Software and Data Engineering Project

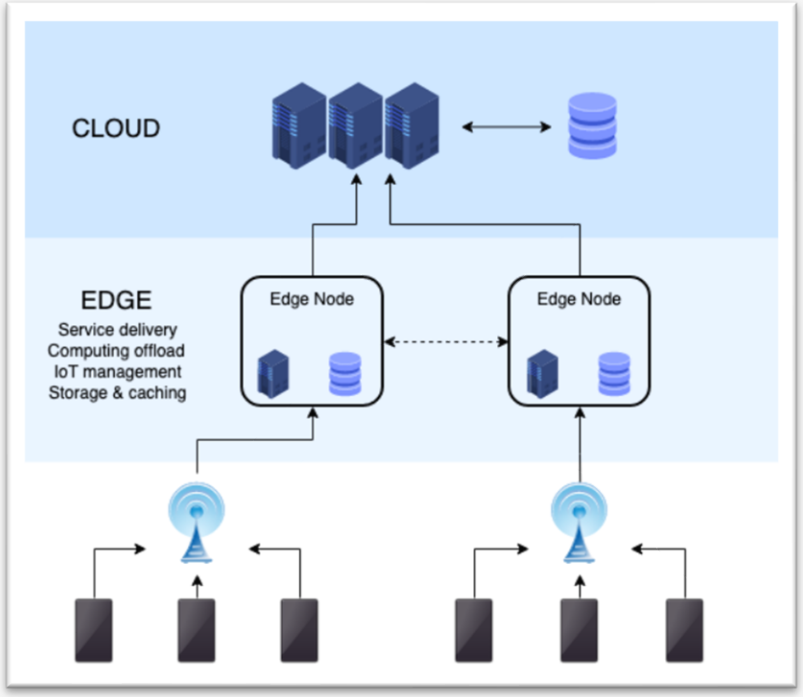
Live Streaming of Data Through Edge Gateways Using Azure.

Submitted By:-
ADITYA THITE - M22AI1001
NIKHIL SETHI - M22AI1006

What is Edge Computing

What is edge computing?

Edge computing takes place at or near the physical location of either the user or the source of the data. By placing computing services closer to these locations, users benefit from faster, more reliable services with better user experiences, while companies benefit by being better able to support latency-sensitive applications, identify trends, and offer better products and services.



Example : When transporting petroleum in a large containers in ship, there is a possibility of leakage in the pipes which can cause loss of millions of dollars. This can be avoid prior by utilizing the power of edge computing and edge gateways.

Ref Link: https://www.redhat.com/en/topics/edge-computing/iot-edge-computing-need-to-work-together#:~:text=An%20IoT%20device%20is%20a,data%20is%20collected%20and%20processed.

What are Edge Gateways

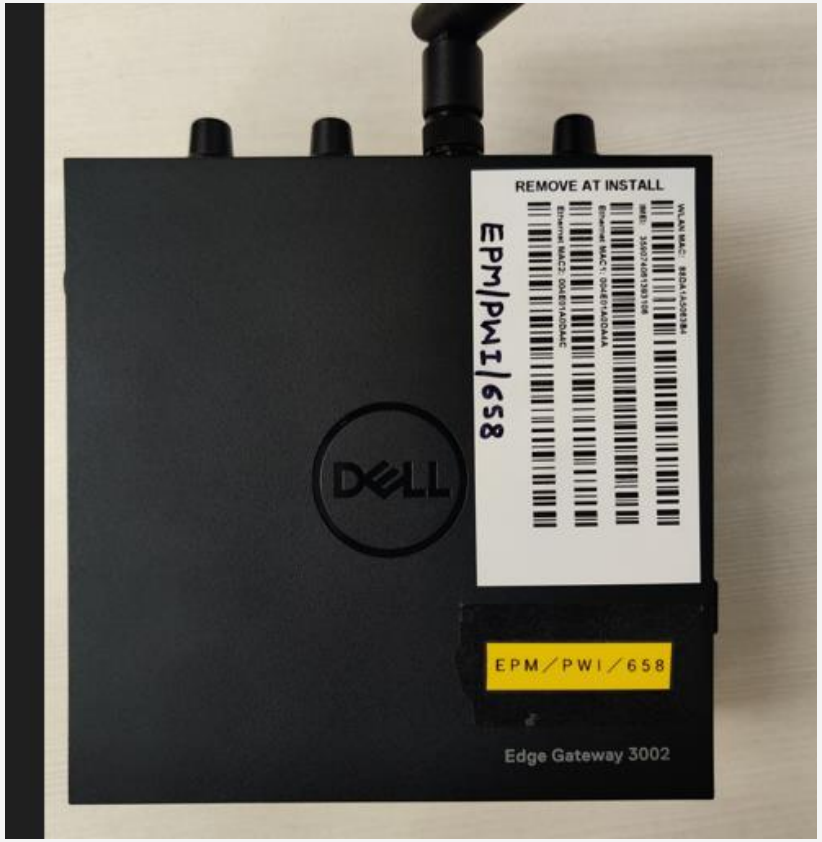
An Edge Gateway is a device or software that acts as an intermediary between edge devices and the cloud. It collects data from the edge devices, performs some processing, and then forwards the data to the cloud or a centralized data center.

There are several companies that manufacture this edge gateways .Eg. Dell, Red lion , etc.

These devices earlier used to have Modbus TCP protocol .Since the trend in the latest MQTT or AMQP protocol , nowadays , these devices are also able to communicate with this protocol. Now, of these devices , some of these devices are able to directly throw the telemetry messages to other sources like cloud technologies like Azure, AWS, and others.

For the legacy devices which are not able to throw those messages, we have to install some modules to connect them to the respective sources.

Edge Gateway



An example of an edge gateway of 3000 series from Dell

What is Azure Iot Hub?

Azure IoT Hub is a cloud-based service provided by Microsoft Azure that enables secure and scalable communication between IoT devices and the cloud. It provides features like device-to-cloud and cloud-to-device messaging, device management, and security for IoT solutions.

Azure IoT Hub helps in implementing edge computing by providing a way to manage and communicate with IoT devices at scale. Azure IoT Hub can be integrated with Azure IoT Edge, which is a service that allows you to deploy and run cloud workloads on edge devices. Azure IoT Edge provides a way to run custom code and Azure services on edge devices, allowing you to perform analytics and processing at the edge.

By using Azure IoT Hub with Azure IoT Edge, you can process and analyze data at the edge, reducing the amount of data sent to the cloud, improving response times, and enhancing security by keeping sensitive data on the edge. Edge gateways can be used to connect the edge devices to Azure IoT Hub and Azure IoT Edge, allowing you to manage and communicate with them securely and reliably.

Ref Link: https://azure.microsoft.com/en-us/products/iot-hub

Data Collection: The Coriolis meter sensors in the petrol pipe collect data on parameters such as temperature, pressure, and flow rate.

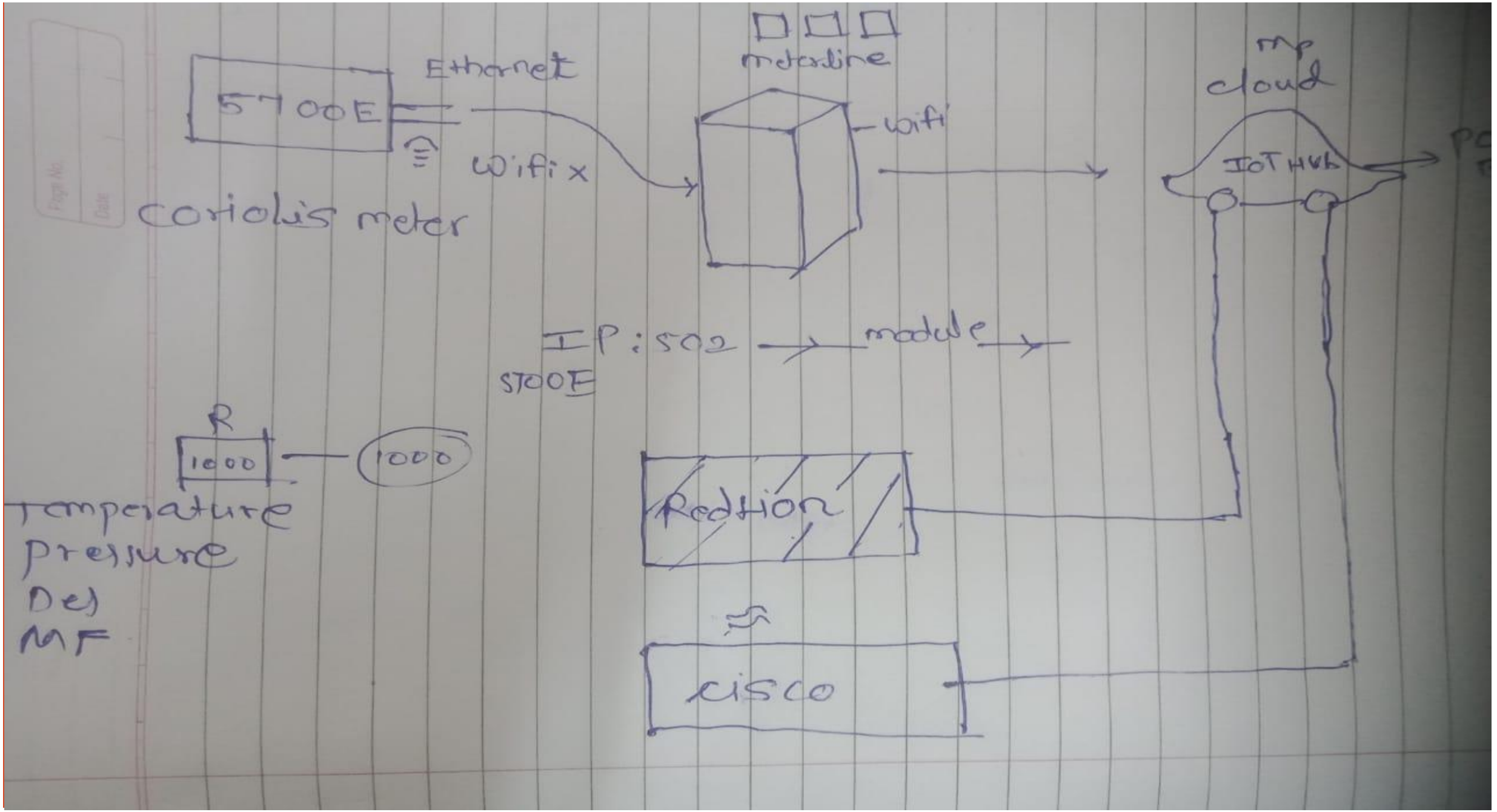
Data Processing: The microcontroller or single-board computer with Ethernet or Wi-Fi module processes the data and prepares it for transmission to Azure IoT Hub. This can involve formatting the data into a specific structure, such as JSON or CSV, and adding any necessary metadata.

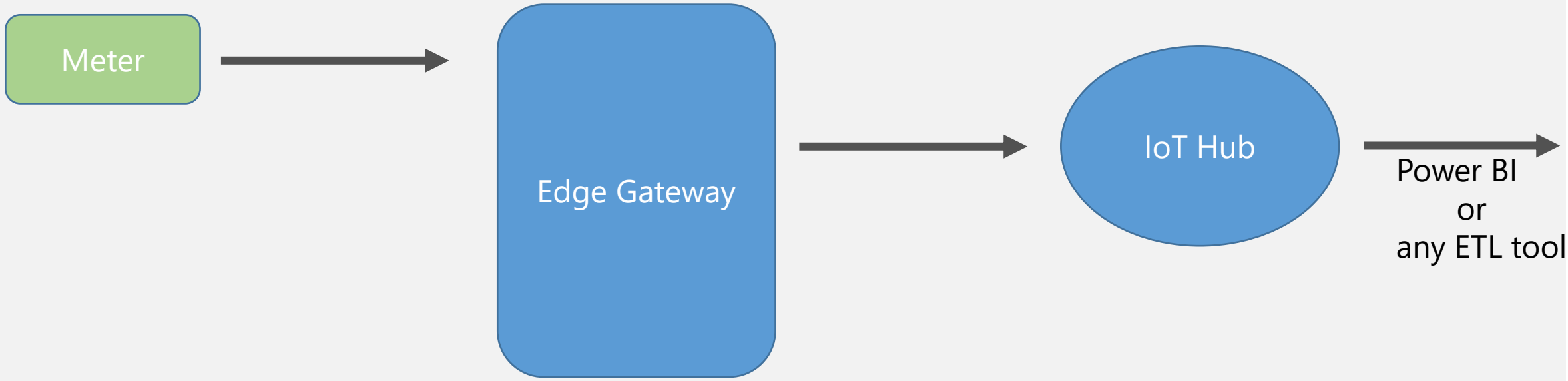
Establishing Connection: The microcontroller or single-board computer connects to Azure IoT Hub using its assigned device identity and security keys. This connection can be established using a protocol such as MQTT or AMQP.

Sending Telemetry Data: The microcontroller or single-board computer sends telemetry data to Azure IoT Hub using the previously established connection. The telemetry data typically includes the actual sensor data from the Coriolis meter, such as the current flow rate, temperature, and pressure. This telemetry data is sent in real-time, allowing for near-instantaneous analysis and processing.

Storing the Data: Azure IoT Hub stores the telemetry data securely and reliably in the cloud. The stored data can be accessed and analyzed in real-time using tools such as Azure Stream Analytics or Azure Functions.

6. Querying the Data: Once the data is stored in Azure IoT Hub, it can be queried using SQL-like queries. This enables data analysts to extract specific information from the data, such as the average flow rate over a given time period.


5700E
Ethernet
Wifi x
coriolis meter
metaline
wifi
my cloud
IoT HUB
IP:502
5700E
module
R
1000
1000
temperature
pressure
Del
MF
Redlion
cisco


Meter
Edge Gateway
IoT Hub
Power BI
or
any ETL tool

Creating the Azure Iot Hub

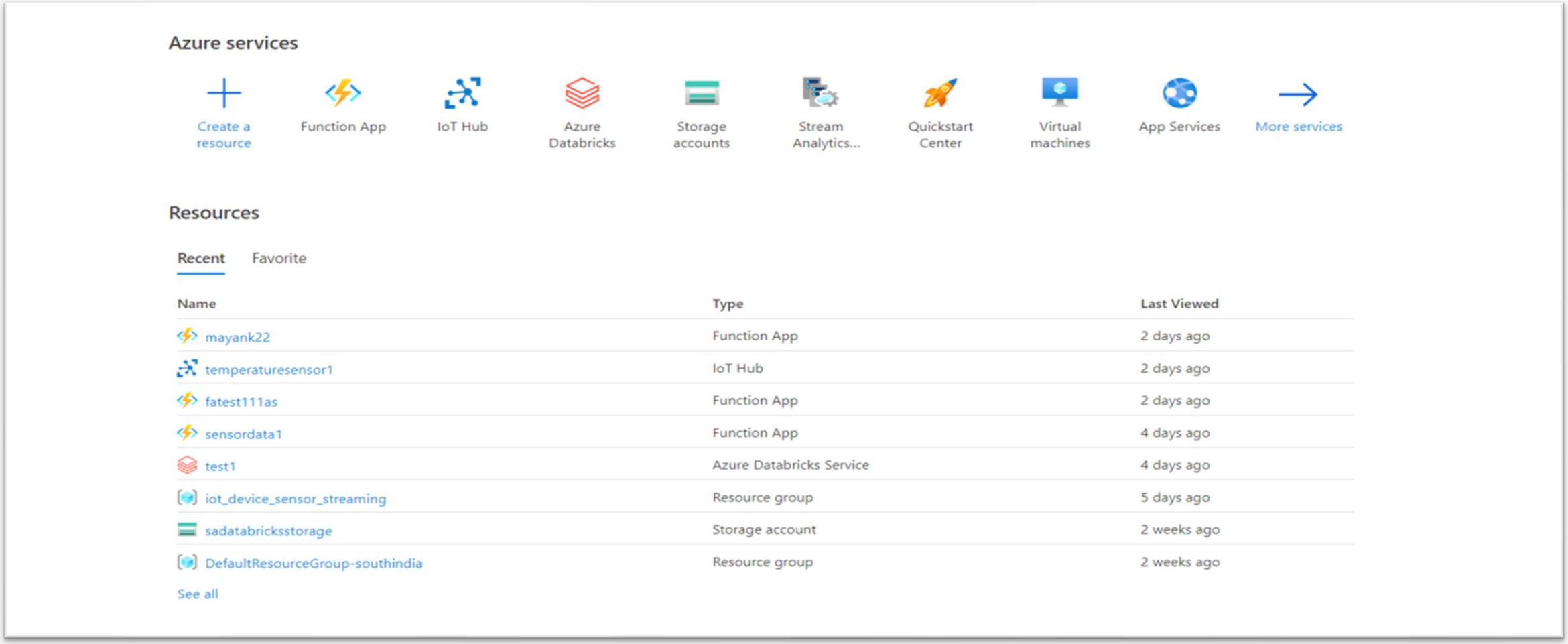
- Here, we will be showing streaming of data from those edge devices using Azure technologies like Azure IoT Hub which is specifically design for IoT devices, Azure Function app which will be useful for the transformation of the semi structured data that we get from those devices and others like Azure SQL, and others.

- Due to limitations in the subscription for our Azure account(students version), we were not able to fully utilize the services and thus the performance issues were quite significant and we were not able to utilize the full capabilities of the Azure as it is available in paid version .

- However , we can fully understand the capabilities that one can have in Azure and understand the end to end data pipeline that we can built for live streaming of data using the students subscription.

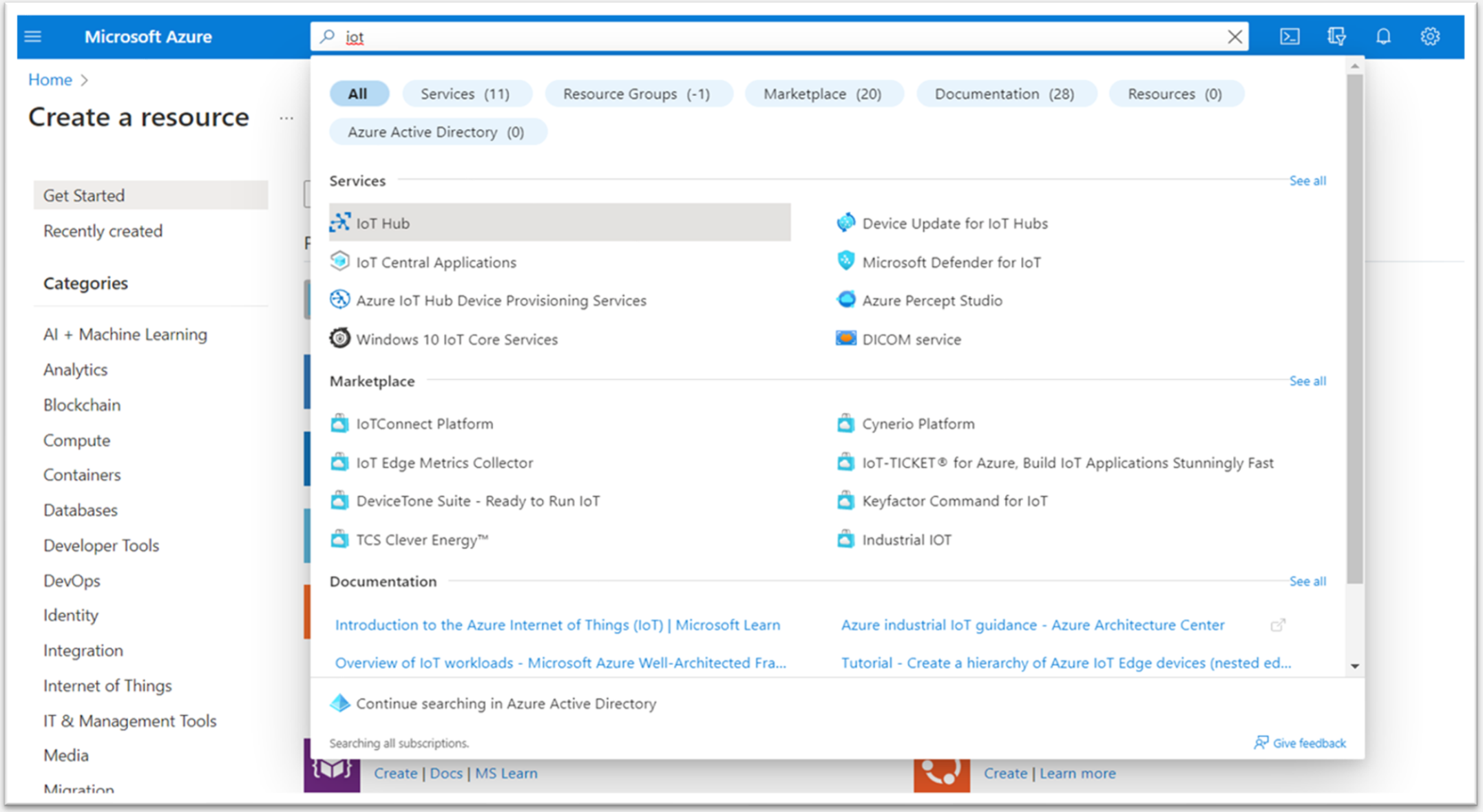
The UI of the Azure looks similar to this

Here, we have created the students free trial account in the Azure

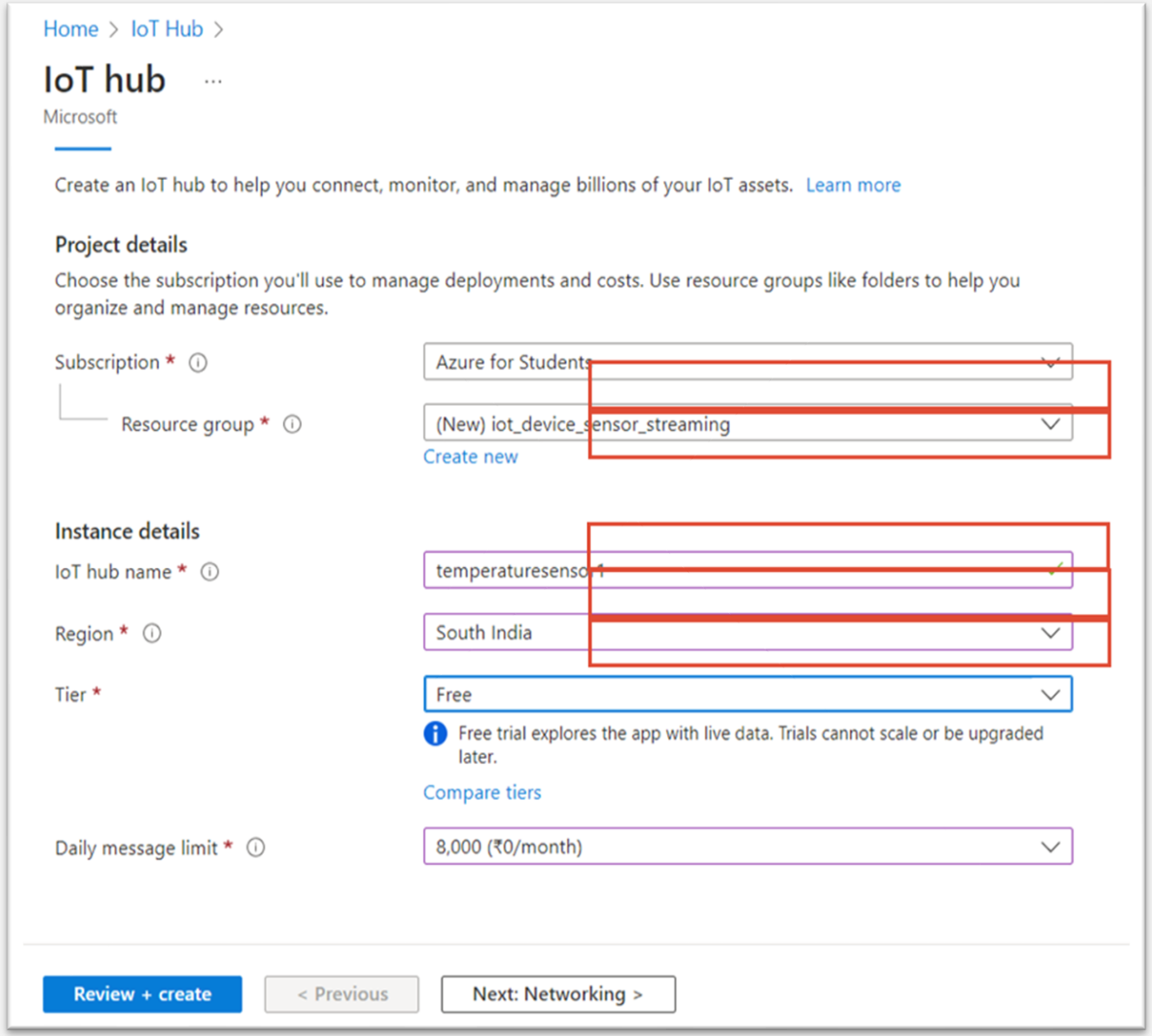


Setting up an IoT Hub

- Create an Azure account and log in to the Azure portal
- Create a new IoT Hub resource and select the desired settings, such as the pricing tier, resource group, and region
- Once the IoT Hub is created, select it and navigate to the "Shared access policies" section
- Create a new policy with "service connect" and "device connect" permissions

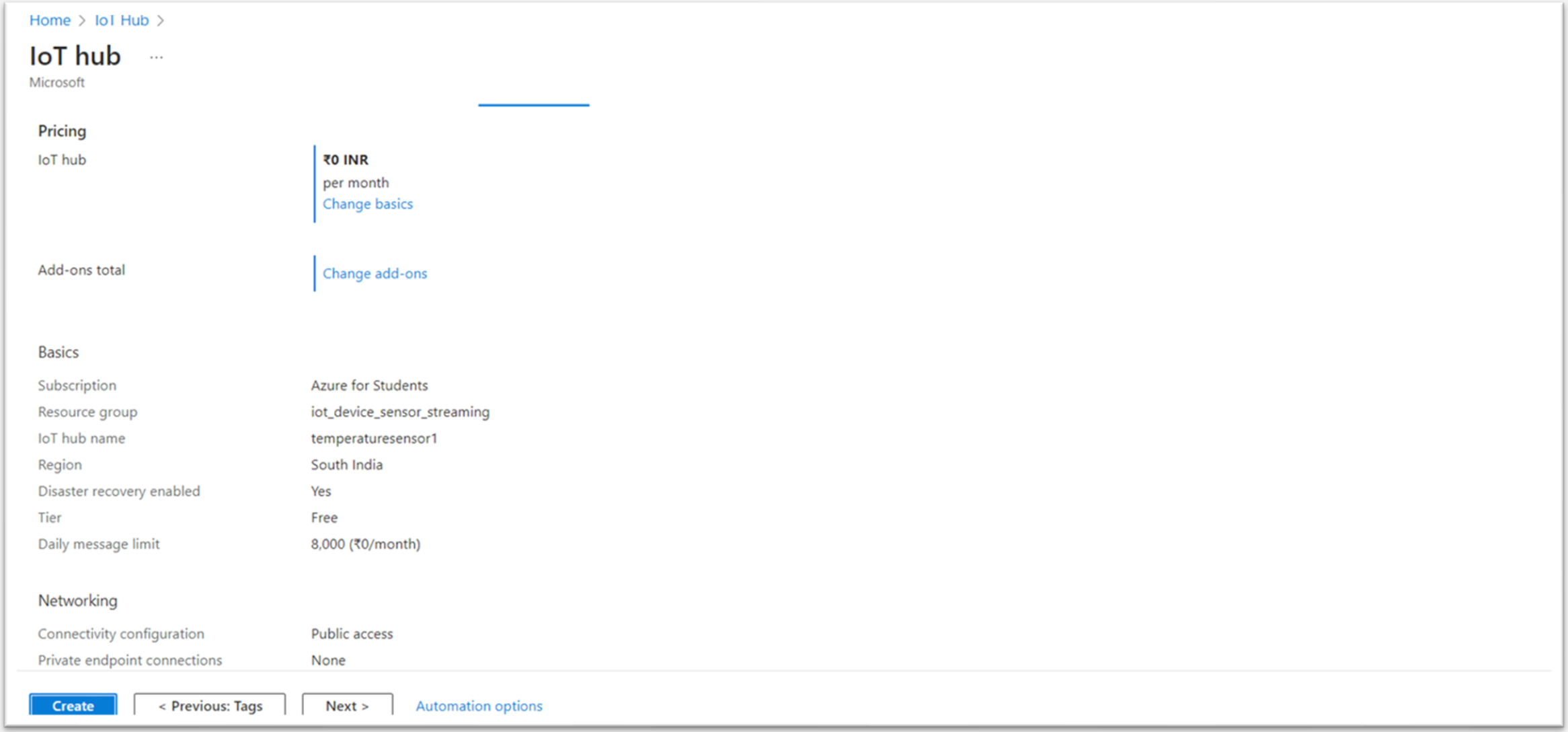


•Create an IoT Hub - This hub acts as a central hub for all your IoT devices, and it will be the gateway for data exchange between devices and the cloud.

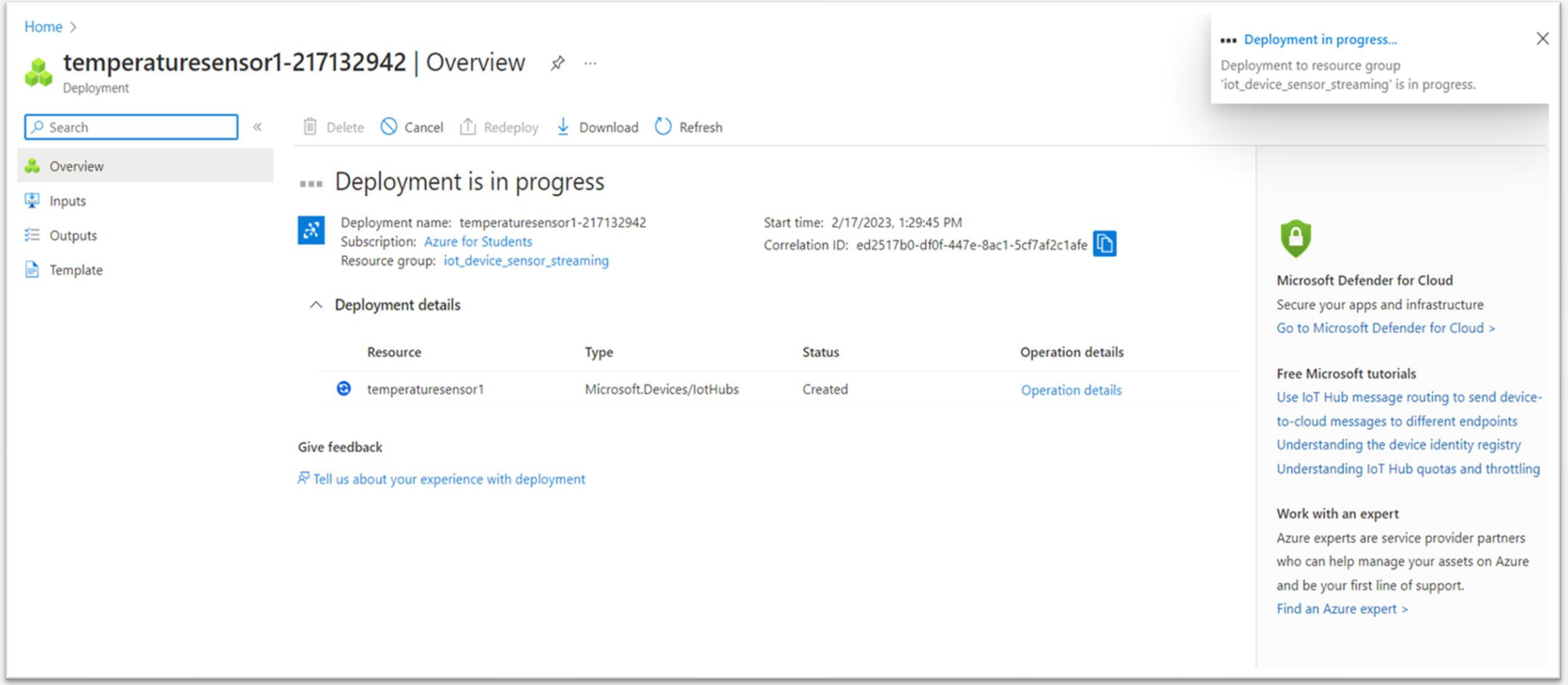


Create a new IoT Hub resource and select the desired settings, such as the pricing tier, resource group, and region

Creating the Azure Iot Hub ...

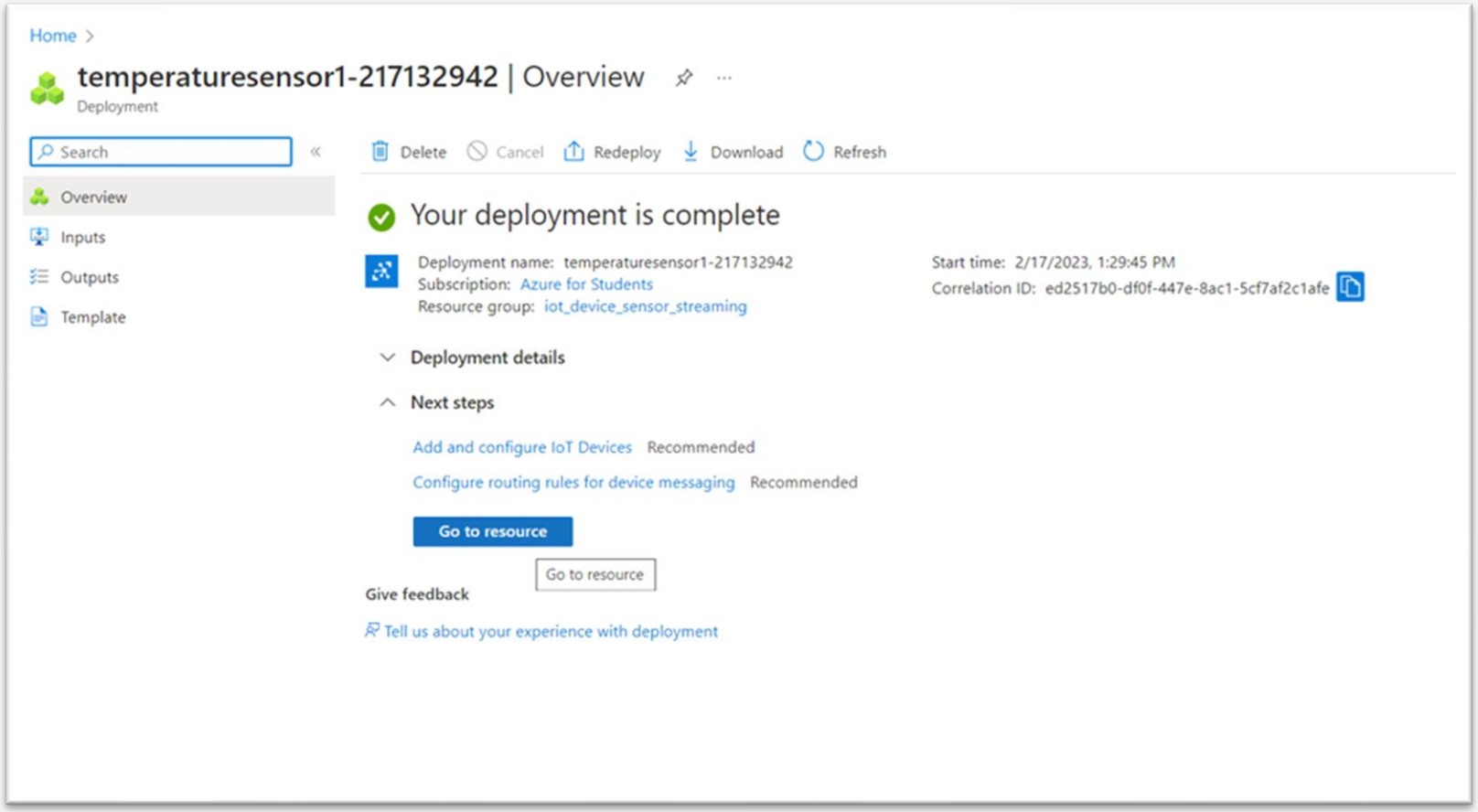


Deployment is progress ….



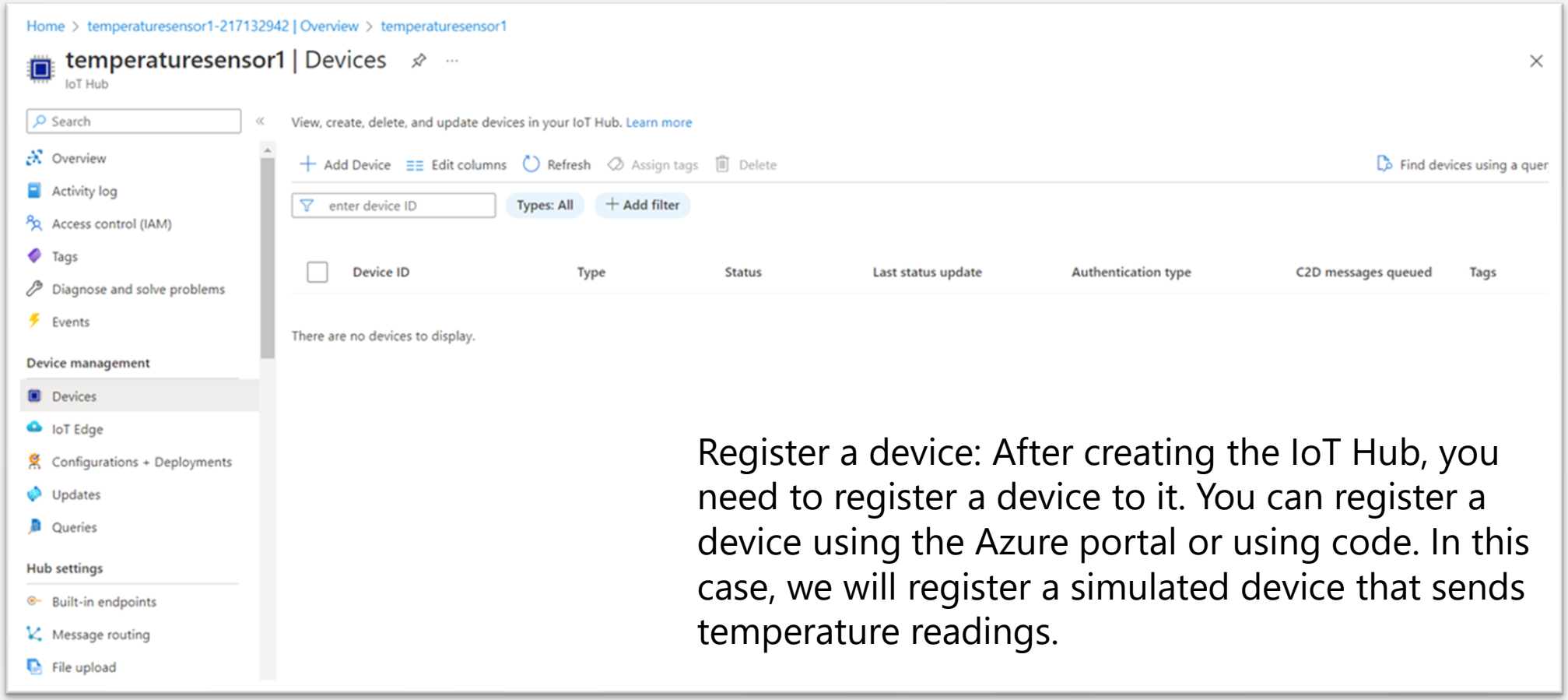
Iot Hub is created

Now, our IoT Hub is ready here. We can now connect this created hub with the edge devices and can get the required data from there .



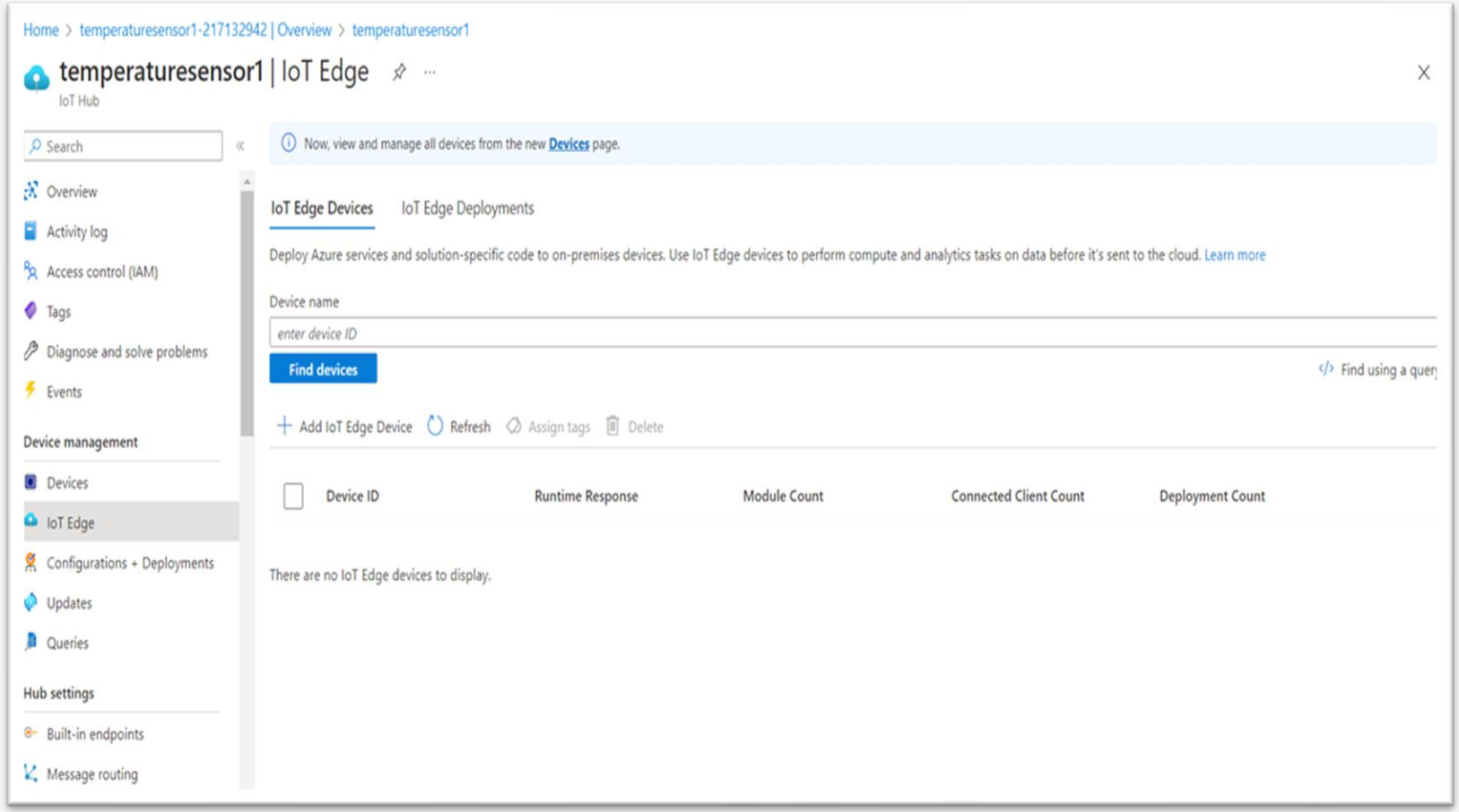
Now, once ready, we can register the device that we have to connect . For this task, it has 2 options as you can see in the image, viz Devices and Iot Edge .

- If some device has directly provision to through the messages to cloud , we can click on the device tab and then create it and then supply the required connection string in the device
- If some device has not the provision to throw messages to Azure, we can use the Iot Edge provided by the Azure and install the IoT Edge runtime on the device



Creating the IoT Edge device

Here, we don't have these devices as these are costly and widely used in the industrial applications . So, thus , for testing we will be considering one Linux virtual machine which will be acting as the edge device for us .



Now, before jumping into device , first lets create the device in the Azure IoT Hub for connecting our device

Home > temperaturesensor1-217132942 | Overview > temperaturesensor1 | IoT Edge >

Create a device ...

Find Certified for Azure IoT devices in the Device Catalog

Device ID *

sensor1

IoT Edge Device

Authentication type

Symmetric key | X.509 Self-Signed

Auto-generate keys

Connect this device to an IoT hub

Enable | Disable

Parent device

No parent device

Set a parent device

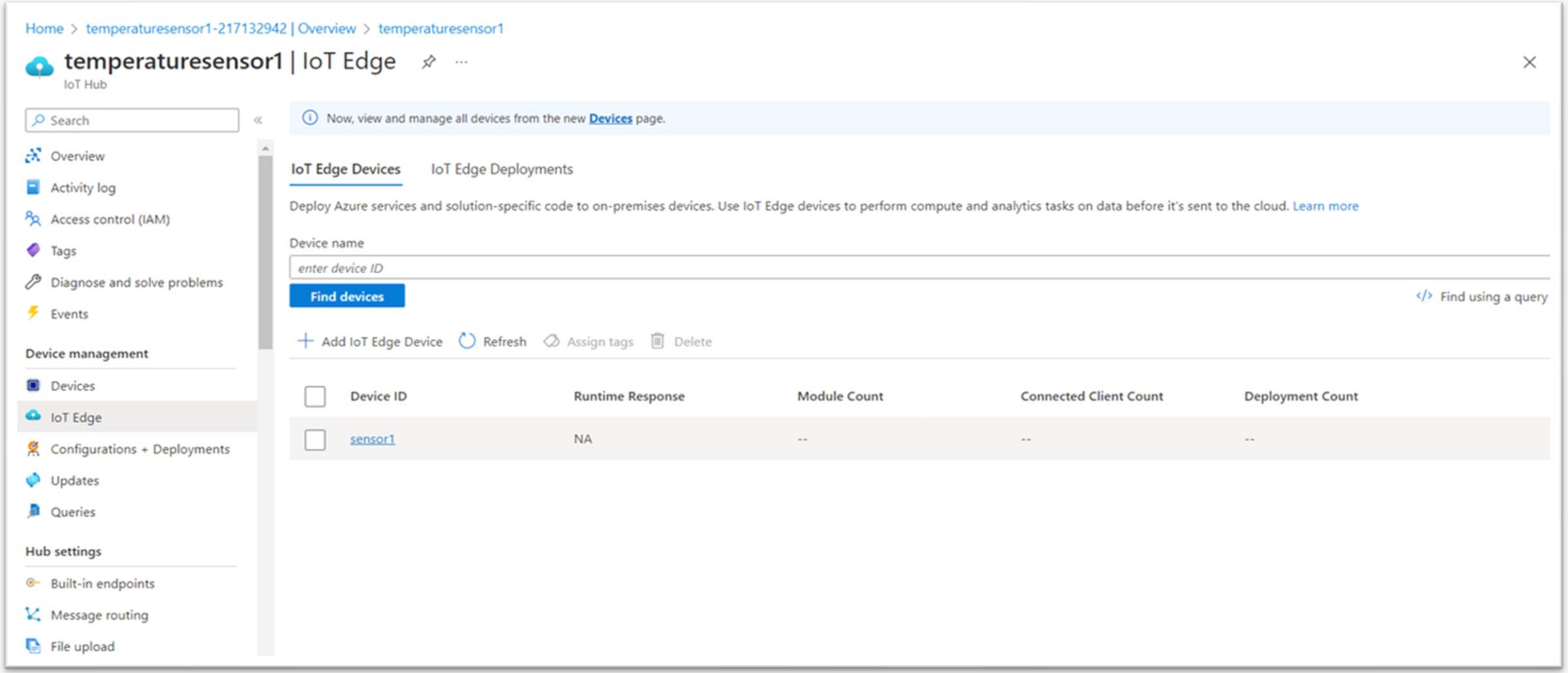
Child devices

0

Choose child devices

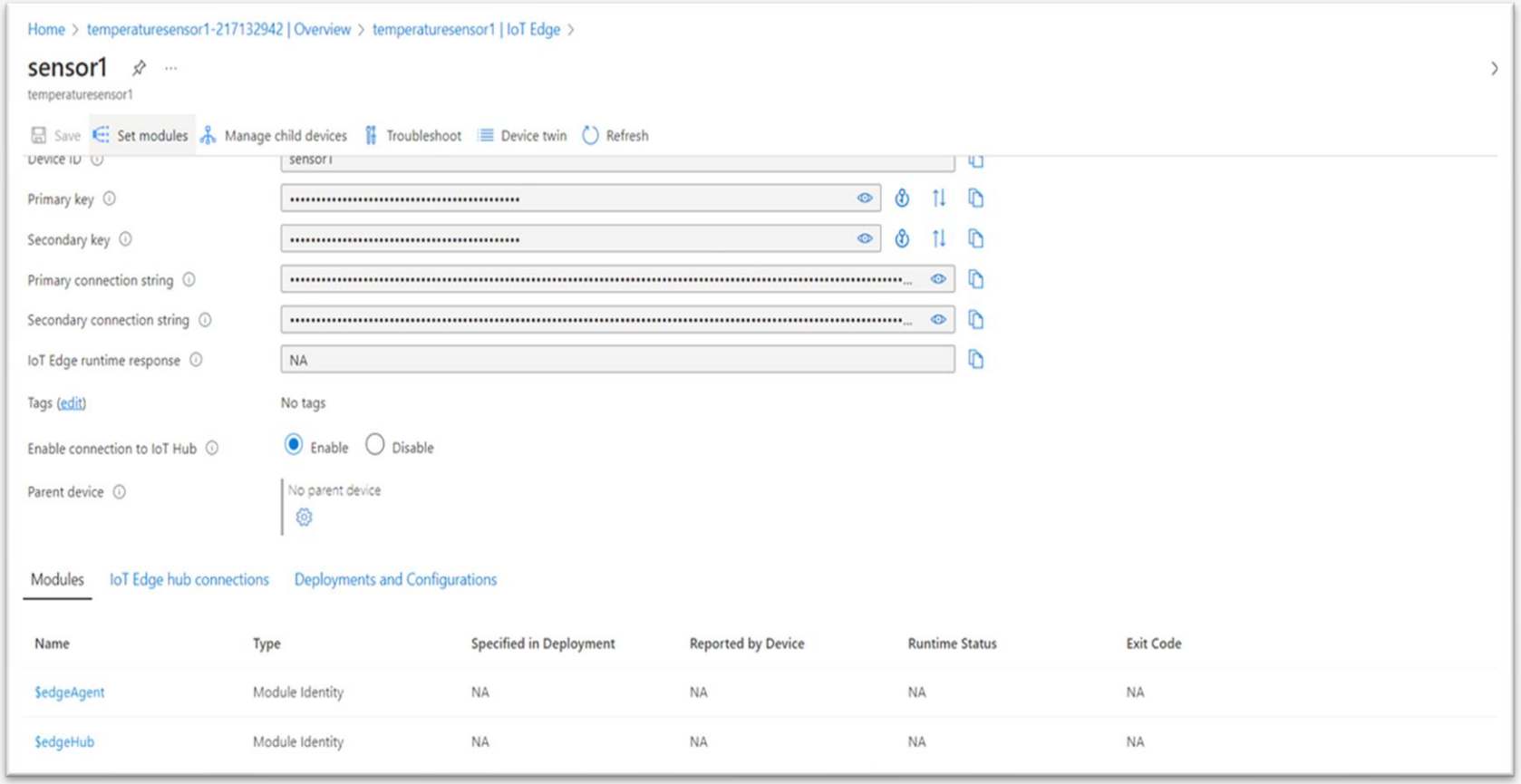
The device is ready…

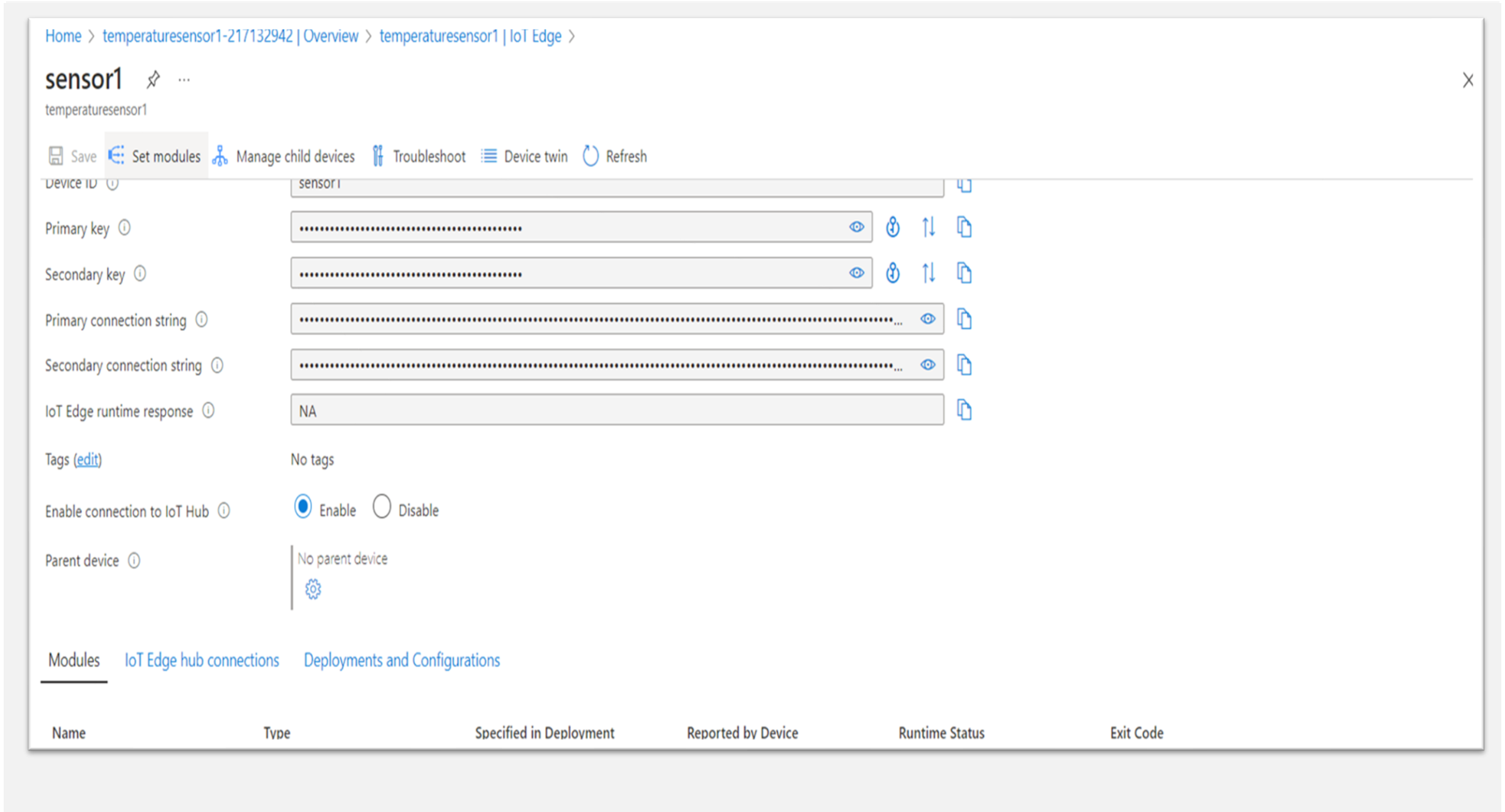
Here, as you can see , we have created the device name sensor1. The device is ready now.



When the device is created , we will get 2 modules created automatically with it , Edge Agent and Edge Hub . Each of them has its significance and is basically is used to run the edge runtime on the device . This can be seen as below image. Also, we can note down the primary connection string from here , which , later we will be using it to connect the device to the Azure IoT Hub.

Reference link : https://learn.microsoft.com/en-us/azure/iot-edge/iot-edge-runtime?view=iotedge-1.4




Home > temperaturesensor1-217132942 | Overview > temperaturesensor1 | IoT Edge >
sensor1
temperaturesensor1
Save
Set modules
Manage child devices
Troubleshoot
Device twin
Refresh
Device ID
sensor1
Primary key
Secondary key
Primary connection string
Secondary connection string
IoT Edge runtime response
NA
Tags (edit)
No tags
Enable connection to IoT Hub
Enable
Disable
Parent device
No parent device
Modules
IoT Edge hub connections
Deployments and Configurations
Name
Type
Specified in Deployment
Reported by Device
Runtime Status
Exit Code

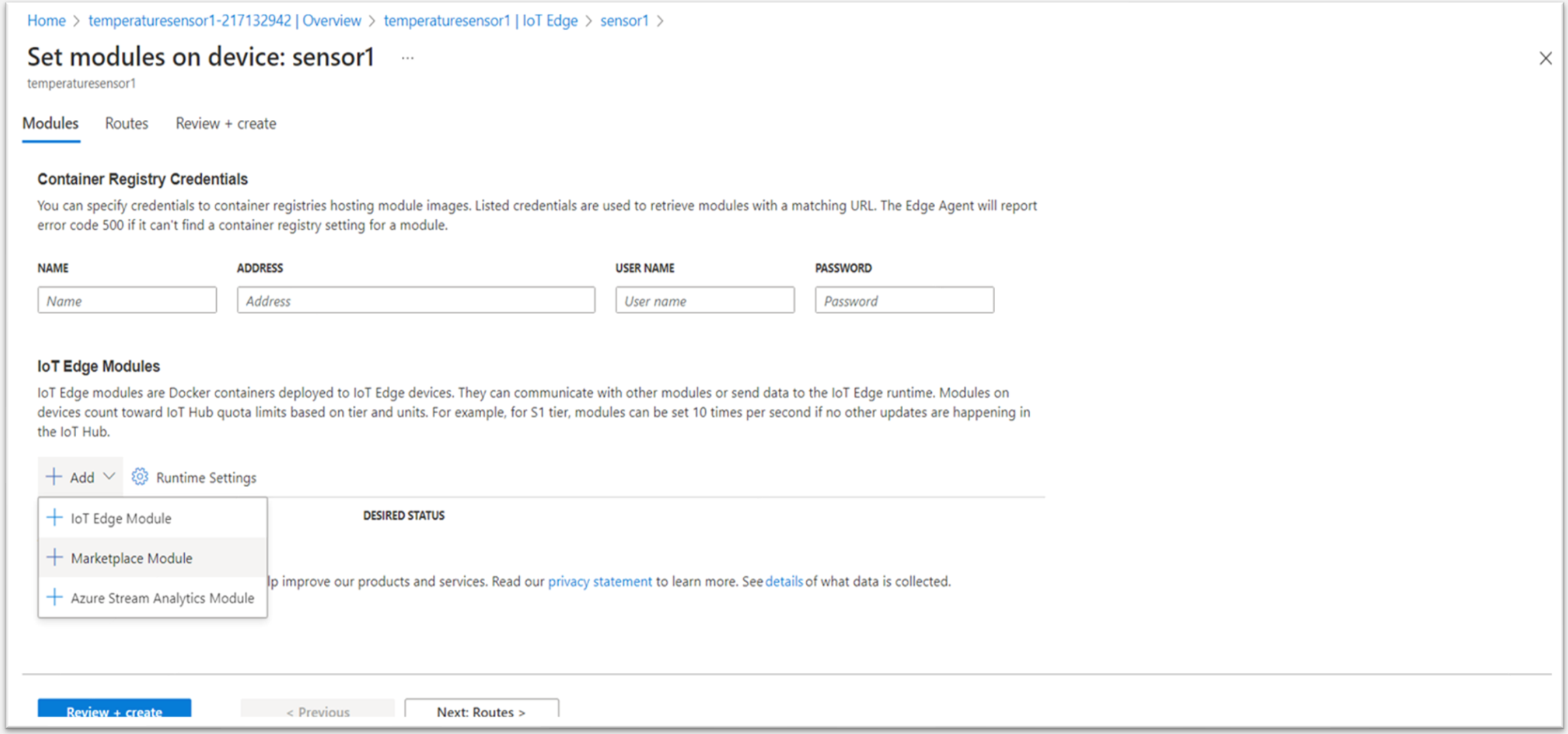
Setting up Temperature Sensor Module

Now, as we are not having any meter to connect to the edge gateway, we will be installing a dummy data module which is provided by the Azure which is called the Temperature Sensor Module. This module throws emits the dummy data as it is from real sensor. Azure has specifically created the module for testing purpose.

For this , we will have to click on the Set Modules tab which can be seen in the picture and then from the Market place section , we can directly instal it .

Note: Some legacy edge gateways doesn’t have the capability to communicate in the MQTT/ AMQP protocol. For that, Azure has provided the readymade module name Modbus which after implementing on the device, readily converts Modbus to MQTT.

Going to Market place from Set Modules to search for Temperature Sensor



IoT Edge Module Marketplace ...

Marketplace

All

Categories

- Internet of Things (73)
- Analytics (25)
- Compute (12)
- Developer Tools (10)
- IT & Management Tools (10)
- Networking (10)
- AI + Machine Learning (7)
- Security (5)
- Containers (4)
- Databases (4)

Showing 1 to 20 of 73 results in **Internet of Things** with **1 selected filters.** Clear filters

Tile view

Azure Blob Storage on IoT Edge

Microsoft

Iot Edge Modules

Azure consistent block blob storage on IoT Edge

Select

Simulated Temperature Sensor

Microsoft

Iot Edge Modules

IoT Edge module that simulates a temperature sensor.

Bring your own license

Select

Azure SQL Edge

Microsoft

Iot Edge Modules

Azure SQL Edge module for IoT Edge

Select

OPC Publisher

Microsoft

Iot Edge Modules

Connects to OPC UA Servers and publishes OPC UA node data values in OPC UA Pub/Sub compatible format

Select

IoT Edge Metrics Collector

Microsoft

Iot Edge Modules

Collect Prometheus metrics from IoT Edge modules and transport them to Azure Monitor or IoT Hub

Select

YOLOv3 gRPC Extension for AVA Edge

motojin.com, Inc.

RedisEdge

Redis Labs

Modbus

Microsoft

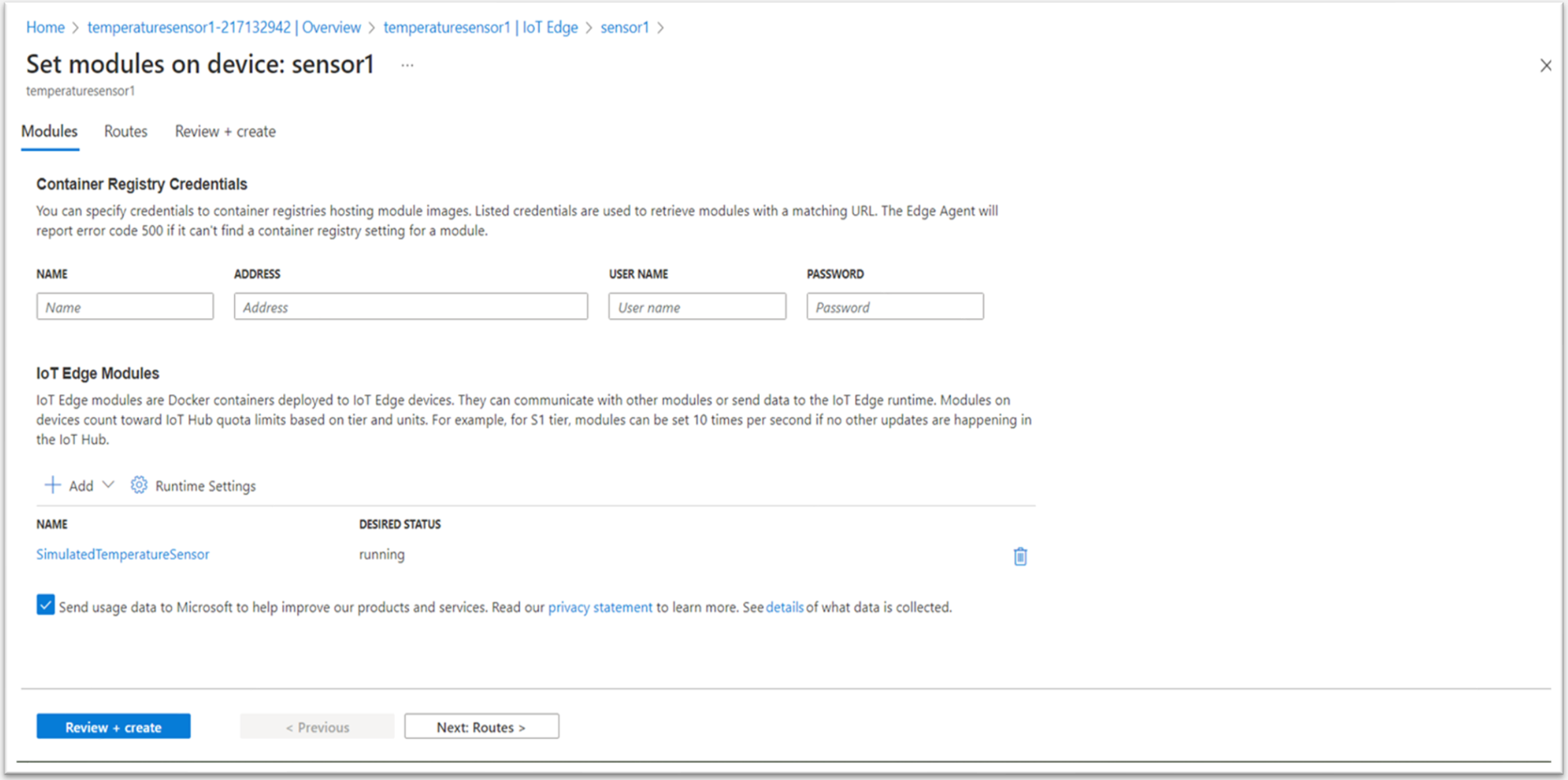
Node-RED on IoT Edge

motojin.com, Inc.

Azure Security Center for IoT

Microsoft

Installed the temperature sensor Module for getting the dummy data for testing



Set modules on device: sensor1

temperaturesensor1

Modules Routes Review + create

Validation passed.

Deployment

The text box below displays the deployment to be submitted.


```
{
    "modulesContent": {
        "$edgeAgent": {
            "properties.desired": {
                "schemaVersion": "1.1",
                "runtime": {
                    "type": "docker",
                    "settings": {}
                },
                "systemModules": {
                    "edgeAgent": {
                        "settings": {
                            "image": "mcr.microsoft.com/azureiotedge-agent:1.4"
                        },
                        "type": "docker"
                    },
                    "edgeHub": {
                        "restartPolicy": "always",
                        "settings": {
```


Create < Previous Next >

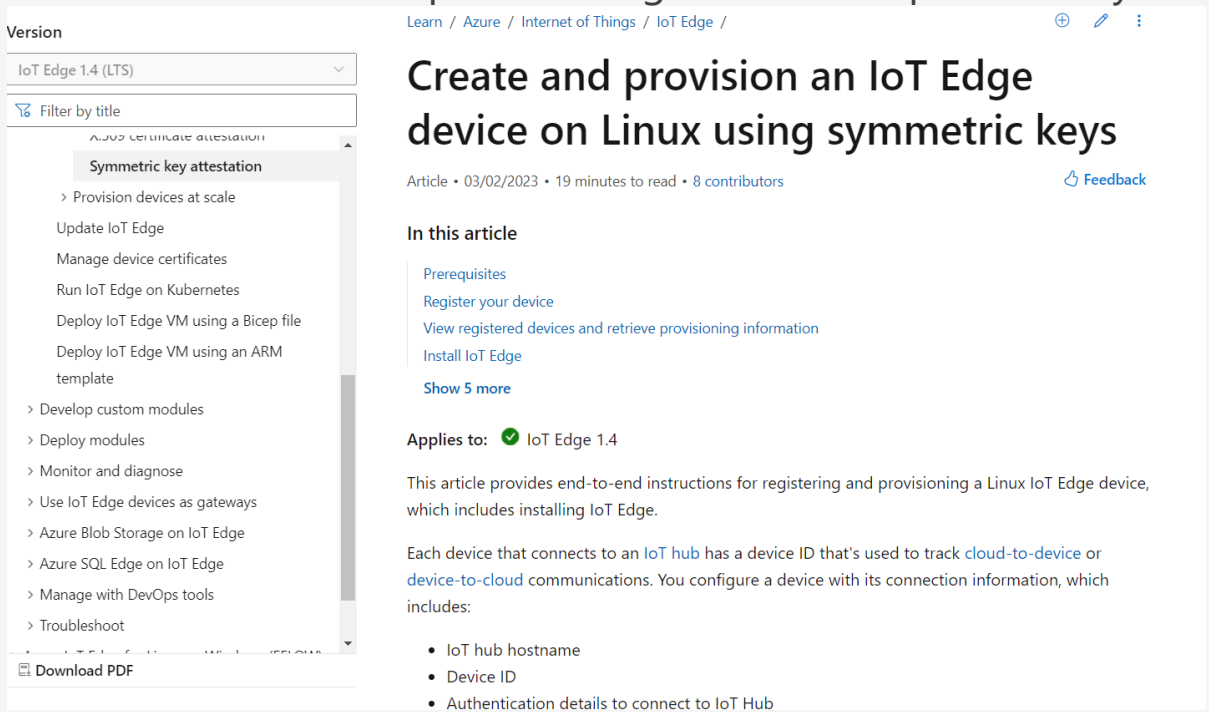
Iot Hub is Ready

Our Iot Hub is ready with the required modules in it that we will be needing to connect with our device. Now, lets connect these Iot hub with the Linux virtual machine that we will be using as a edge gateway.

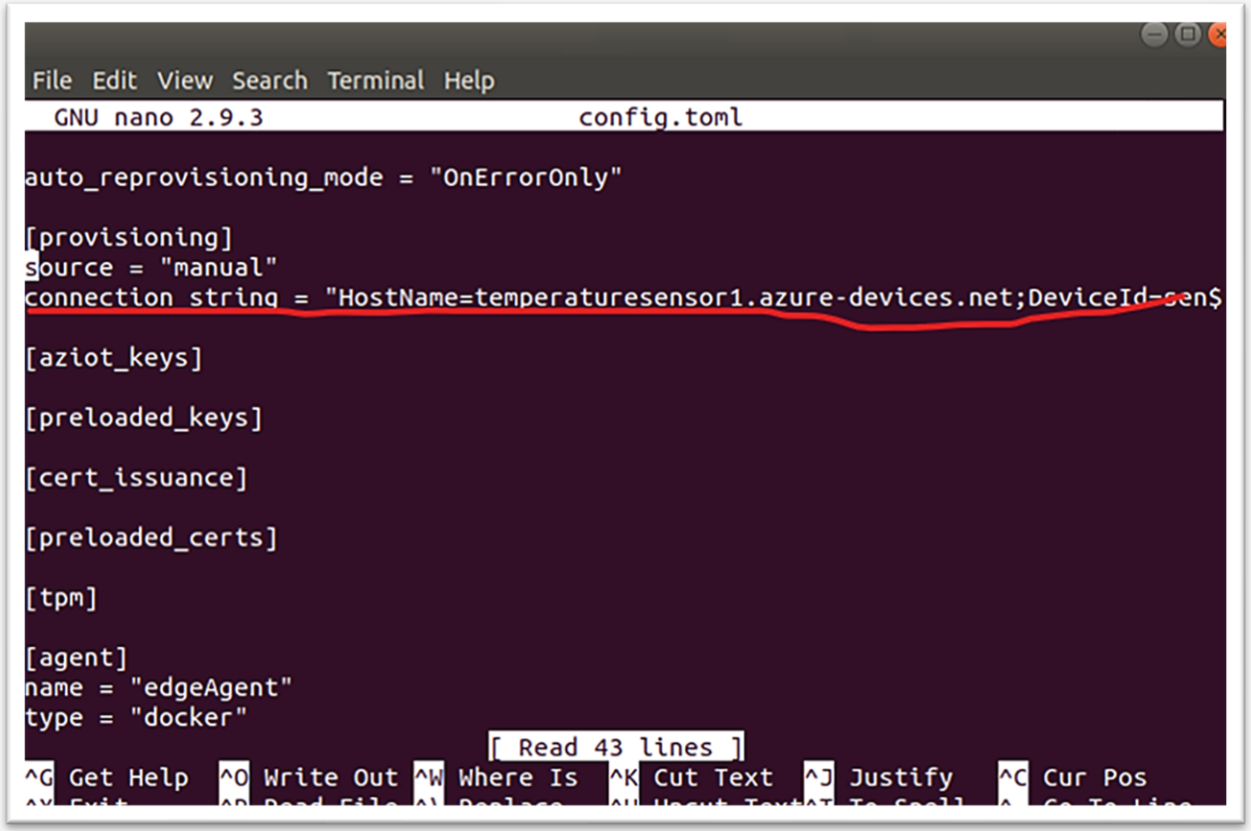
I have created the virtual machine with the OS as ubuntu 18.04. As this machine is not capable of throwing messages directly, we will need to install the Azure Iot edge runtime on the machine. The steps according to the OS is provided by the Microsoft for the installation.

I have followed those steps and run commands accordingly on my Linux system for the installation of Iot Edge runtime.

The reference link : https://learn.microsoft.com/en-us/azure/iot-edge/how-to-provision-single-device-linux-symmetric?view=iotedge-1.4&tabs=azure-portal%2Cubuntu



After installation of azure iotedge run time , the config file will be formed by the names config.toml in the installation path and thus, for connecting our Iot Hub with the device, we have to give our connection string that we have noted earlier in the configuration file as shown below



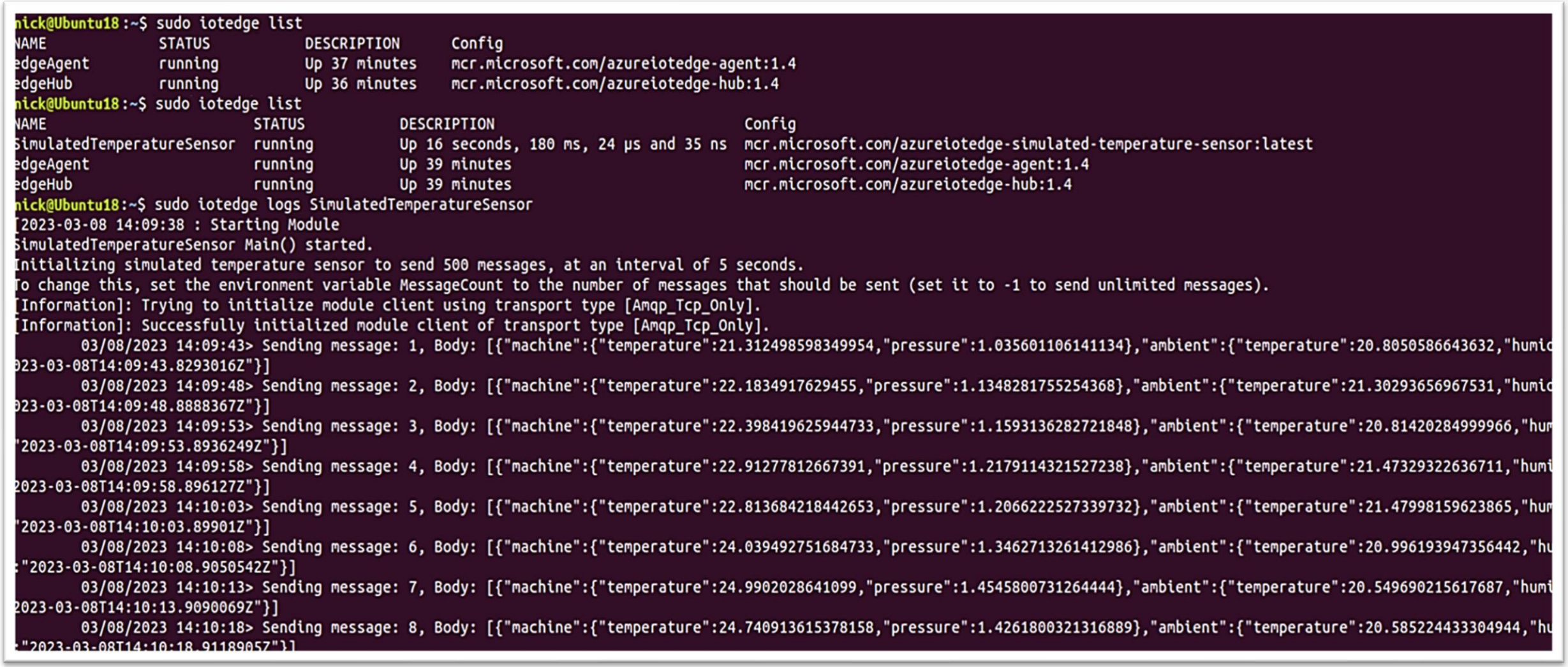
We can save the config file now .

Now , lets check if the iot edge runtime is installed successfully on our machine.

We can check this by using the command - **sudo iotedge check**

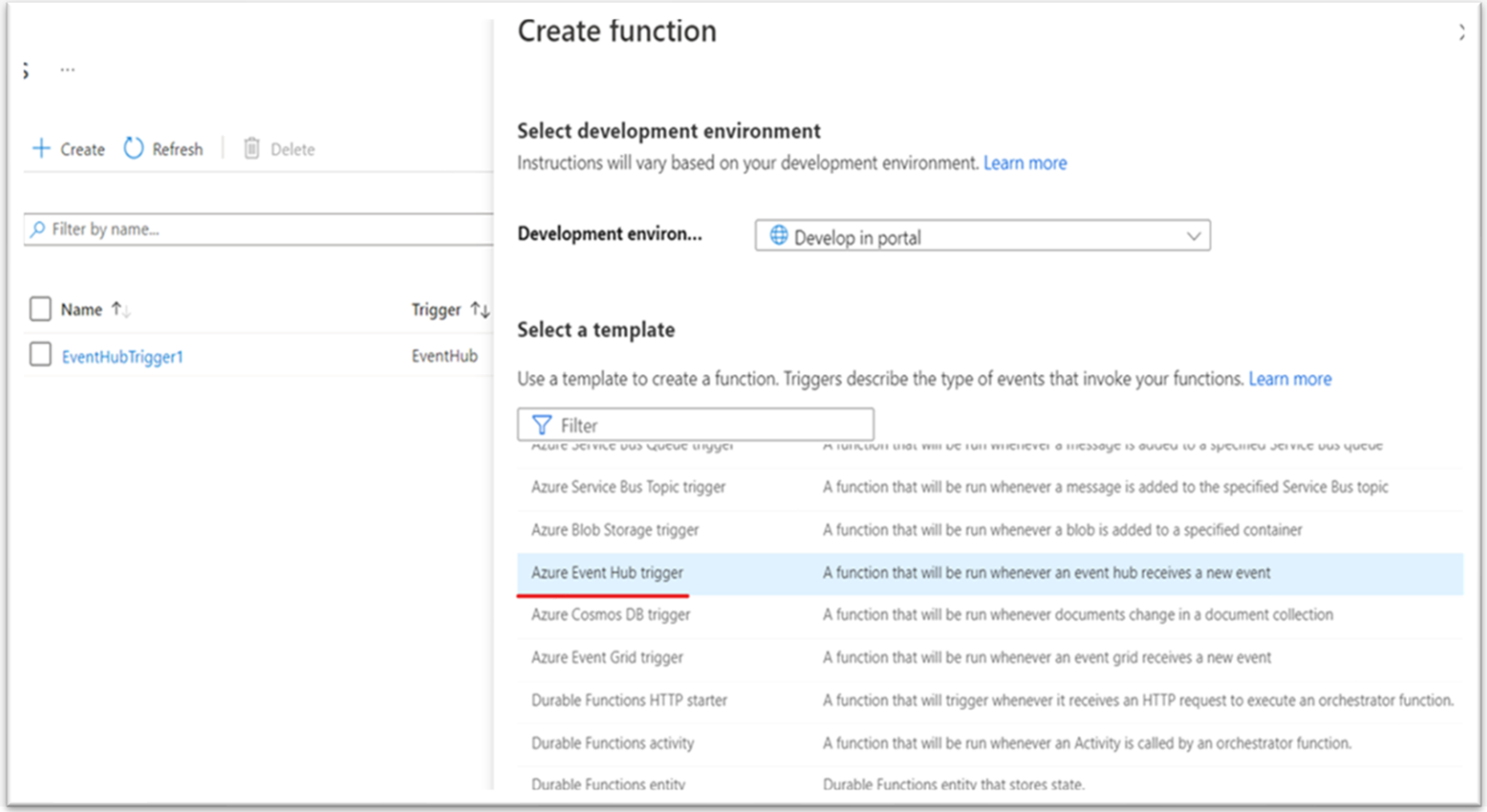
This command will run and check if everything is in the right place as shown . Looks like the connectivity to the Azure iothub is successful.

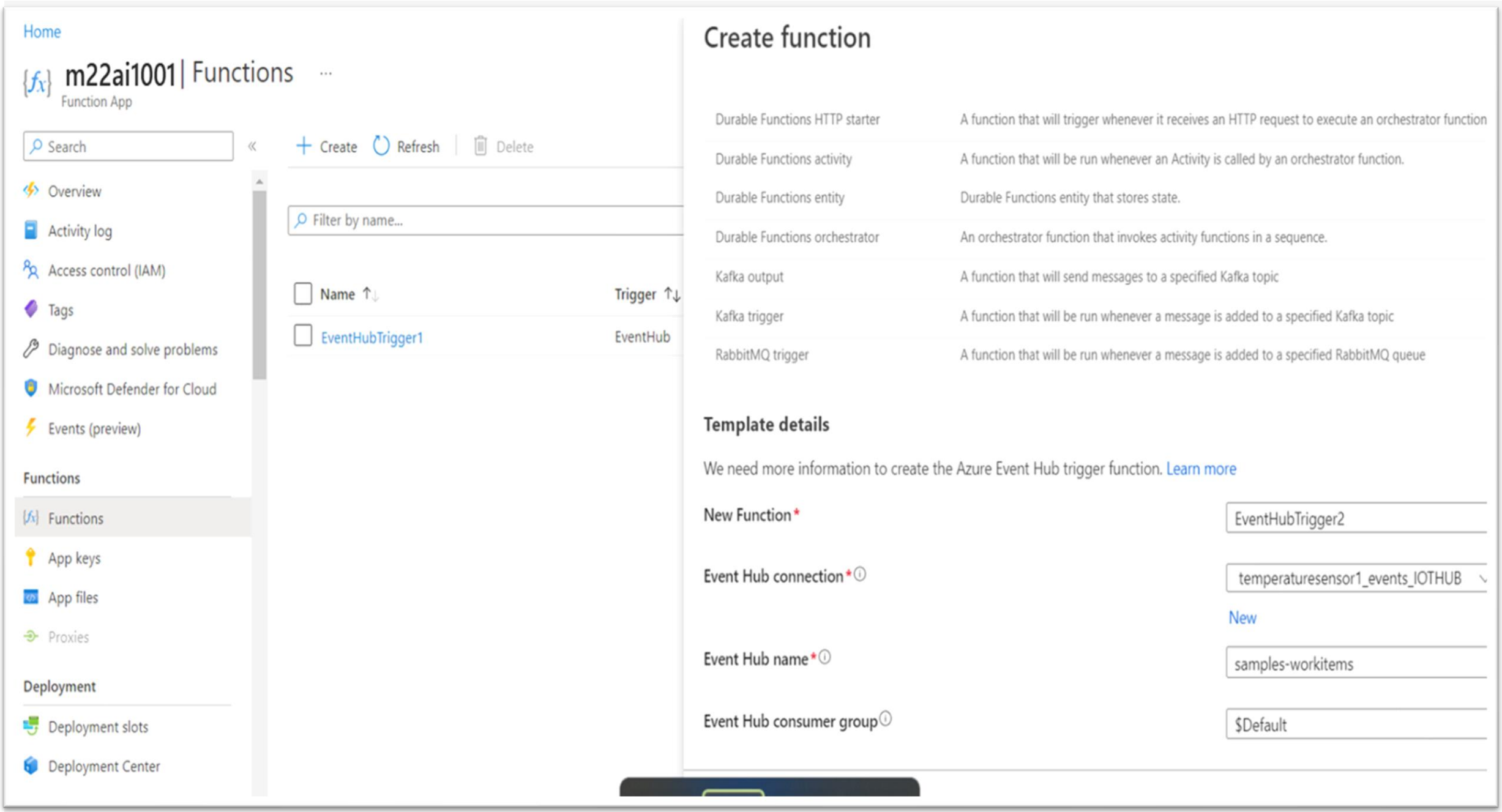
We can see our respective module here as shown through the commands as shown in the figure and also can see the logs for the temperature sensor that we are getting.



Creating a function to the IoT Hub

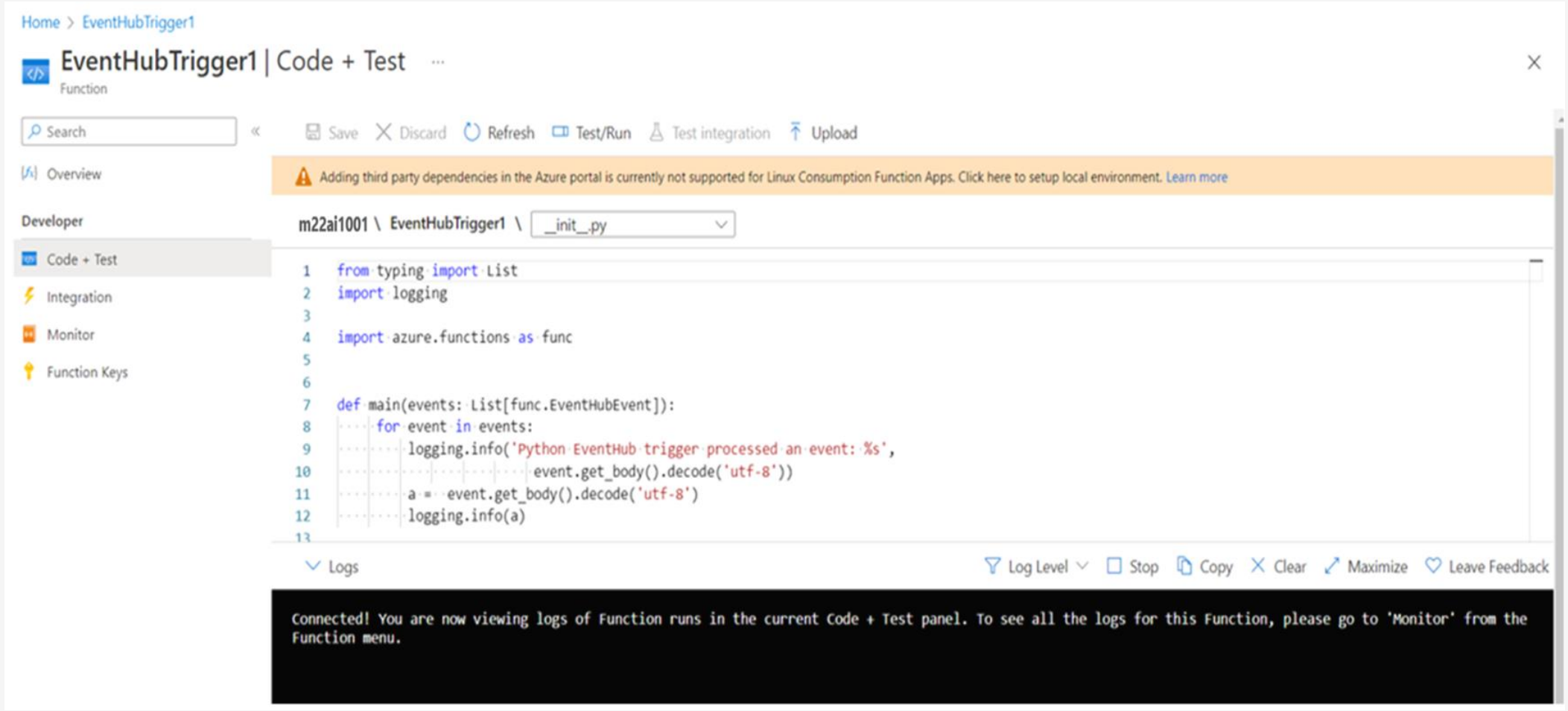
As we can see here , the data is in semi structured format and thus we will have to do some processing over the data. For Serverless computation , Azure provides the service know as Function App where we can write the code in Python, Java, JavaScript (NodeJS). Here, there are different triggers available for the function meaning when we have to run that function . There is specific trigger also available for IotHub on which the function will run whnever the data will come to the Hub.




Home
m22ai1001 | Functions
Function App
Search
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Microsoft Defender for Cloud
Events (preview)
Functions
Functions
App keys
App files
Proxies
Deployment
Deployment slots
Deployment Center
Create
Refresh
Delete
Filter by name...
Name
Trigger
EventHubTrigger1
EventHub
Create function
Durable Functions HTTP starter
A function that will trigger whenever it receives an HTTP request to execute an orchestrator function
Durable Functions activity
A function that will be run whenever an Activity is called by an orchestrator function.
Durable Functions entity
Durable Functions entity that stores state.
Durable Functions orchestrator
An orchestrator function that invokes activity functions in a sequence.
Kafka output
A function that will send messages to a specified Kafka topic
Kafka trigger
A function that will be run whenever a message is added to a specified Kafka topic
RabbitMQ trigger
A function that will be run whenever a message is added to a specified RabbitMQ queue
Template details
We need more information to create the Azure Event Hub trigger function. Learn more
New Function *
EventHubTrigger2
Event Hub connection *
temperaturesensor1_events_IOTHUB
New
Event Hub name *
samples-workitems
Event Hub consumer group
$Default

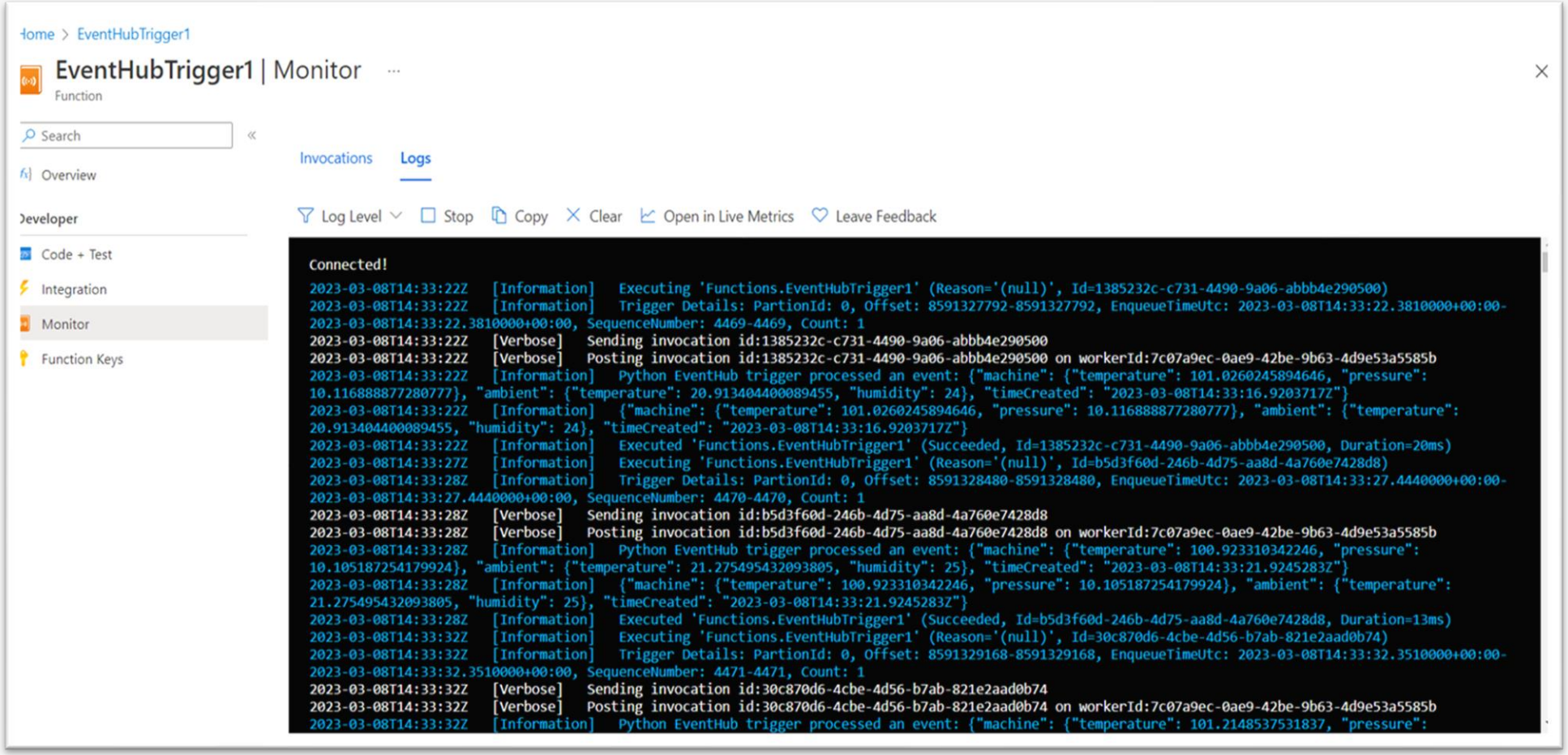
Code

Here , inside the function , we can code as we want and fetch the results from any kind of unstructured data.



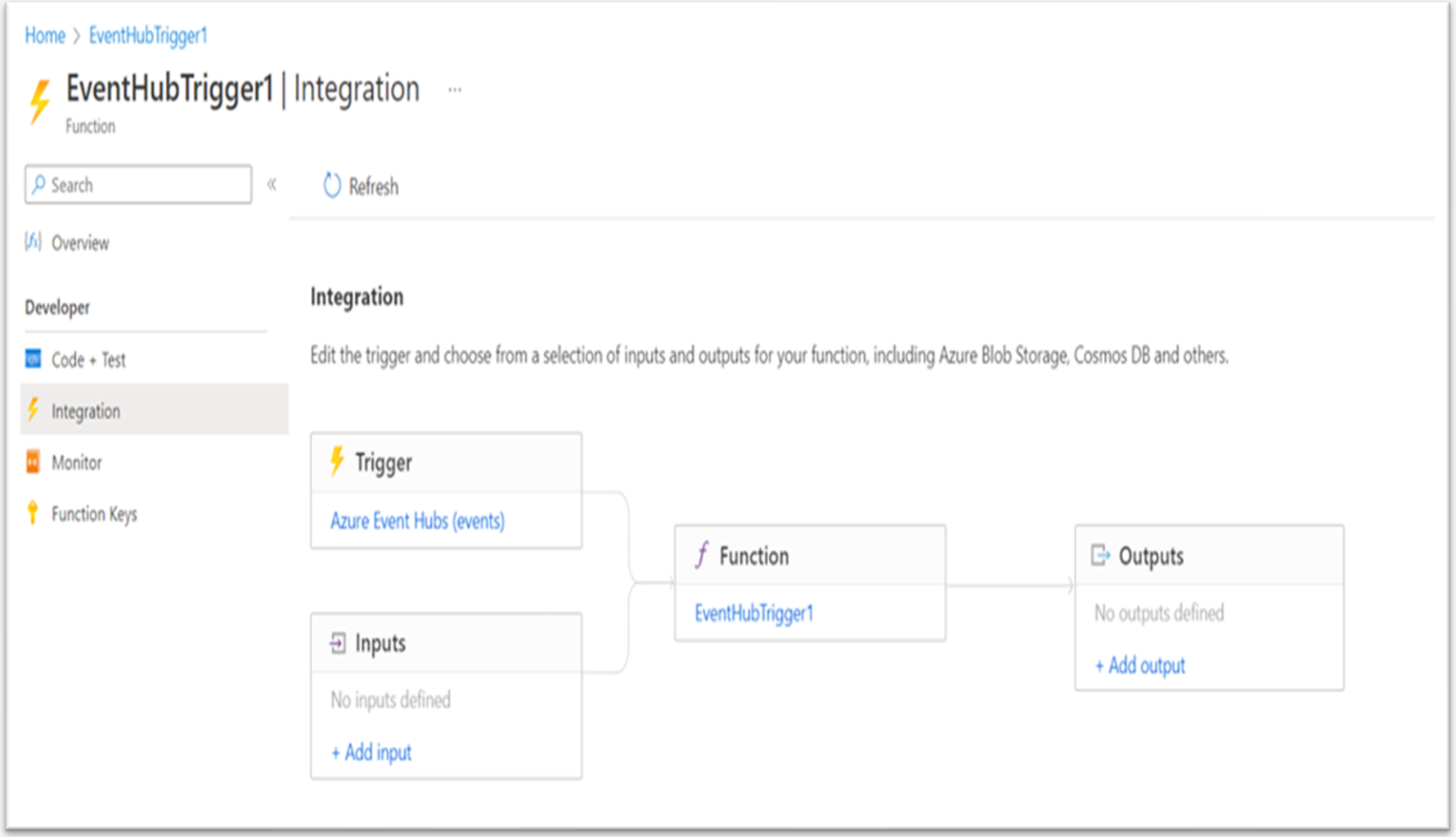
Data coming ...

Here, as you can see, we are getting the data from the temperature sensor per 5 seconds ,which is a live stream. We can write code and fetch the results as needed per message coming and thus can append that results to various targets like dbms , powerbi , and others.



Integrating with Other Sources

We can integrate this transformed results to various targets such as Azure SQL , PowerBI, data bricks where we can write the machine learning code, analyse or visualise the data from the Integration tab in the Function as shown.



EventHubTrigger1 | Integration

Function

Search

Refresh

Overview

Developer

Code + Test

Integration

Monitor

Function Keys

Integration

Edit the trigger and choose from a selection of inputs and outputs for your function, including Azure Blob Storage, Cosmos DB and others.

THANK YOU!