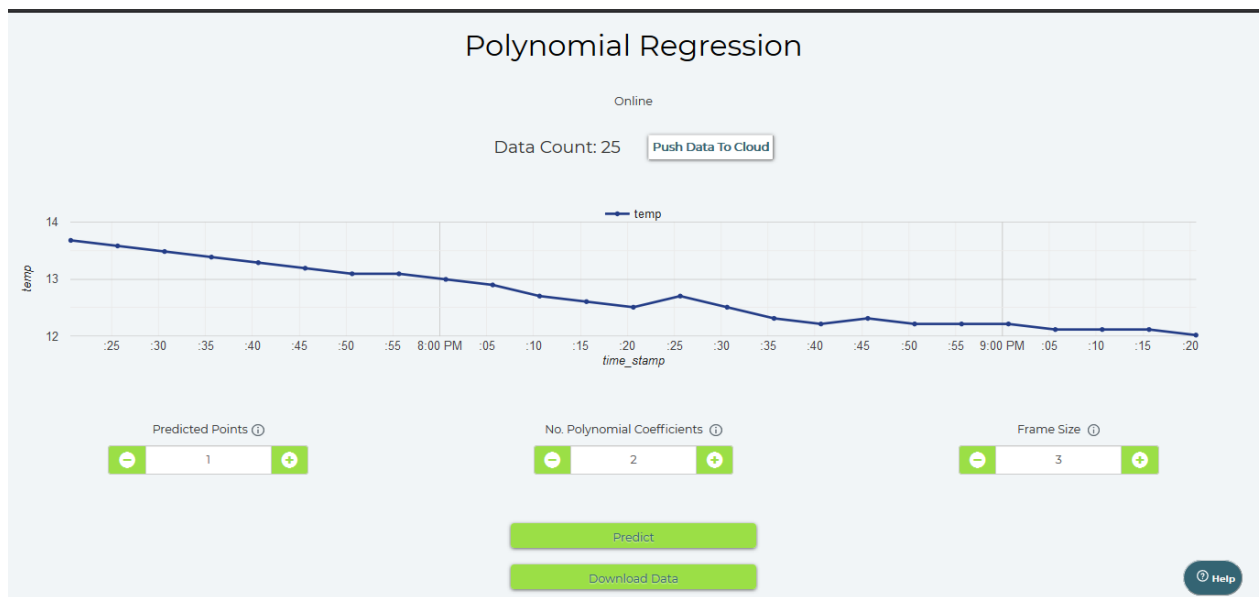
[Click here](#) to access the Documentation

D. Keep the temperature monitoring circuit inside your fridge with the door of the fridge closed, and let the system record the temperature readings for about 2 hours.



The temperature monitoring circuit kept in the fridge.

E. Using the reading that you received in the 2 hours, set boundaries for the temperature within the fridge.



The temperature reading received in the 2 hours

The boundaries are set to 12 degree Celsius and 13.7 degree Celsius because these are the minimum and maximum temperatures inside the fridge respectively as observed from the above chart between temperature and time.

F. Write a python code which will fetch the temperature data, every 10 seconds, and send out an email alert, if the temperature goes beyond the temperature thresholds you decided on in objective "E".

```
capstone_temp.py x
1 import email_conf
2 from boltiot import Email, Bolt
3 import json, time
4
5 min_limit = 123 #12 degree celsius
6 max_limit = 140 #13.7 degree celsius
7
8 mybolt = Bolt(email_conf.API_KEY, email_conf.DEVICE_ID)
9 mailer = Email(email_conf.MAILGUN_API_KEY, email_conf.SANDBOX_URL, email_conf.SENDER_EMAIL, email_conf.RECIPIENT_EMAIL)
10
11 while True:
12     print ("Reading sensor value")
13     response = mybolt.analogRead('A0')
14     data = json.loads(response)
15     print ("Sensor value is: " + str(data['value']))
16     try:
17         sensor_value = int(data['value'])
18         if sensor_value > max_limit or sensor_value < min_limit:
19             print("Making request to Mailgun to send an email")
20             response = mailer.send_email("Alert", "The Current temperature sensor value is " + str(sensor_value))
21             response_text = json.loads(response.text)
22             print("Response received from Mailgun is: " + str(response_text['message']))
23     except Exception as e:
24         print ("Error occured: Below are the details")
25         print (e)
26     time.sleep(10)
```

The python code for the objective F

```

nilesh@nilesh-HP-Laptop-15g-br0xx:/media/nilesh/Windows/Users/NILESH TIWARI/Documents/Bolt IoT/IoT_Projects$ python3 capstone_temp.py
Reading sensor value
Sensor value is: 134
Reading sensor value
Sensor value is: 135
Reading sensor value
Sensor value is: 134
Reading sensor value
Sensor value is: 134
Reading sensor value
Sensor value is: 132
Reading sensor value
Sensor value is: 134
Reading sensor value
Sensor value is: 134
Reading sensor value
Sensor value is: 134
Reading sensor value
Sensor value is: 132
Reading sensor value
Sensor value is: 157
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 154
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 147
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 142
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 140
Reading sensor value

```


The output of the program in execution.

```


Reading sensor value
Sensor value is: 134
Reading sensor value
Sensor value is: 134
Reading sensor value
Sensor value is: 132
Reading sensor value
Sensor value is: 157
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 154
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 147
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 142
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 140
Reading sensor value
Sensor value is: 139
Reading sensor value
Sensor value is: 138
Reading sensor value
Sensor value is: 136
Reading sensor value
Sensor value is: 135
Reading sensor value
Sensor value is: 136
Reading sensor value
Sensor value is: 135

```

The output of the program in execution.

 **test@sandboxe6a...** 1:39 AM
to me ▾


The Current temperature sensor value is 157

 **test@sandboxe6a...** 1:40 AM
to me ▾

The Current temperature sensor value is 154

 **test@sandboxe6a...** 1:40 AM
to me ▾

The Current temperature sensor value is 147

 **test@sandboxe6a...** 1:40 AM
to me ▾

The Current temperature sensor value is 142

The emails received when the temperature went beyond the temperature thresholds

G. Modify the python code, to also do a Z-score analysis and print the line “Someone has opened the fridge door” when an anomaly is detected.

Here, `conf.FRAME_SIZE = 10` and `conf.MUL_FACTOR = 2`.


```

capstone_temp.py x capstone_temp_anomaly.py x
1 import email_conf, conf
2 from boltiot import Email, Bolt
3 import json, time, math, statistics
4
5 def compute_bounds(history_data, frame_size, factor):
6     if len(history_data) < frame_size :
7         return None
8
9     if len(history_data) > frame_size :
10        del history_data[0:len(history_data)-frame_size]
11
12        Mn = statistics.mean(history_data)
13        Variance = 0
14        for data in history_data :
15            Variance += math.pow((data-Mn),2)
16        Zn = factor * math.sqrt(Variance / frame_size)
17        High_bound = history_data[frame_size-1] + Zn
18        Low_bound = history_data[frame_size-1] - Zn
19        return [High_bound, Low_bound]
20
21 min_limit = 123 #12 degree celsius
22 max_limit = 140 #13.7 degree celsius
23
24 mybolt = Bolt(email_conf.API_KEY, email_conf.DEVICE_ID)
25 mailer = Email(email_conf.MAILGUN_API_KEY, email_conf.SANDBOX_URL, email_conf.SENDER_EMAIL, email_conf.RECIPIENT_EMAIL)
26 history_data = []
27
28 while True:
29     print ("Reading sensor value")
30     response = mybolt.analogRead('A0')
31     data = json.loads(response)
32     print ("Sensor value is: " + str(data['value']))
33     try:
34         sensor_value = int(data['value'])
35         bound = compute_bounds(history_data, conf.FRAME_SIZE, conf.MUL_FACTOR)
36         if sensor_value > max_limit or sensor_value < min_limit:
37             print("Making request to Mailgun to send an email")
38             response = mailer.send_email("Alert", "The Current temperature sensor value is " +str(sensor_value))
39             response_text = json.loads(response.text)
40             print("Response received from Mailgun is: " + str(response_text['message']))
41         if not bound :
42             history_data.append(sensor_value)
43             time.sleep(10)
44             continue
45         elif sensor_value > bound[0] or sensor_value < bound[1]:
46             print ("Someone has opened the fridge door.")
47             history_data.append(sensor_value);
48     except Exception as e:
49         print ("Error occured: Below are the details")
50         print (e)
51         time.sleep(10)

```

The python code for the objective G

```

nilesh@nilesh-HP-Laptop-15g-br0xx:/media/nilesh/Windows/Users/NILESH TIWARI/Documents/Bolt IoT/IoT_Projects$ python3 capstone_temp_anomaly.py
Reading sensor value
Sensor value is: 130
Reading sensor value
Sensor value is: 127
Reading sensor value
Sensor value is: 128
Reading sensor value
Sensor value is: 127
Reading sensor value
Sensor value is: 129
Reading sensor value
Sensor value is: 127
Reading sensor value
Sensor value is: 125
Reading sensor value
Sensor value is: 126
Reading sensor value
Sensor value is: 124
Reading sensor value
Sensor value is: 125
Reading sensor value
Sensor value is: 124
Reading sensor value
Sensor value is: 124
Reading sensor value
Sensor value is: 167
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Someone has opened the fridge door.
Reading sensor value
Sensor value is: 159
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 153
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.

```

The output of the program in execution.


```

Sensor value is: 127
Reading sensor value
Sensor value is: 125
Reading sensor value
Sensor value is: 126
Reading sensor value
Sensor value is: 124
Reading sensor value
Sensor value is: 125
Reading sensor value
Sensor value is: 124
Reading sensor value
Sensor value is: 124
Reading sensor value
Sensor value is: 167
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Someone has opened the fridge door.
Reading sensor value
Sensor value is: 159
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 153
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 141
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 133
Reading sensor value
Sensor value is: 133
Reading sensor value
Sensor value is: 132

```

The output of the program in execution.

 **test@sandboxe6a...** 1:50 AM
to me ▾

The Current temperature sensor value is 167

 **test@sandboxe6a...** 1:50 AM
to me ▾

The Current temperature sensor value is 159

 **test@sandboxe6a...** 1:51 AM
to me ▾

The Current temperature sensor value is 153

 **test@sandboxe6a...** 1:51 AM
to me ▾

The Current temperature sensor value is 141

The emails received when the temperature went beyond the temperature thresholds

H. Tune the Z-score analysis code, such that, it detects an anomaly when someone opens the door of the fridge.

After tuning, `conf.FRAME_SIZE = 10` and `conf.MUL_FACTOR = 3`.

```
nilesh@nilesh-HP-Laptop-15g-br0xx: /media/nilesh/Windows/Users/NILESH TIWARI/Documents/Bolt IoT/IoT_Projects$ python3 capstone_temp_anomaly.py
Reading sensor value
Sensor value is: 128
Reading sensor value
Sensor value is: 125
Reading sensor value
Sensor value is: 126
Reading sensor value
Sensor value is: 123
Reading sensor value
Sensor value is: 124
Reading sensor value
Sensor value is: 124
Reading sensor value
Sensor value is: 124
Reading sensor value
Sensor value is: 128
Reading sensor value
Sensor value is: 130
Reading sensor value
Sensor value is: 126
Reading sensor value
Sensor value is: 127
Reading sensor value
Sensor value is: 124
Reading sensor value
Sensor value is: 125
Reading sensor value
Sensor value is: 150
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Someone has opened the fridge door.
Reading sensor value
Sensor value is: 144
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 138
```

The output of the program in execution.


```
Sensor value is: 124
Reading sensor value
Sensor value is: 124
Reading sensor value
Sensor value is: 128
Reading sensor value
Sensor value is: 130
Reading sensor value
Sensor value is: 126
Reading sensor value
Sensor value is: 127
Reading sensor value
Sensor value is: 124
Reading sensor value
Sensor value is: 125
Reading sensor value
Sensor value is: 150
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Someone has opened the fridge door.
Reading sensor value
Sensor value is: 144
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Reading sensor value
Sensor value is: 138
Reading sensor value
Sensor value is: 135
Reading sensor value
Sensor value is: 133
Reading sensor value
Sensor value is: 131
Reading sensor value
Sensor value is: 129
Reading sensor value
Sensor value is: 128
```

The output of the program in execution.

 **test@sandboxe6a...** 2:10 AM
to me ▾



The Current temperature sensor value is 150

 **test@sandboxe6a...** 2:10 AM
to me ▾



The Current temperature sensor value is 144

The emails received when the temperature went beyond the temperature thresholds