IOT DOMAIN ANALYST LAB 5

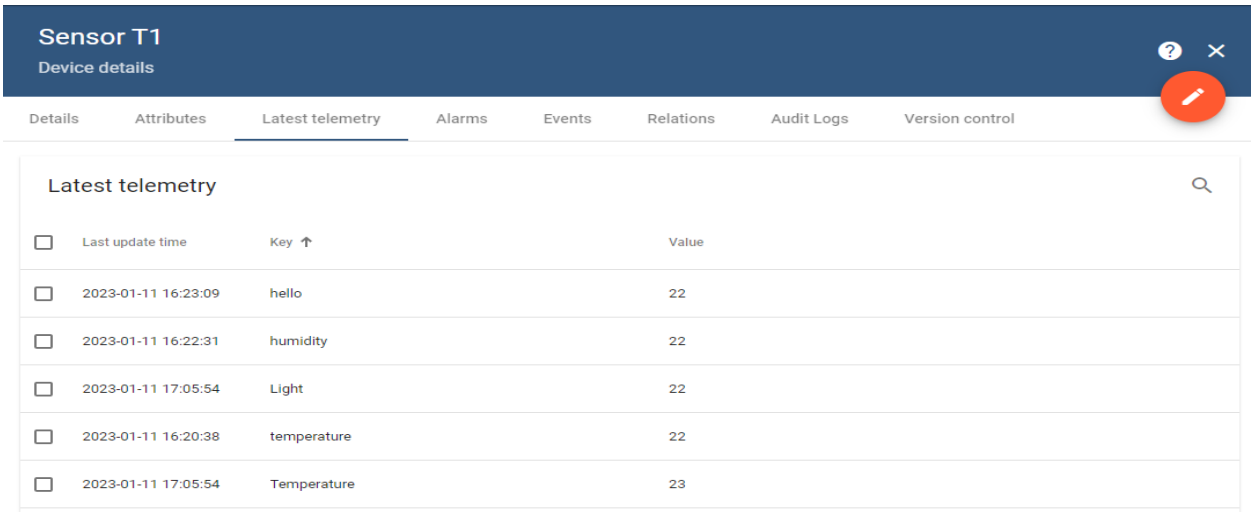
NAME : Vishal Kumar Mahatha

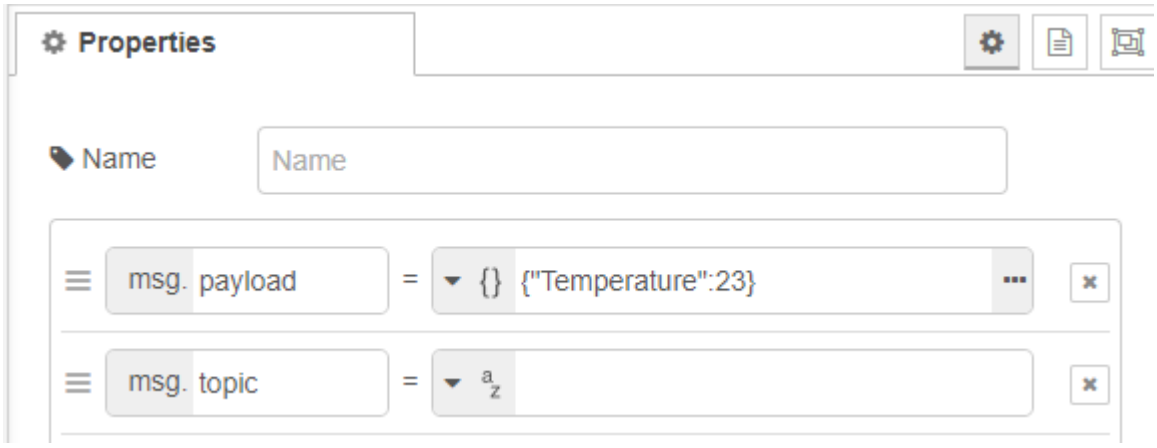
REG NO : 20BRS1168

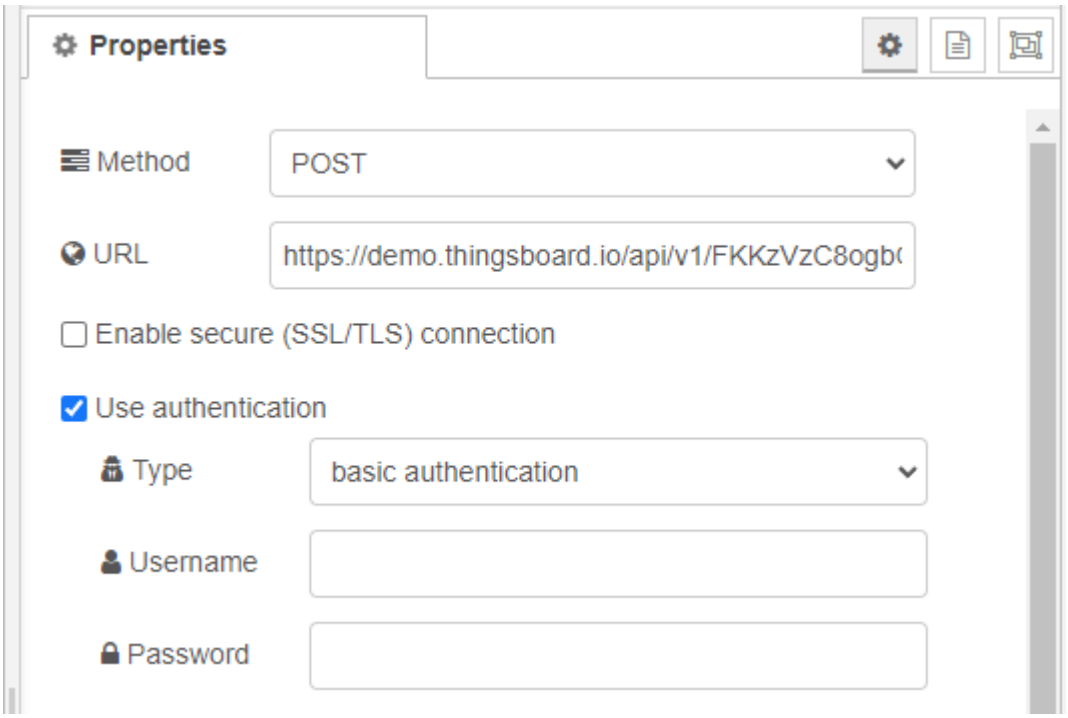
PRACTICE -1

Aim : print the temperature data in thingsboard from node red

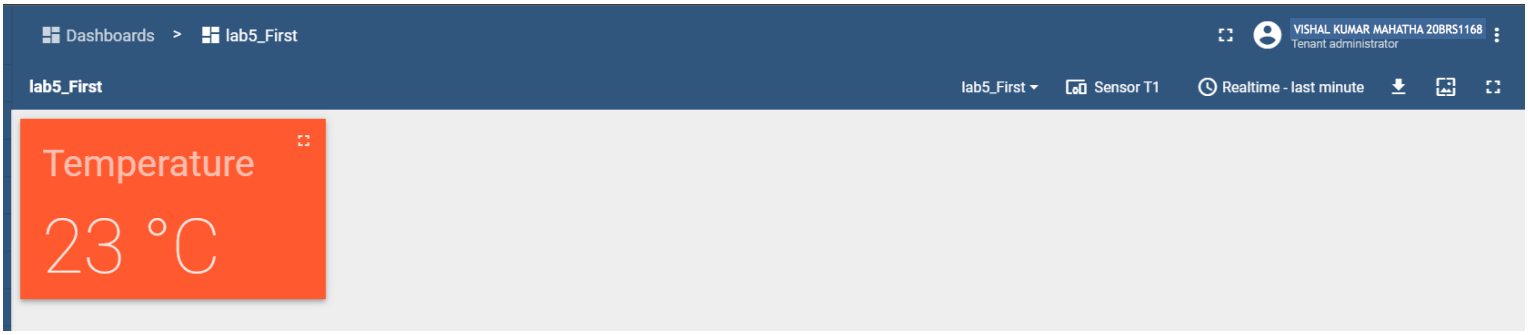
Components : Node-red, Thingsboard






Properties
Method
POST
URL
https://demo.thingsboard.io/api/v1/FKKzVzC8ogb(
Enable secure (SSL/TLS) connection
Use authentication
Type
basic authentication
Username
Password

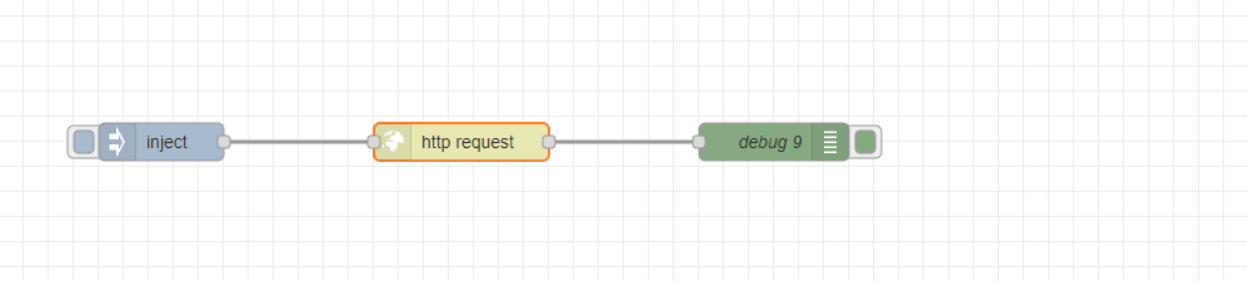
OUTPUT :

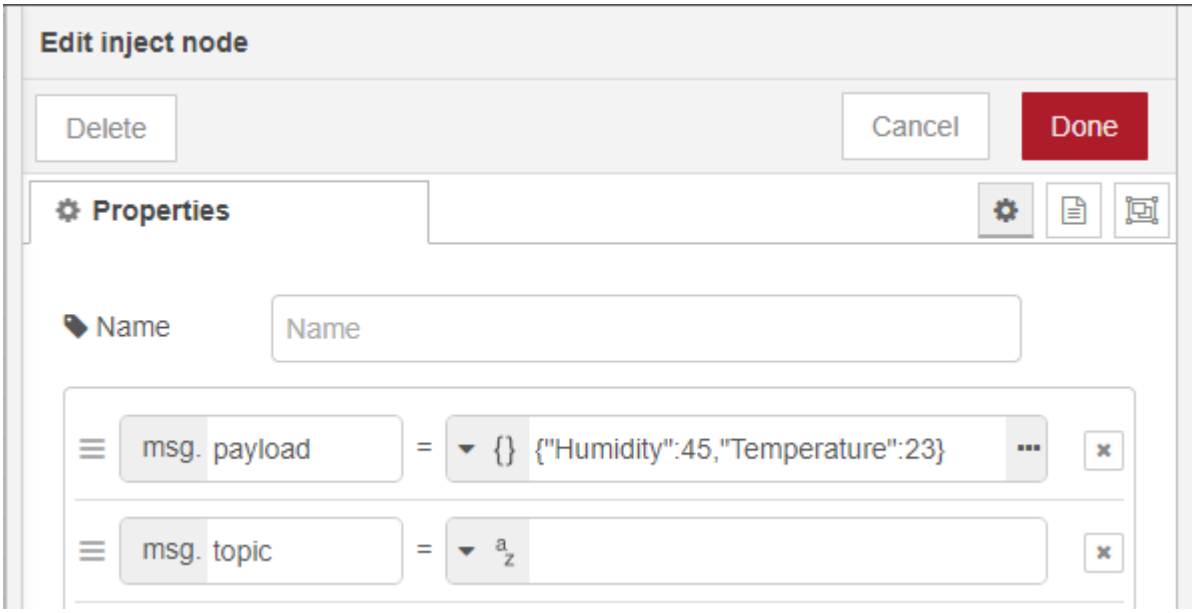

Dashboards
lab5_First
VISHAL KUMAR MAHATHA 20BRS1168
Tenant administrator
lab5_First
lab5_First
Sensor T1
Realtime - last minute
Temperature
23 °C

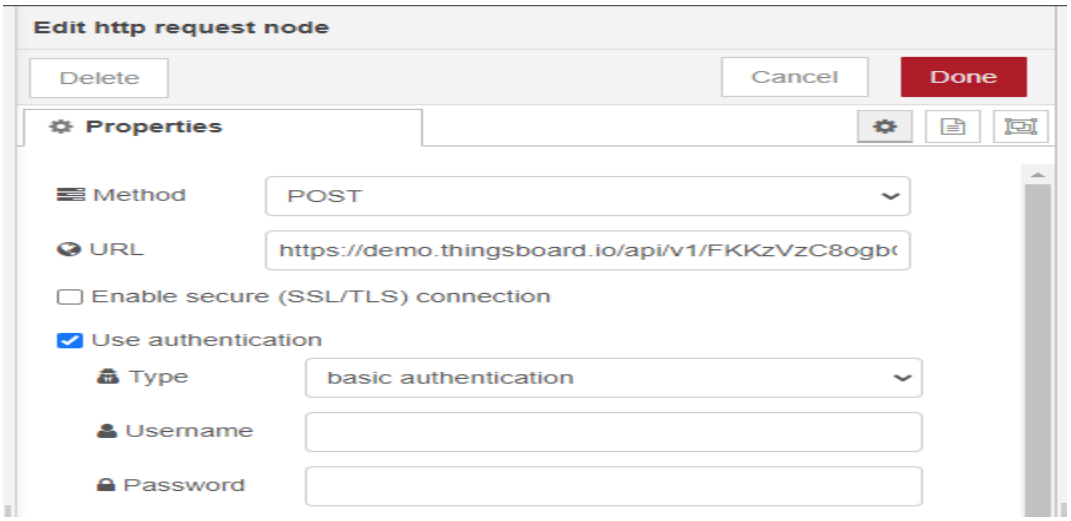
PRACTICE - 2

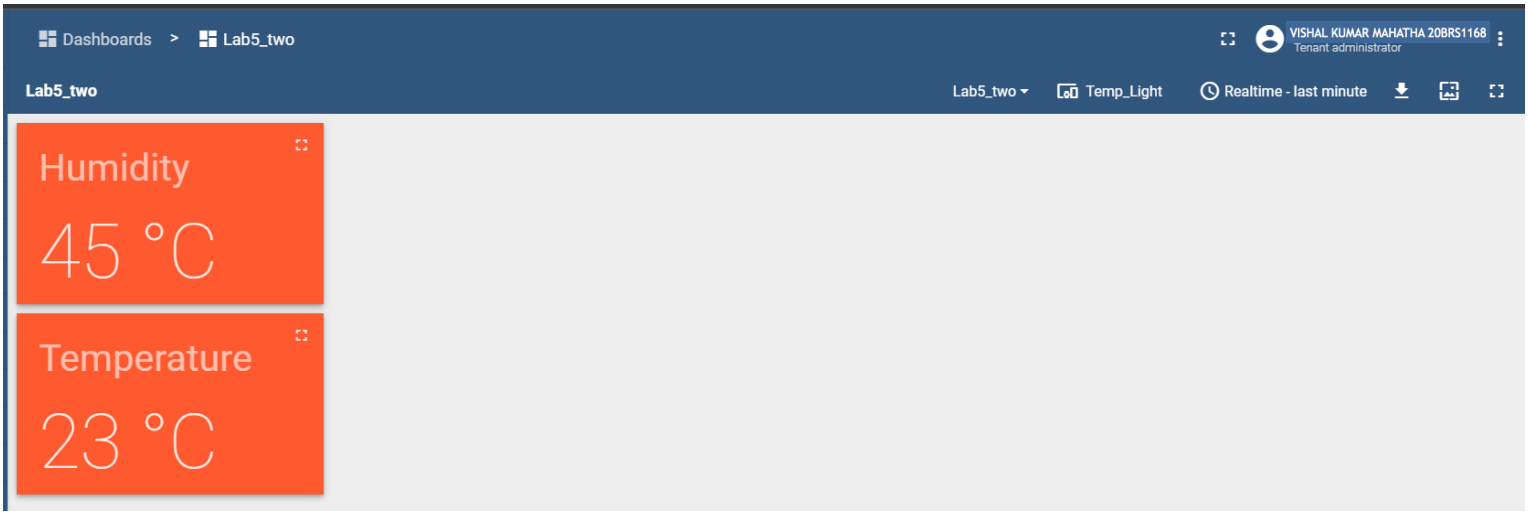
Aim : print the temperature and humidity data in Arduino

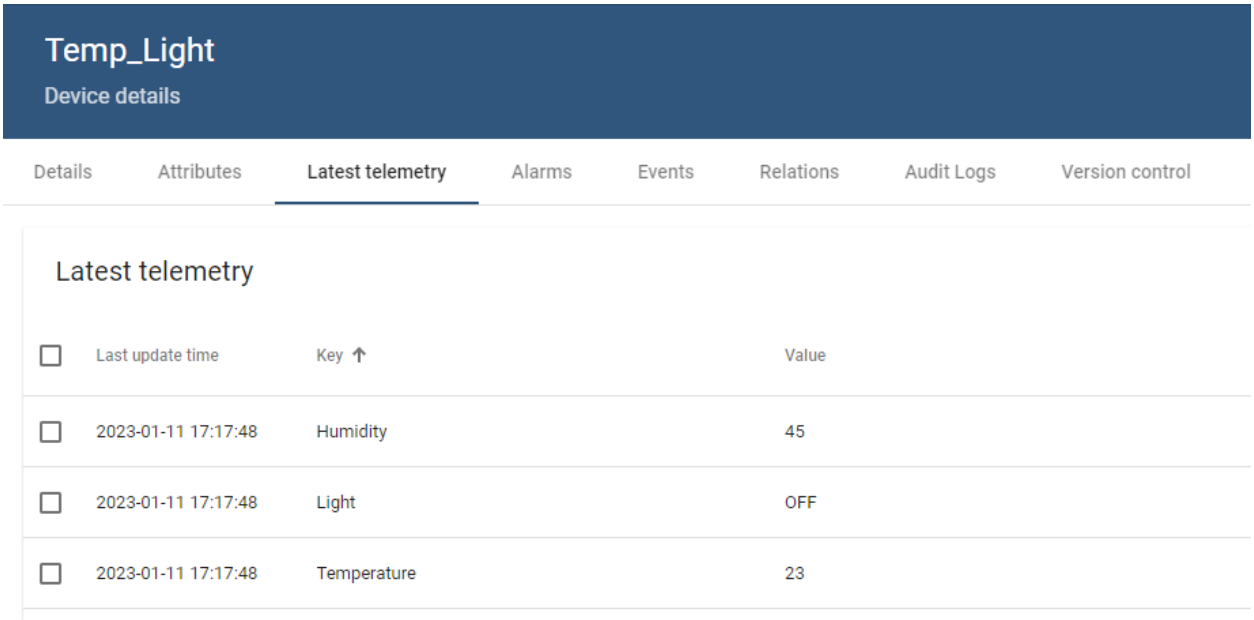
Components : node-red , thingsboard.






Edit http request node
Delete
Cancel
Done
Properties
Method
POST
URL
https://demo.thingsboard.io/api/v1/FKKzVzC8ogb
Enable secure (SSL/TLS) connection
Use authentication
Type
basic authentication
Username
Password


Dashboards
Lab5_two
VISHAL KUMAR MAHATHA 20BRS1168
Tenant administrator
Lab5_two
Lab5_two
Temp_Light
Realtime - last minute
Humidity
45 °C
Temperature
23 °C

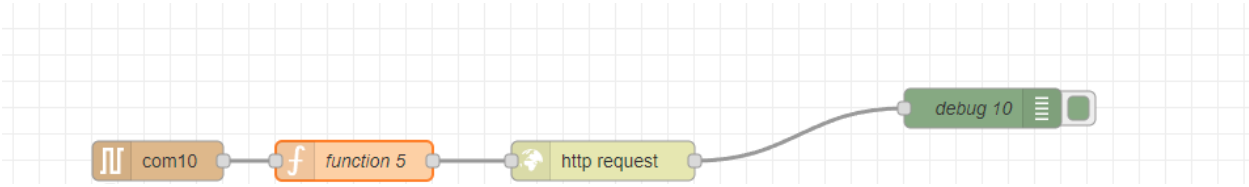


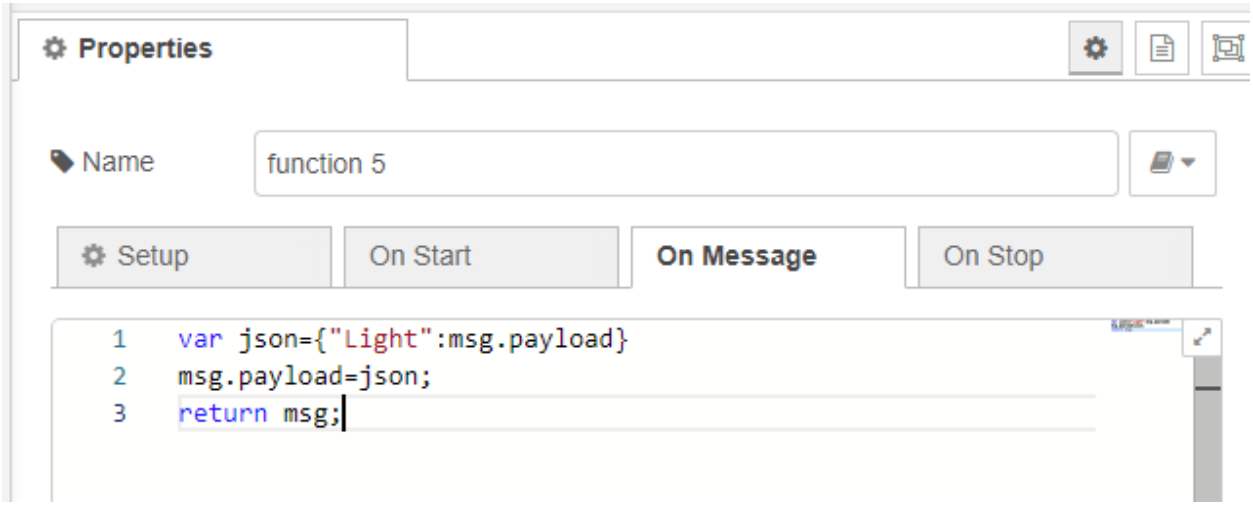
PRACTICE - 3

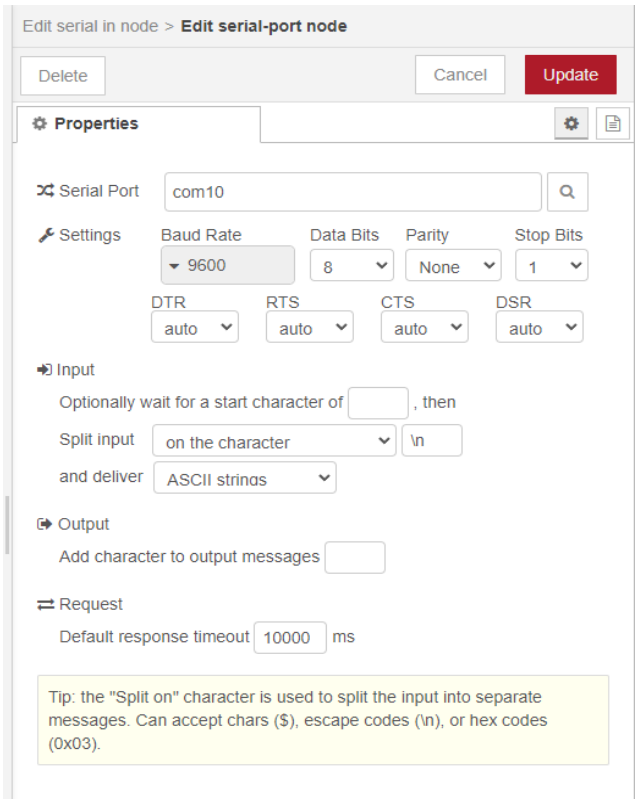
AIM : Simulate the Led data in Thingsboard

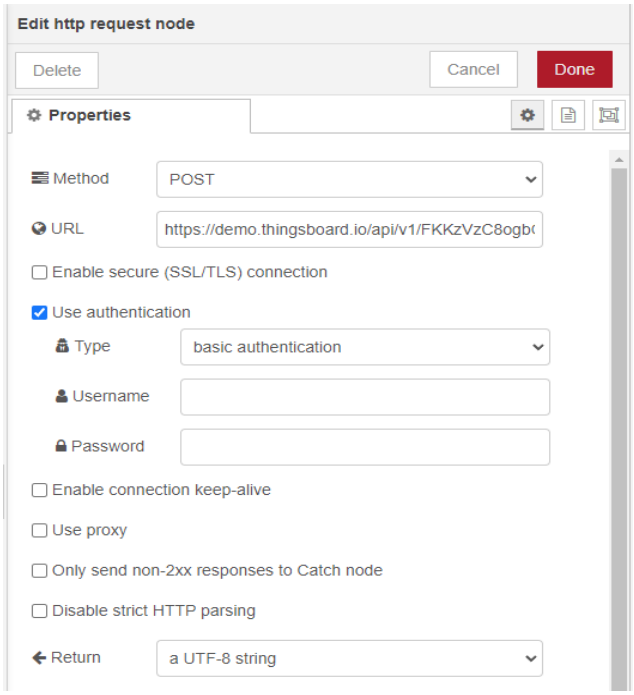
Components : Led , Arduino , Thingsboard , Node-Red

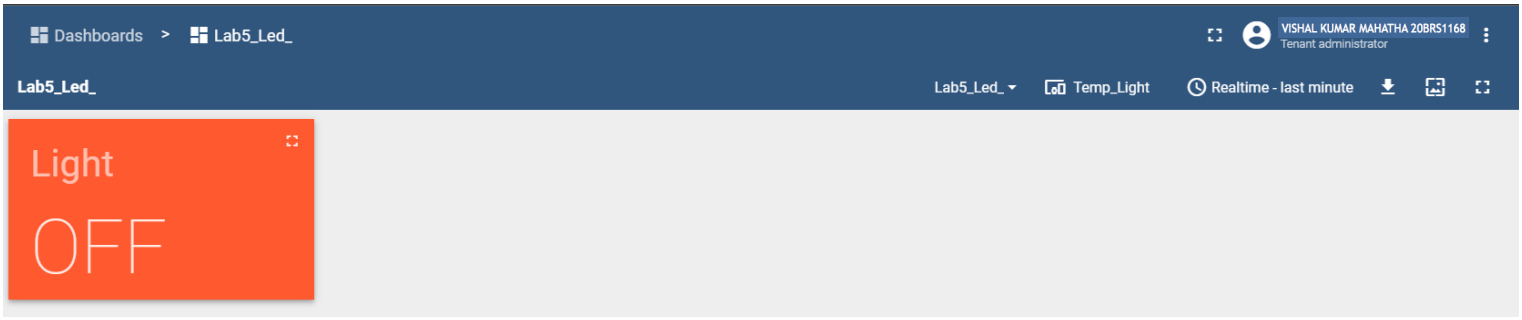
```ino
const int led = 13;
void setup() {
  // put your setup code here, to run once:
  pinMode(led,OUTPUT);
  Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(led,HIGH);
  delay(2000);
  Serial.println("ON");
  digitalWrite(led,LOW);
  delay(2000);
    Serial.println("OFF");

}
```



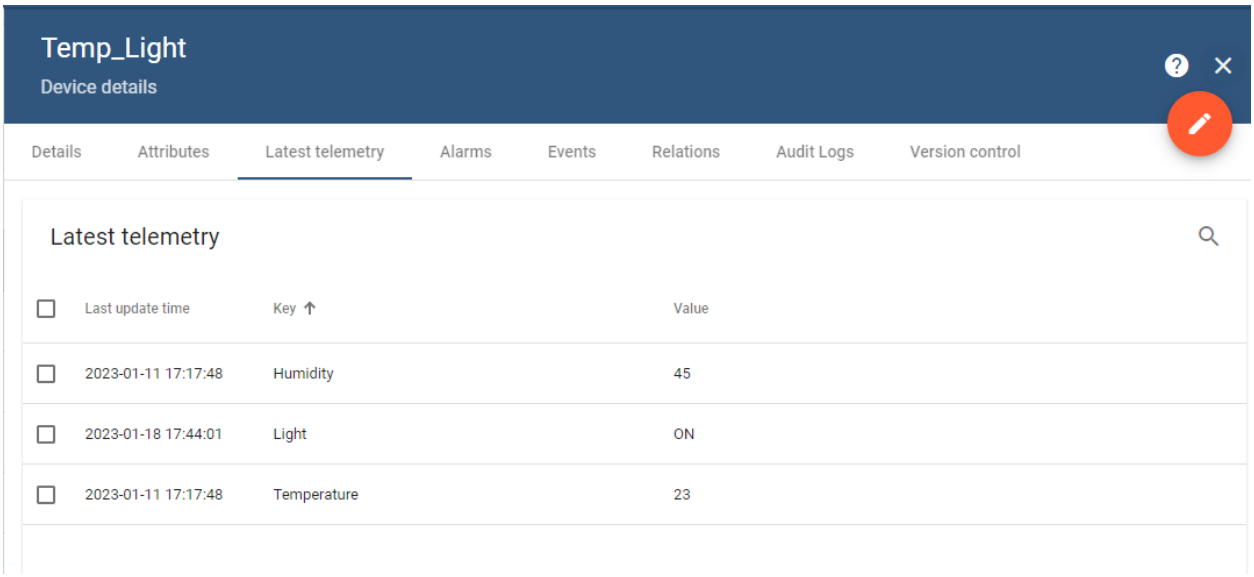


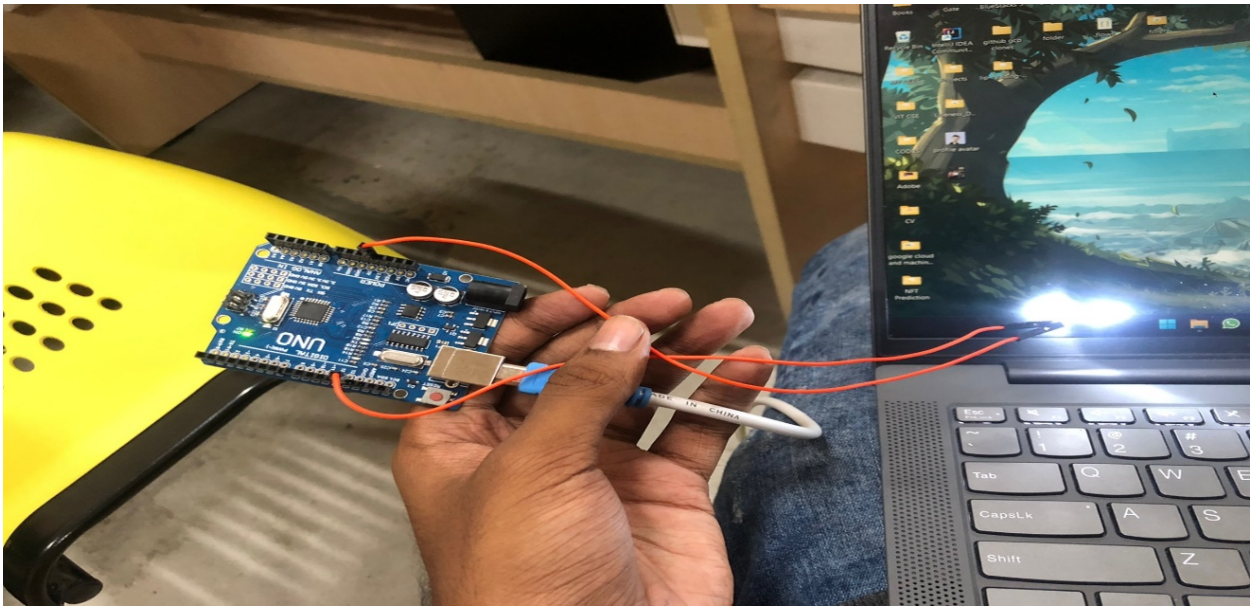

Edit serial in node > Edit serial-port node
Delete
Cancel
Update
Properties
Serial Port
com10
Settings
Baud Rate
9600
Data Bits
8
Parity
None
Stop Bits
1
DTR
auto
RTS
auto
CTS
auto
DSR
auto
Input
Optionally wait for a start character of
, then
Split input
on the character
\n
and deliver
ASCII strings
Output
Add character to output messages
Request
Default response timeout
10000
ms
Tip: the "Split on" character is used to split the input into separate messages. Can accept chars ($), escape codes (\n), or hex codes (0x03).


Edit http request node
Delete
Cancel
Done
Properties
Method
POST
URL
https://demo.thingsboard.io/api/v1/FKKzVzC8ogb
Enable secure (SSL/TLS) connection
Use authentication
Type
basic authentication
Username
Password
Enable connection keep-alive
Use proxy
Only send non-2xx responses to Catch node
Disable strict HTTP parsing
Return
a UTF-8 string


Dashboards
Lab5_Led_
VISHAL KUMAR MAHATHA 20BRS1168
Tenant administrator
Lab5_Led_
Lab5_Led_
Temp_Light
Realtime - last minute
Light
OFF

OUTPUT :


Temp_Light
Device details
Details
Attributes
Latest telemetry
Alarms
Events
Relations
Audit Logs
Version control
Latest telemetry
Last update time
Key
Value
2023-01-11 17:17:48
Humidity
45
2023-01-18 17:44:01
Light
ON
2023-01-11 17:17:48
Temperature
23



EXERCISE - 1

AIM : Create a smart home automation system that will consider Humidity (range between 20 to 100) and temperature (range of 20 to 100) values. Convert the values of temperature into degrees. These values are generated by the Arduino for every 2 seconds. When the temperature is > 30 degrees Switch ON the FAN otherwise OFF . Similarly if the humidity values are <20, the Light will ONotherwise OFF. Simulate the above scenario using things board and Arduino along with the timestamp.

PROCEDURE :

A smart home automation system that takes into account humidity and temperature values generated by an Arduino every 2 seconds could be implemented as follows:

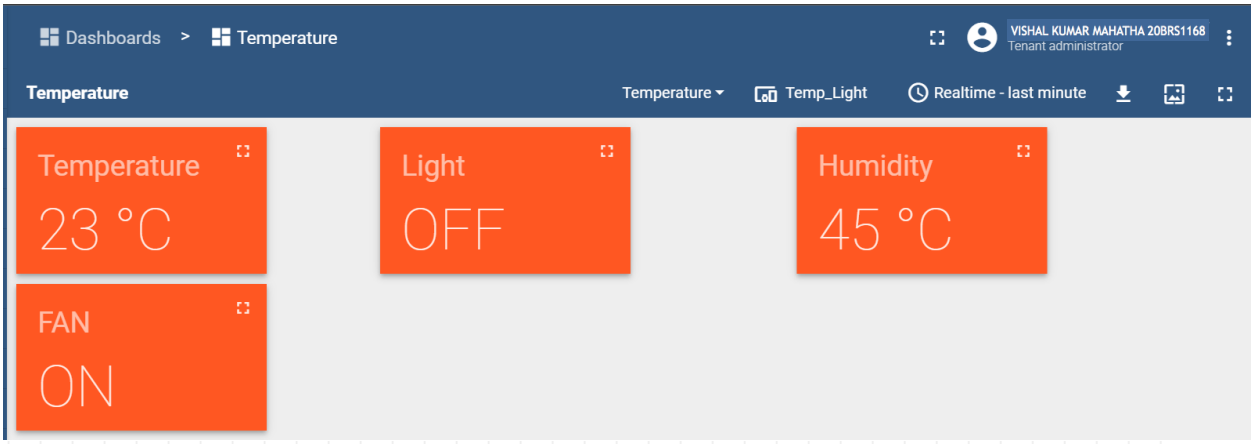
1. The Arduino would be connected to a humidity and temperature sensor, and programmed to read the values every 2 seconds and send them to a Thingsboard IoT platform.

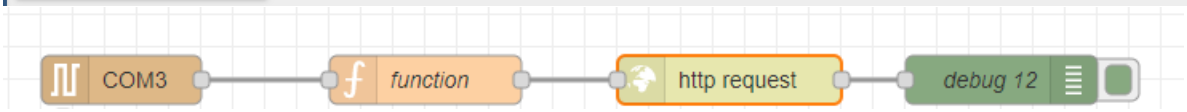
2. The Thingsboard platform would receive the humidity and temperature values from the Arduino and store them in a database, along with a timestamp.
3. The Thingsboard platform would also have a rule engine that would process the incoming values and control the FAN and Light accordingly. The rule engine would be programmed as follows:

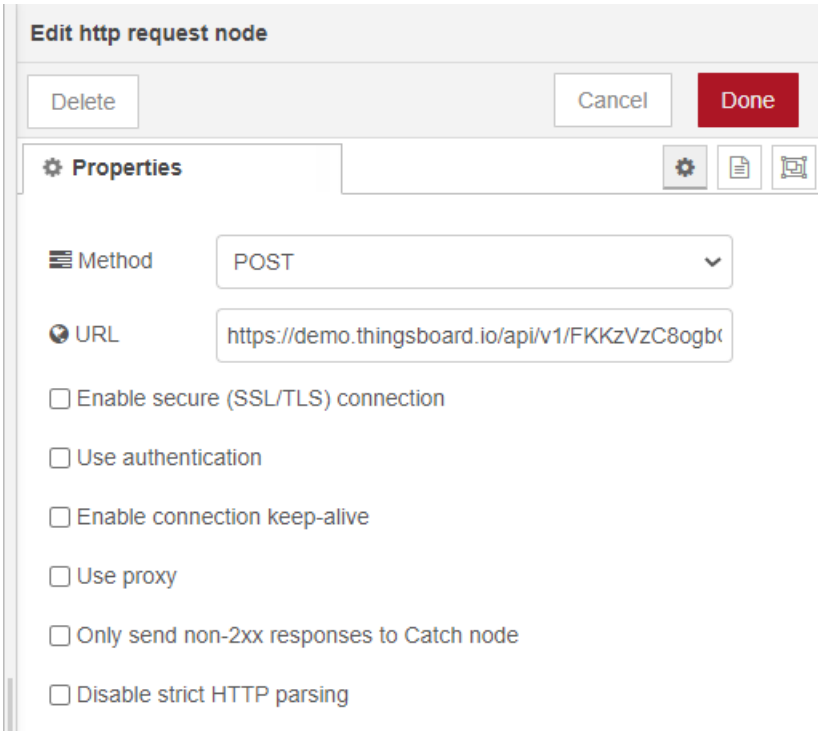
- If the temperature value is greater than 30 degrees, it would send a command to switch ON the FAN
- If the temperature value is less than 30 degrees, it would send a command to switch OFF the FAN
- If the humidity value is less than 20, it would send a command to switch ON the Light
- If the humidity value is greater than 20, it would send a command to switch OFF the Light

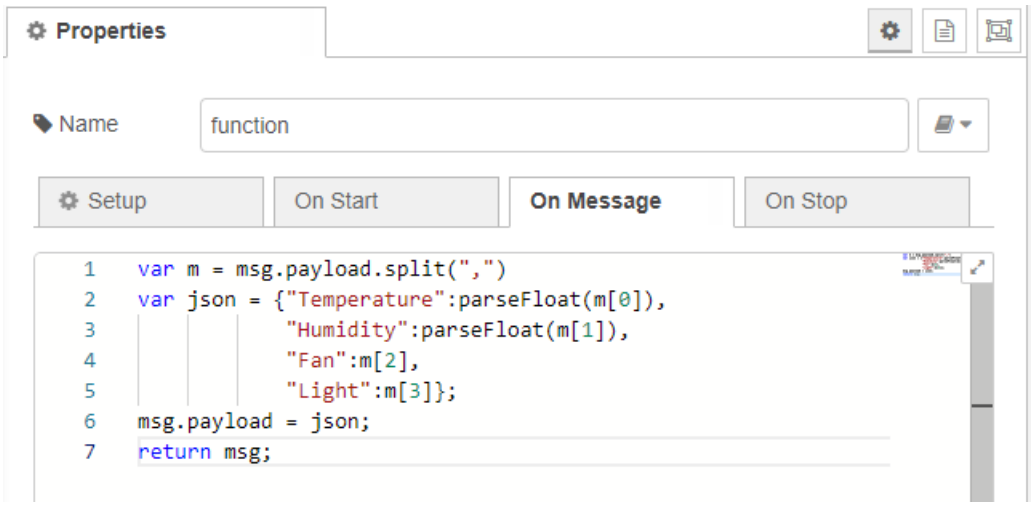
4. The Thingsboard platform would also provide a user interface to display the temperature and humidity values with their timestamps, as well as the status of the FAN and Light.
5. The Arduino would also be connected to actuators such as relays to control the FAN and Light based on the commands received from Thingsboard.
6. The system would be designed in a way to be able to handle multiple devices, and more complex rules can be set up in the rules engine to control other devices as well.

```
#include <DHT.h>
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_TSL2561_U.h>
#include <L298N.h>

#define DHTPIN 2
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

#define LIGHT_PIN 9

int temperature;
int humidity;

L298N motor(A1, A2, A3, A4); //initialize motor driver

void setup() {
  //initialize communication
  Serial.begin(9600);
  dht.begin();
  motor.begin();
  pinMode(LIGHT_PIN, OUTPUT);
}

void loop() {
  // read values from the sensor
  temperature = dht.readTemperature();
  humidity = dht.readHumidity();

  String myString;
  myString = ""+Temperatues;
  myString += ","+humidity ;



  // control the motor and light based on the temperature and humidity values
  if (temperature > 30) {
    motor.setSpeed(255); // set the speed of the motor
    motor.run(FORWARD); // run the motor
    myString+=","+"ON";
  } else {
    motor.run(RELEASE); // release the motor
    myString+=","+"OFF";
  }

  if (humidity < 20) {
    digitalWrite(LIGHT_PIN, HIGH);
    myString+=","+"ON";
  } else {
    digitalWrite(LIGHT_PIN, LOW);
    myString+=","+"OFF";
  }

  delay(2000);
}
```

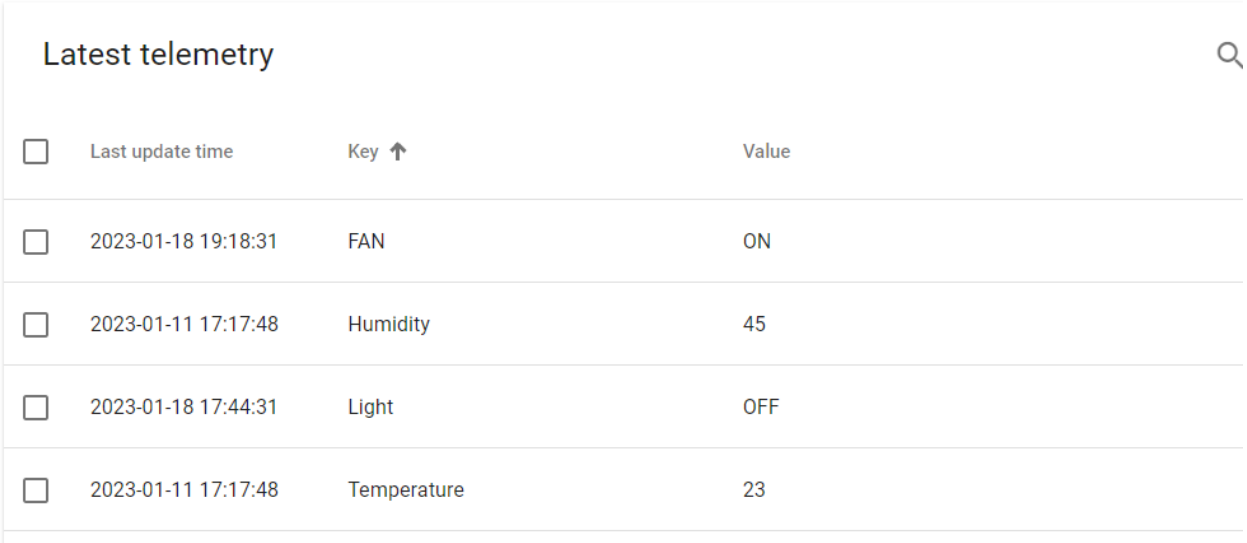

Dashboards > Temperature
VISHAL KUMAR MAHATHA 20BRS1168
Tenant administrator
Temperature
Temperature
Temp_Light
Realtime - last minute
Temperature
23 °C
Light
OFF
Humidity
45 °C
FAN
ON


COM3
function
http request
debug 12


Edit http request node
Delete
Cancel
Done
Properties
Method
POST
URL
https://demo.thingsboard.io/api/v1/FKKzVzC8ogb(
Enable secure (SSL/TLS) connection
Use authentication
Enable connection keep-alive
Use proxy
Only send non-2xx responses to Catch node
Disable strict HTTP parsing


Properties
Name
function
Setup
On Start
On Message
On Stop
var m = msg.payload.split(",")
var json = {"Temperature":parseFloat(m[0]),
"Humidity":parseFloat(m[1]),
"Fan":m[2],
"Light":m[3]};
msg.payload = json;
return msg;

OUTPUT :


Latest telemetry
Last update time
Key
Value
2023-01-18 19:18:31
FAN
ON
2023-01-11 17:17:48
Humidity
45
2023-01-18 17:44:31
Light
OFF
2023-01-11 17:17:48
Temperature
23