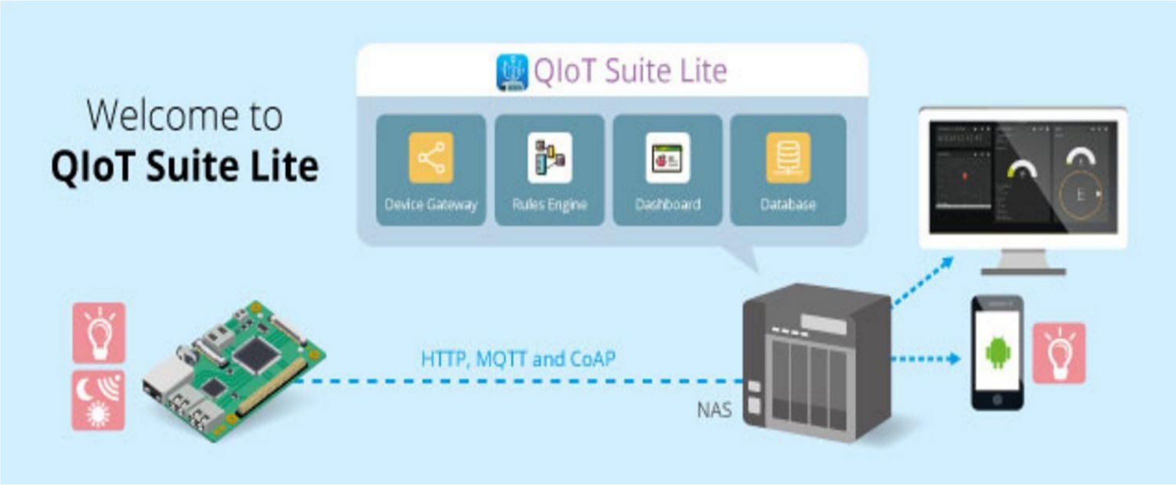
Get started with Intel Edison (Nodejs)

In this tutorial, you begin by learning the basics of working with Intel Edison that's running Linux OS based on Yocto. You then learn how to seamlessly connect your devices to QNAP NAS by using QIoT Suite Lite.
Please ensure your Intel Edison and NAS is under the same LAN.



Lesson 1: Configure your device

In this lesson, you configure your Intel Edison device with an operating system, set up your development environment, and deploy an application to Intel Edison.

1.1 Download Intel Edison drivers

Download Intel Edison drivers based on your host PC, available here:

- Windows users
- Mac users
- Linux users

1.2 Configure and test your device

- Get started with Intel Edison
  If this is the first time you use your Intel Edison board, you will have to follow some steps to assemble it.

  Please follow steps 1-3 of the instructions.

  - Step 1: Assemble boards and sensors.
  - Step 2: Run setup tools.

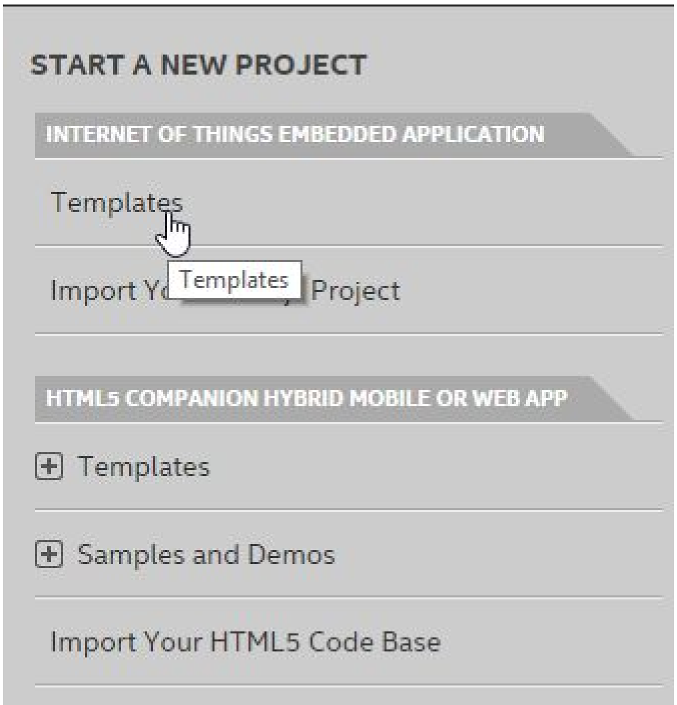
    Use the setup tool to flash the latest firmware on the Intel® Edison development board via a convenient wizard. The setup tool also lets you enable SSH and Wi-Fi* connectivity to your board, as described in the steps to follow.
  - Step 3: Download development environment.

    Download Intel® **XDK** cross-platform integrated development environment

- Verify and upload your onBoardBlink project to Intel Edison

  - Creating your project

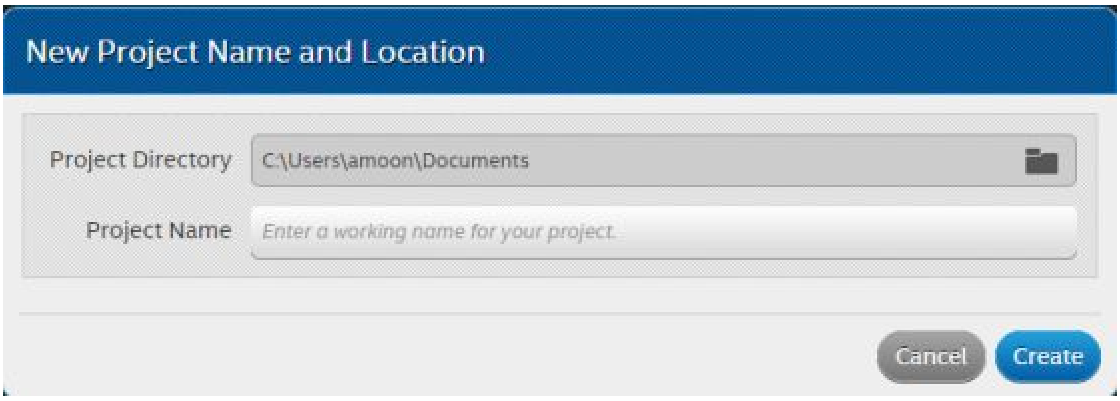
    1. Launch the Intel XDK.
    2. Follow the on-screen instructions to log in to your Intel® XDK account or sign up for a new Intel XDK account.
    3. From the Projects tab, click **Start a New Project** in the bottom left. The Start a New Project page opens.
    4. In the Internet of Things Embedded Application list on the left, click **Templates**. A list of templates are displayed on the right.


START A NEW PROJECT
INTERNET OF THINGS EMBEDDED APPLICATION
Templates
Templates
Import Yo
Project
HTML5 COMPANION HYBRID MOBILE OR WEB APP
Templates
Samples and Demos
Import Your HTML5 Code Base

5. Select the **Onboard LED Blink** template, then click **Continue**.

6. If desired, type or browse for the folder you want to store your projects in the **Project Directory** field.

7. In the **Project Name** field, type a name for your project.



New Project Name and Location

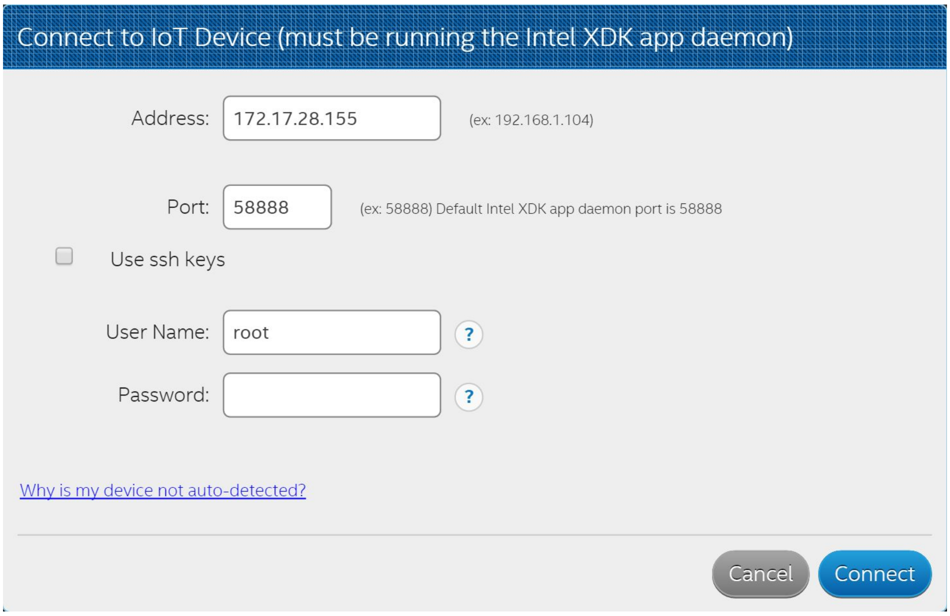
Project Directory C:\Users\amoon\Documents

Project Name *Enter a working name for your project.*

8. Click **Create**. Your project is created.

- Connecting to the board

1. From the **IoT Device** drop-down list in the bottom left, select your development board. The Connect to your IoT Device dialog box opens.



Connect to IoT Device (must be running the Intel XDK app daemon)

Address: 172.17.28.155 (ex: 192.168.1.104)

Port: 58888 (ex: 58888) Default Intel XDK app daemon port is 58888

☐ Use ssh keys

User Name: root

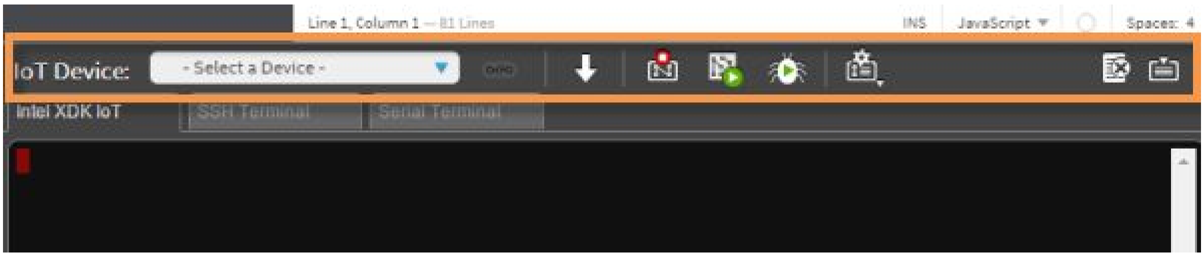
Password:

If your development board is not included in the drop-down list, you must connect to your board manually. Select **Add Manual Connection** and supply your board's IP address in the **Address** field.

2. If you have created a user name and password to log in to your board, type them in the **User Name** and **Password** fields. Otherwise, leave the default values.
3. Click **Connect**.
4. If you see a message saying that your daemon is out of date, update the daemon. This will disconnect from the board, as the daemon has to be restarted, so follow the instructions to reconnect when prompted.
5. If you see a message about the clock on your board and the Intel XDK IoT Edition being out of sync, click **Sync**.
6. A confirmation message appears, displaying the connection status and IP address of your development board. Click **Dismiss**. Your board is now connected.

- Upload and run the project

1. From the top of the application window, click the **Develop** tab. Look for the IoT toolbar at the bottom of the window.



2. Click the **Upload** icon ⬇ to upload your project to the board.
3. Now click the **Run** icon ▶ to run your code. You should see an LED on your board flashing

on and off. Click the **Stop** icon to stop the LED.

Congratulations, you have run your first IoT app using the Intel XDK IoT Edition!

- More Intel Edison setup guide, please refer to https://software.intel.com/en-us/get-started-edison-windows

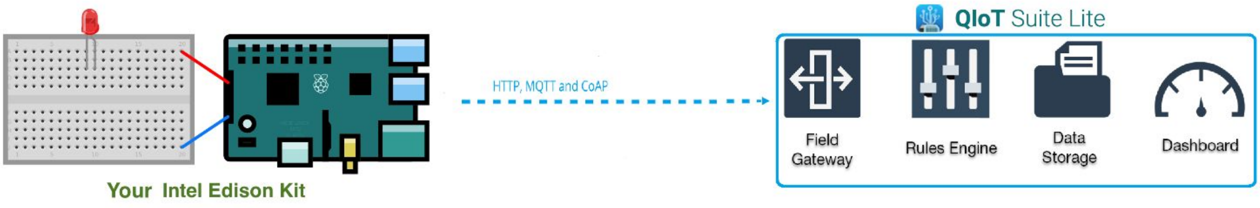
1.3 Install Nodejs and required libraries

- Open a terminal application on your computer and connect to Intel Edison.
- Install the latest nodejs

```
root@Edison:~# opkg update
root@Edison:~# opkg install nodejs
```

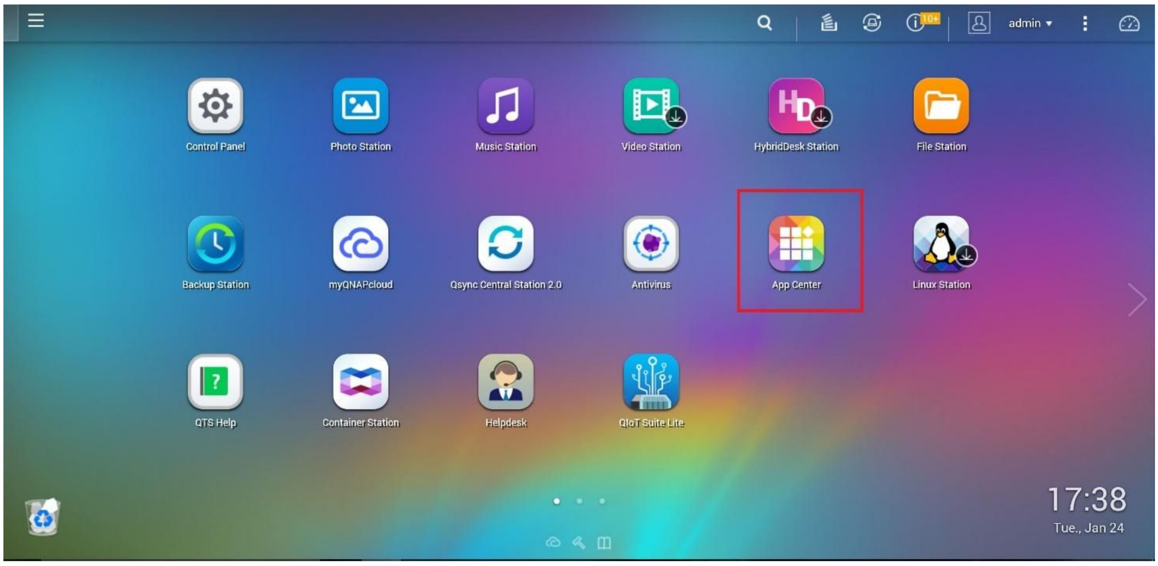
Lesson 2: Create your device in QIoT Suite Lite

In this lesson, you provision your QNAP QIoT Suite Lite software, and create your first device in QIoT Suite Lite.

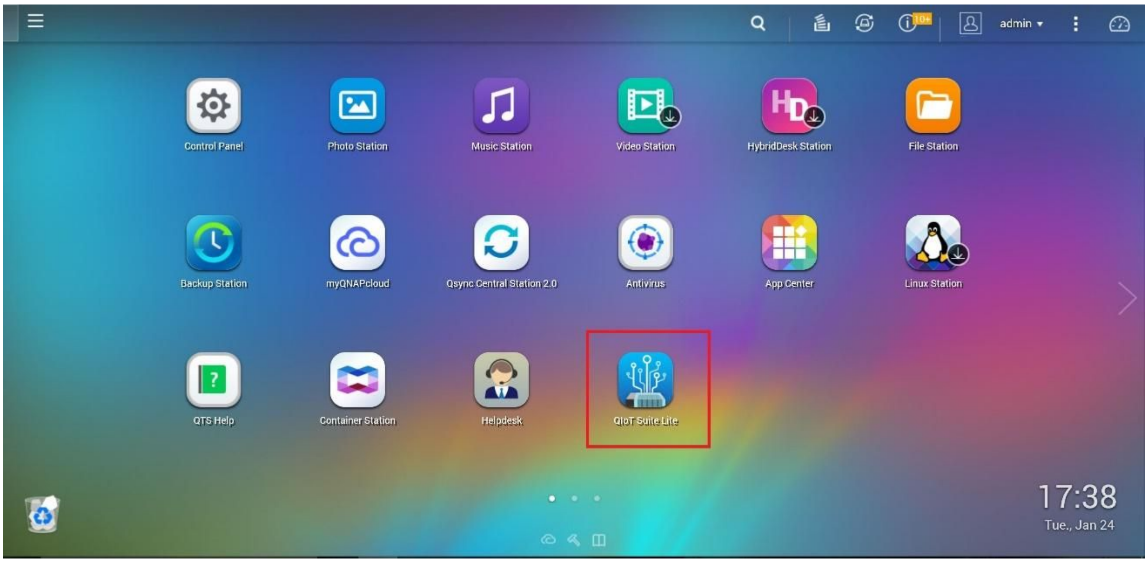


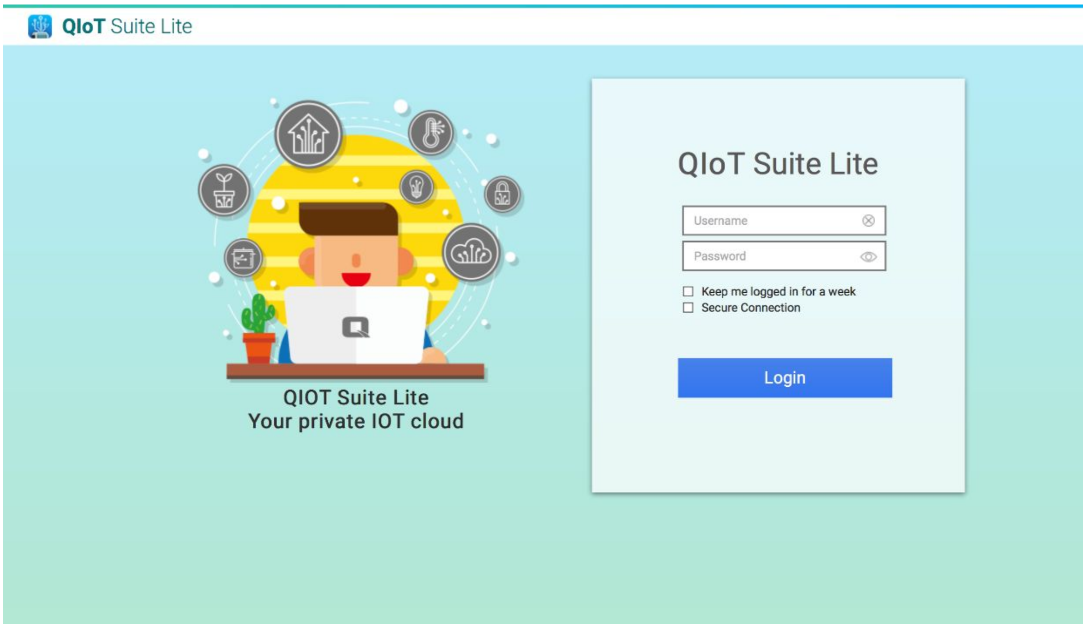
2.1 Install QIoT Suite Lite

- Go to QNAP App Center and download QIoT Suite Lite sofeware.



- Launch and log in QIoT Suite Lite software. Use Nas admin and password to login.





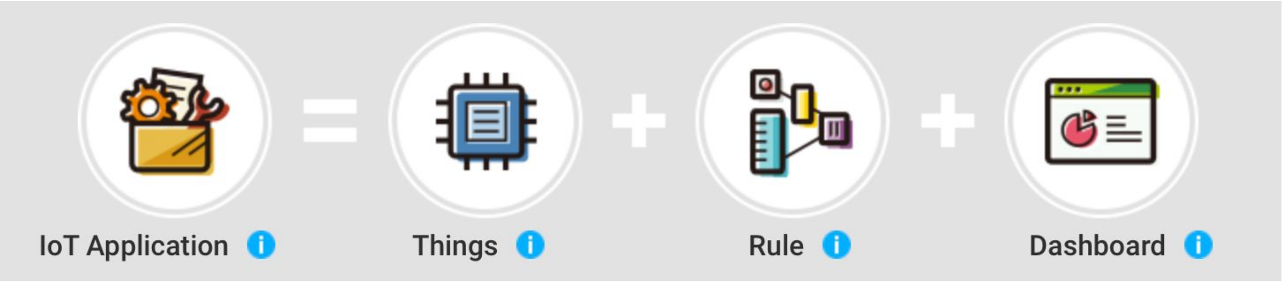
2.2 Create a new IoT application

IoT Application is a combination of multiple Things, Rule, and Dashboard. We recommend that you first create a “Things” in IoT Application. This IoT Application allows you to keep a record of all of the devices that are connected to your NAS.

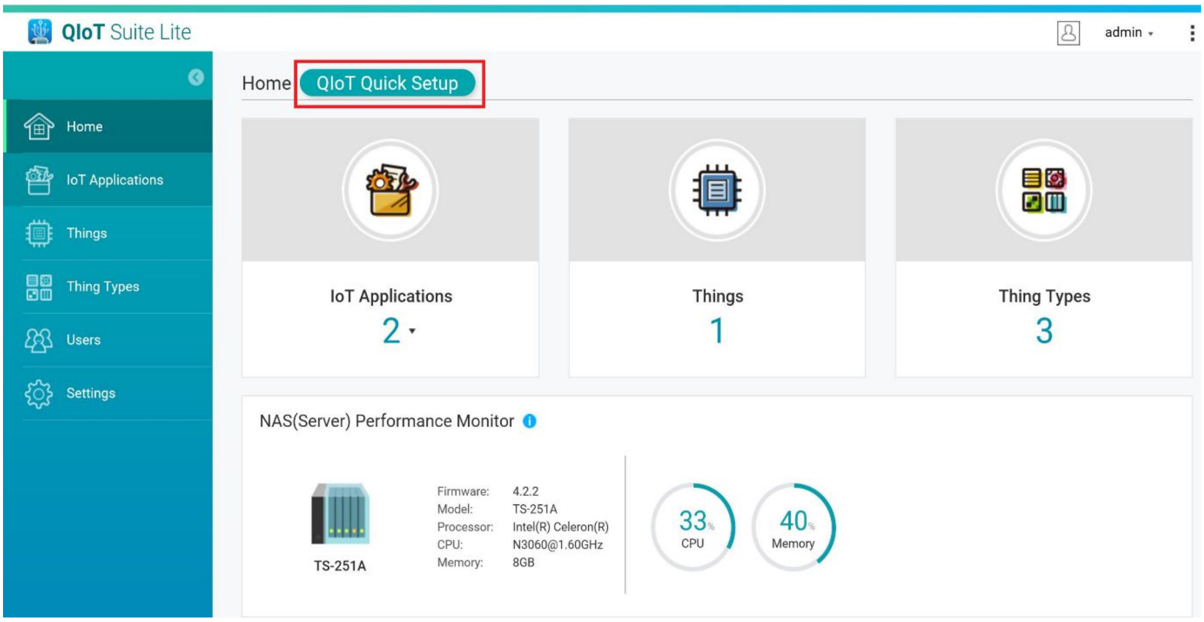
The Rule makes it possible to build IoT applications that gather, process, analyze and act on data generated by connected devices based on business rules you define. A rule can apply to data from one or many devices, and it can take one or many actions in parallel.

With Dashboard, you can turn your data processing efforts into analytics and reports that provide real-time insights into your business.

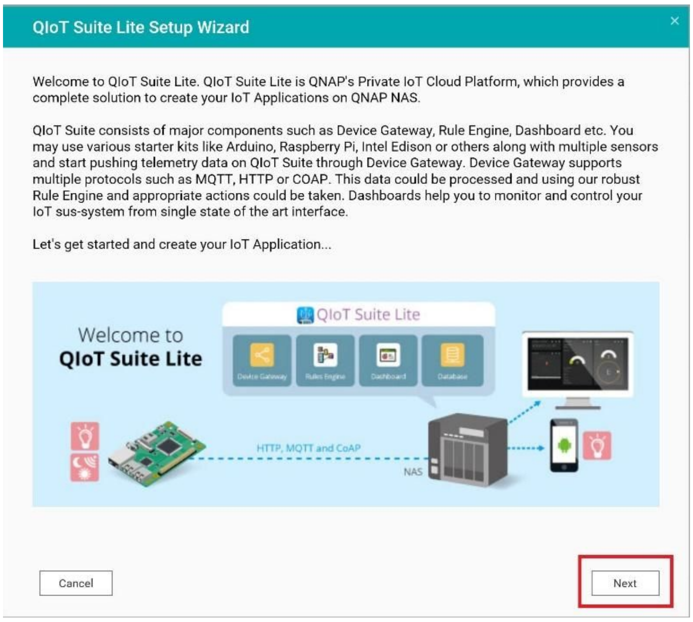
All these elements provide user a complete IoT Application environment.



- If this is your first time to use QIoT Suite Lite, QIoT Suite Lite provides a wizard to help you quick setup a IoT application. Click on **QIoT Quick Setup.** After you read the QIoT Suite Lite introduction, click **Next** button.



Firmware: 4.2.2
Model: TS-251A
Processor: Intel(R) Celeron(R) N3060@1.60GHz
CPU:
Memory: 8GB

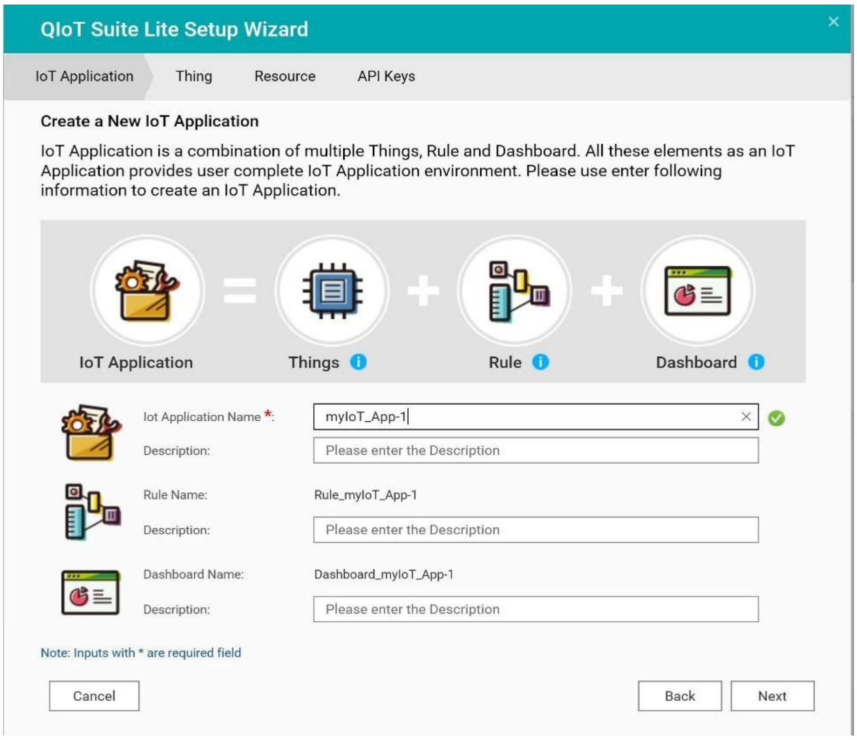

QIoT Suite Lite Setup Wizard

Welcome to QIoT Suite Lite. QIoT Suite Lite is QNAP's Private IoT Cloud Platform, which provides a complete solution to create your IoT Applications on QNAP NAS.

QIoT Suite consists of major components such as Device Gateway, Rule Engine, Dashboard etc. You may use various starter kits like Arduino, Raspberry Pi, Intel Edison or others along with multiple sensors and start pushing telemetry data on QIoT Suite through Device Gateway. Device Gateway supports multiple protocols such as MQTT, HTTP or COAP. This data could be processed and using our robust Rule Engine and appropriate actions could be taken. Dashboards help you to monitor and control your IoT sus-system from single state of the art interface.

Let's get started and create your IoT Application...

- In **IoT Application** tab, type a name for your IoT Application, e.g. myIoT_App-1. Rule name and Dashboard name will be generated automatically based on the name of IoT Application you fill in. Then click **Next** button.



QIoT Suite Lite Setup Wizard

Create a New IoT Application

IoT Application is a combination of multiple Things, Rule and Dashboard. All these elements as an IoT Application provides user complete IoT Application environment. Please use enter following information to create an IoT Application.

Iot Application Name *: myIoT_App-1

Description: Please enter the Description

Rule Name: Rule_myIoT_App-1

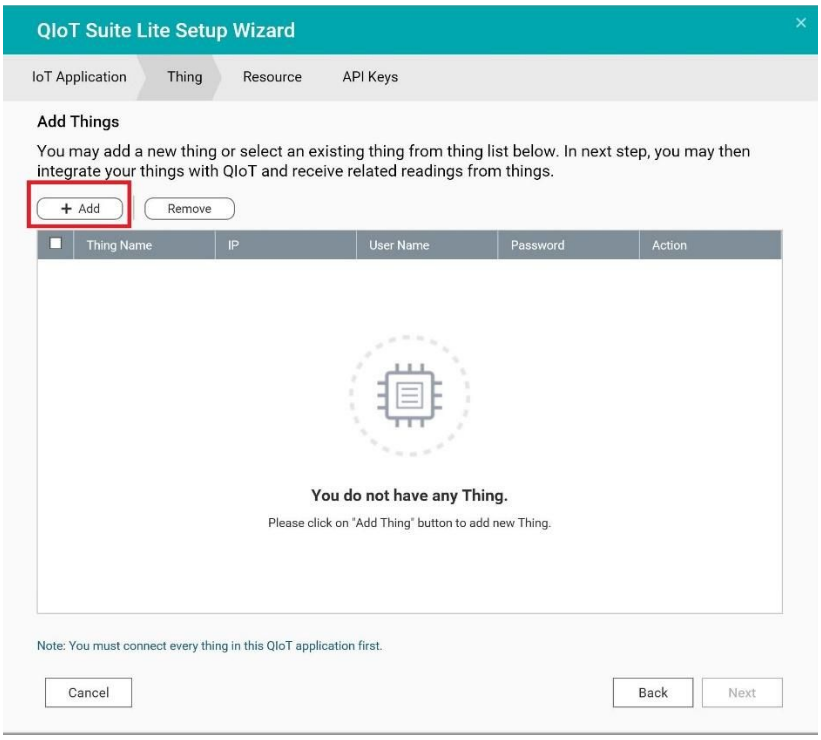
Description: Please enter the Description

Dashboard Name: Dashboard_myIoT_App-1

Description: Please enter the Description

Note: Inputs with * are required field

- In **Thing** tab, click on **Add.** Please provide a name for your device (ex. myThing-1). You can also choose **Add attribute** to provide information about your device (for example, its serial number, manufacturer, and more). If your device is already supported by QIoT software, you can choose **Thing Type** predefined by QNAP. Click on **Add** to add the device.



QIoT Suite Lite Setup Wizard

Add Things

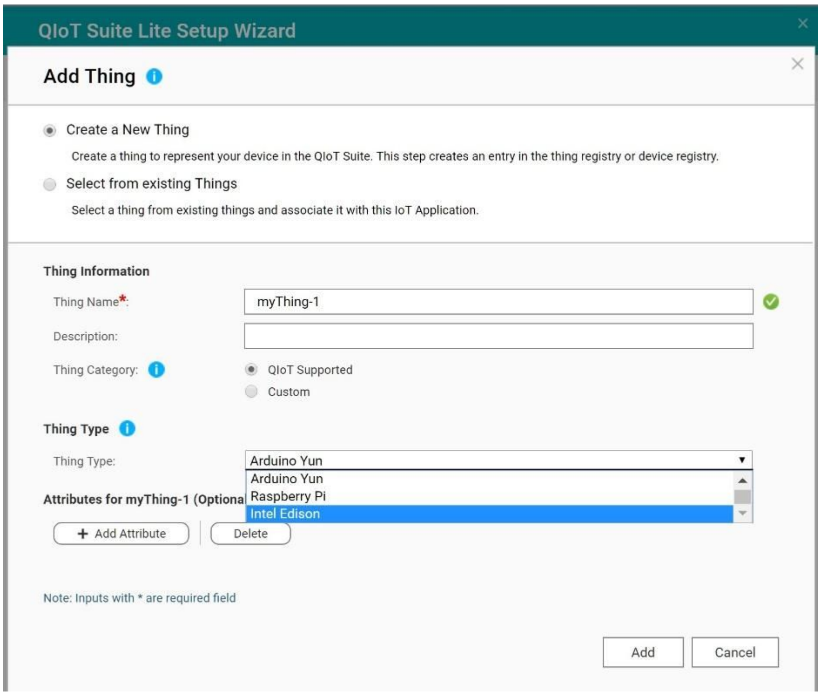
You may add a new thing or select an existing thing from thing list below. In next step, you may then integrate your things with QIoT and receive related readings from things.

□	Thing Name	IP	User Name	Password	Action

You do not have any Thing.

Please click on "Add Thing" button to add new Thing.

Note: You must connect every thing in this QIoT application first.



QIoT Suite Lite Setup Wizard

Add Thing

● Create a New Thing

Create a thing to represent your device in the QIoT Suite. This step creates an entry in the thing registry or device registry.

○ Select from existing Things

Select a thing from existing things and associate it with this IoT Application.

Thing Information

Thing Name*: myThing-1 ✓

Description:

Thing Category: ● QIoT Supported ○ Custom

Thing Type

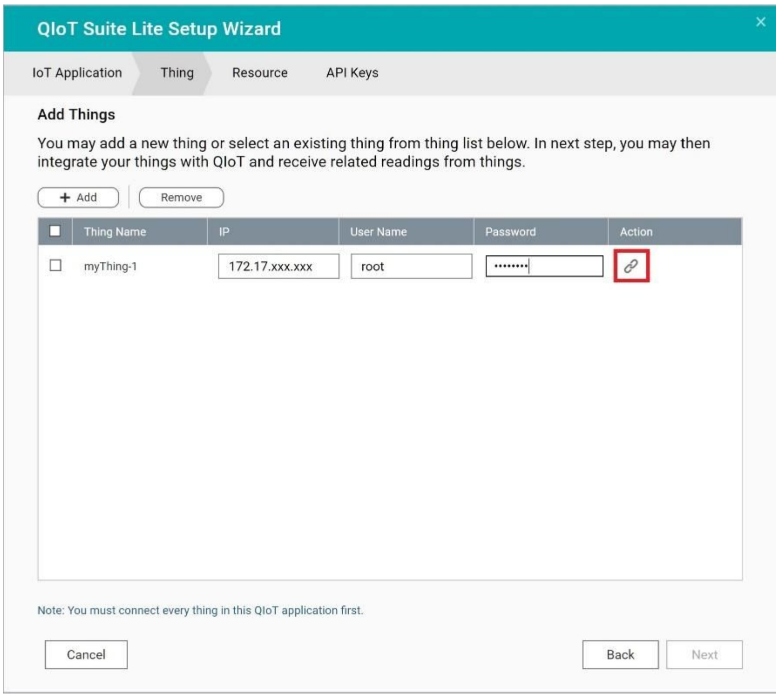
Thing Type: Arduino Yun

Arduino Yun
Raspberry Pi
Intel Edison

Attributes for myThing-1 (Optiona

Note: Inputs with * are required field

Please provide IP address, User Name, and Password of your device, then click Connect icon. QIoT Suite Lite will check the connection with the device. After the connection is verified, please click **Next**.



QIoT Suite Lite Setup Wizard

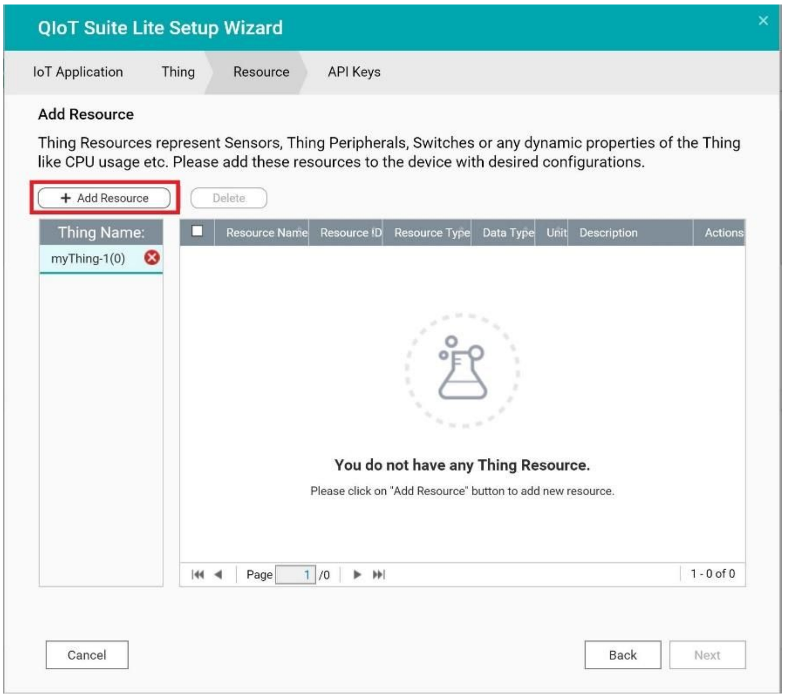
Add Things

You may add a new thing or select an existing thing from thing list below. In next step, you may then integrate your things with QIoT and receive related readings from things.

□	Thing Name	IP	User Name	Password	Action
□	myThing-1	172.17.xxx.xxx	root	••••••••	🔗

Note: You must connect every thing in this QIoT application first.

- In **Resource** tab, click on **Add Resource.** Give name and ID for the sensor you use into **Resource Name and Resource ID**. Please be noted that Resource ID will be used to create a Topic in the QIOT Broker. This ID should be unique for the device and no duplicates should be allowed for the same device.



Add Resource

Thing Resources represent Sensors, Thing Peripherals, Switches or any dynamic properties of the Thing like CPU usage etc. Please add these resources to the device with desired configurations.

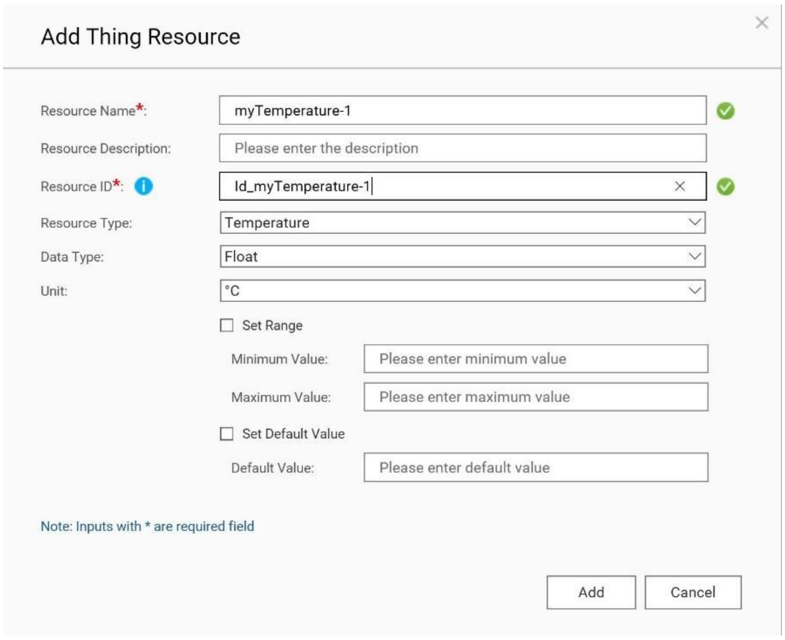
Thing Name:
myThing-1(0)

□	Resource Name	Resource ID	Resource Type	Data Type	Unit	Description	Actions

You do not have any Thing Resource.

Please click on "Add Resource" button to add new resource.

Page 1 /0 1 - 0 of 0

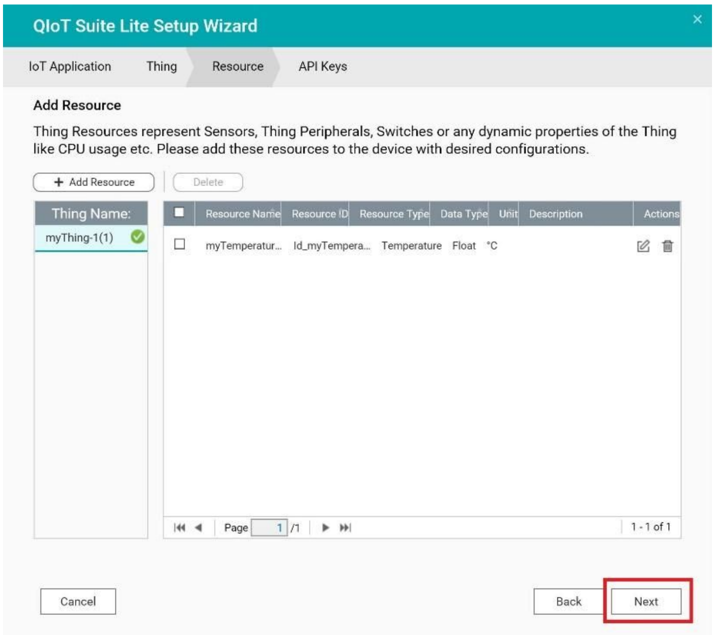


Add Thing Resource

Resource Name*:	myTemperature-1 ✓
Resource Description:	Please enter the description
Resource ID*:	Id_myTemperature-1 ✓
Resource Type:	Temperature
Data Type:	Float
Unit:	°C
	□ Set Range
Minimum Value:	Please enter minimum value
Maximum Value:	Please enter maximum value
	□ Set Default Value
Default Value:	Please enter default value

Note: Inputs with * are required field

Click **Next** after you add all the resources (sensors) on your device.



Add Resource

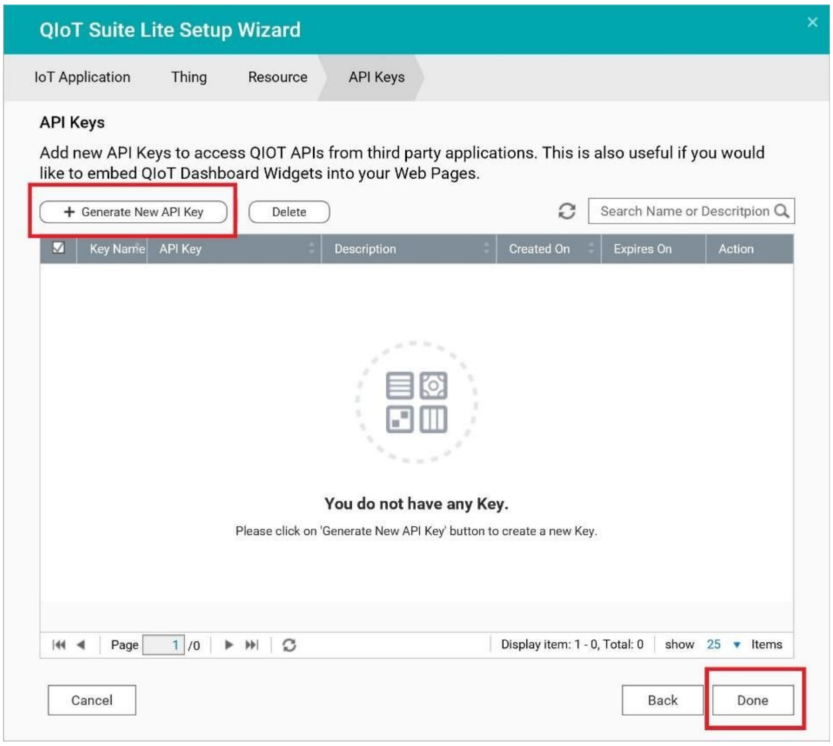
Thing Resources represent Sensors, Thing Peripherals, Switches or any dynamic properties of the Thing like CPU usage etc. Please add these resources to the device with desired configurations.

Thing Name:
myThing-1(1) ✓

□	Resource Name	Resource ID	Resource Type	Data Type	Unit	Description	Actions
□	myTemperatur...	Id_myTempera...	Temperature	Float	°C		

Page 1 /1 1 - 1 of 1

- In API Keys tab, click on **Generate New API Key** if you would like to embed QIoT dashboard widgets into your web pages or access QIoT APIs from third party applications. Click on **Done.**



API Keys

Add new API Keys to access QIOT APIs from third party applications. This is also useful if you would like to embed QIoT Dashboard Widgets into your Web Pages.

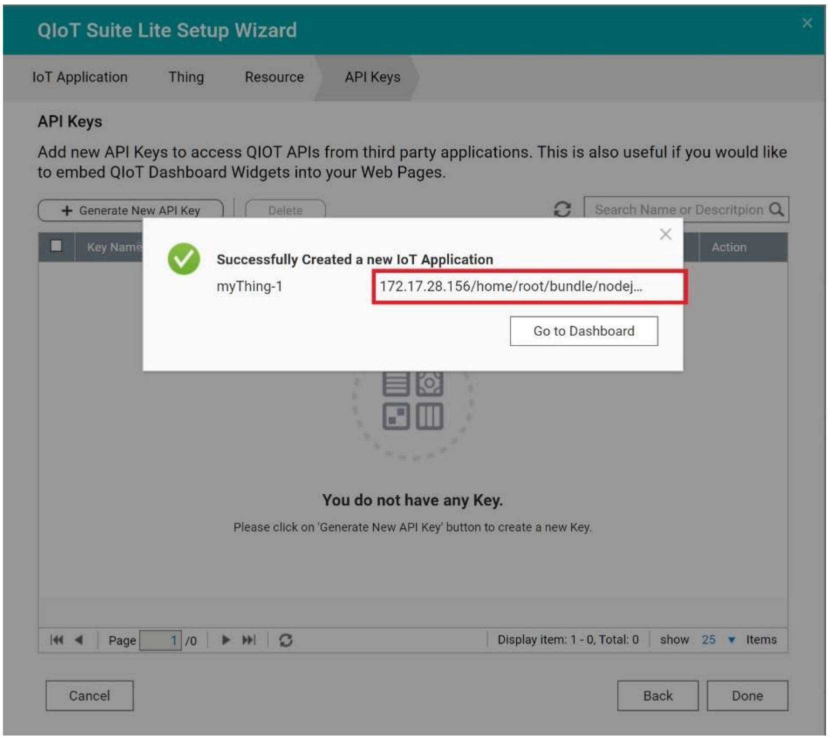
☑	Key Name	API Key	Description	Created On	Expires On	Action

You do not have any Key.

Please click on 'Generate New API Key' button to create a new Key.

Page 1 /0 Display item: 1 - 0, Total: 0 show 25 Items

Your IoT application is created successfully. Node.js sample codes and related files (certificate, resource information) have been upload to the specified path on your device.



QIoT Suite Lite Setup Wizard

IoT Application | Thing | Resource | API Keys

API Keys

Add new API Keys to access QIOT APIs from third party applications. This is also useful if you would like to embed QIoT Dashboard Widgets into your Web Pages.

Successfully Created a new IoT Application

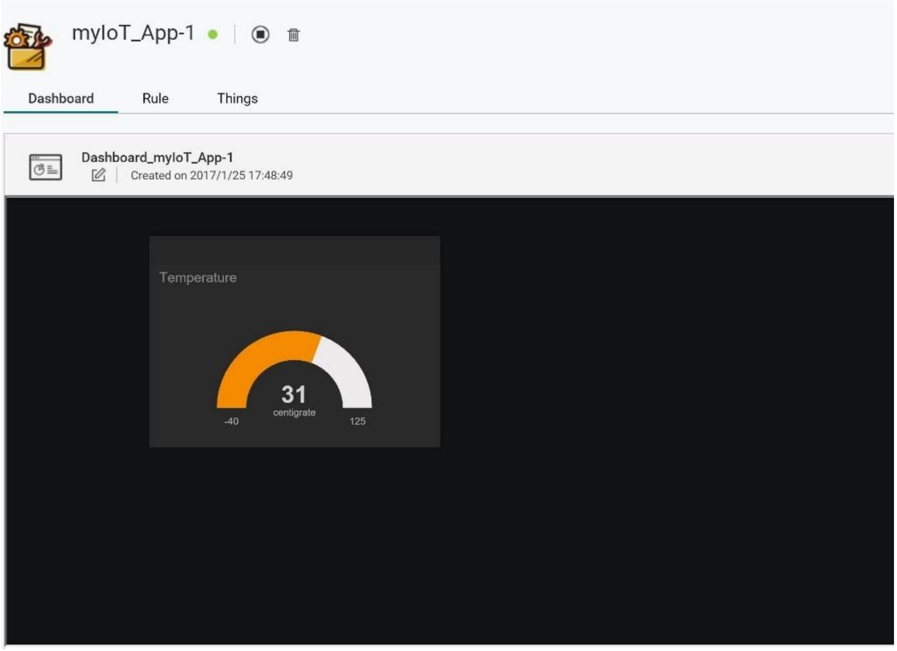
myThing-1 172.17.28.156/home/root/bundle/nodej...

You do not have any Key.

Please click on 'Generate New API Key' button to create a new Key.

Display item: 1 - 0, Total: 0 show 25 Items

- A sample dashboard is created.

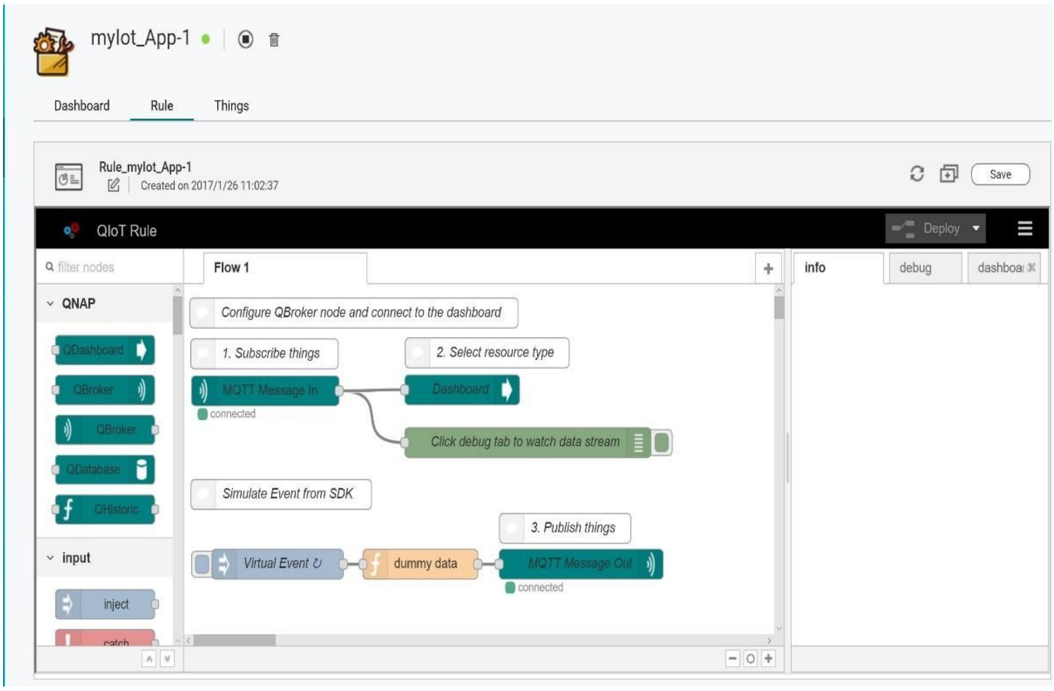


myIoT_App-1

Dashboard_myIoT_App-1

Created on 2017/1/25 17:48:49

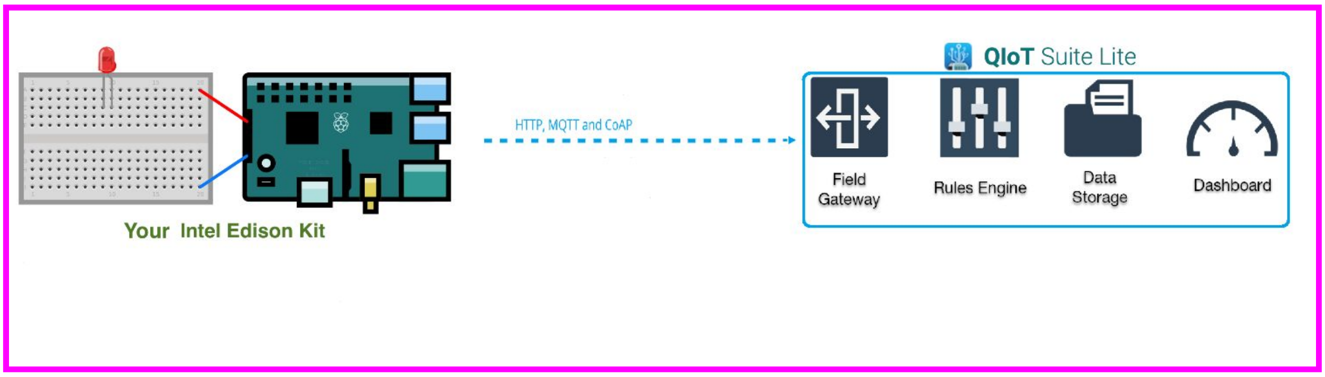
You may go to **Rule** tab to define the flow or rule about how to process the data sent from the device, and how to present in dashboard.



mylot_App-1
Dashboard
Rule
Things
Rule_mylot_App-1
Created on 2017/1/26 11:02:37
Save
QIoT Rule
Deploy
filter nodes
QNAP
QDashboard
QBroker
QBroker
QDatabase
QHistoric
input
inject
catch
Flow 1
Configure QBroker node and connect to the dashboard
1. Subscribe things
2. Select resource type
MQTT Message In
connected
Dashboard
Click debug tab to watch data stream
Simulate Event from SDK
3. Publish things
Virtual Event
dummy data
MQTT Message Out
connected
info
debug
dashboar

Lesson 3: Connect your device to QIoT Suite Lite using MQTTS

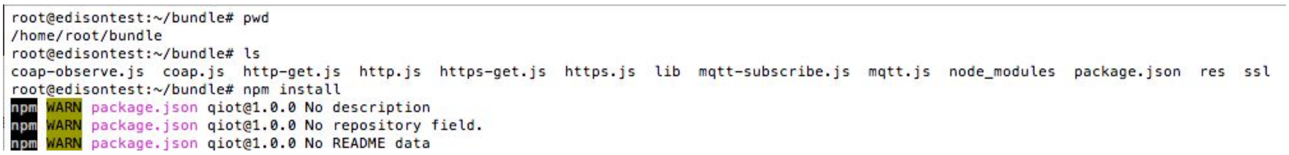
In this lesson, you generate certificate from QIoT Suite Lite, download SDK, and connect Intel Edison to QIoT Suite Lite.



3.1 Connect to QIoT Suite.

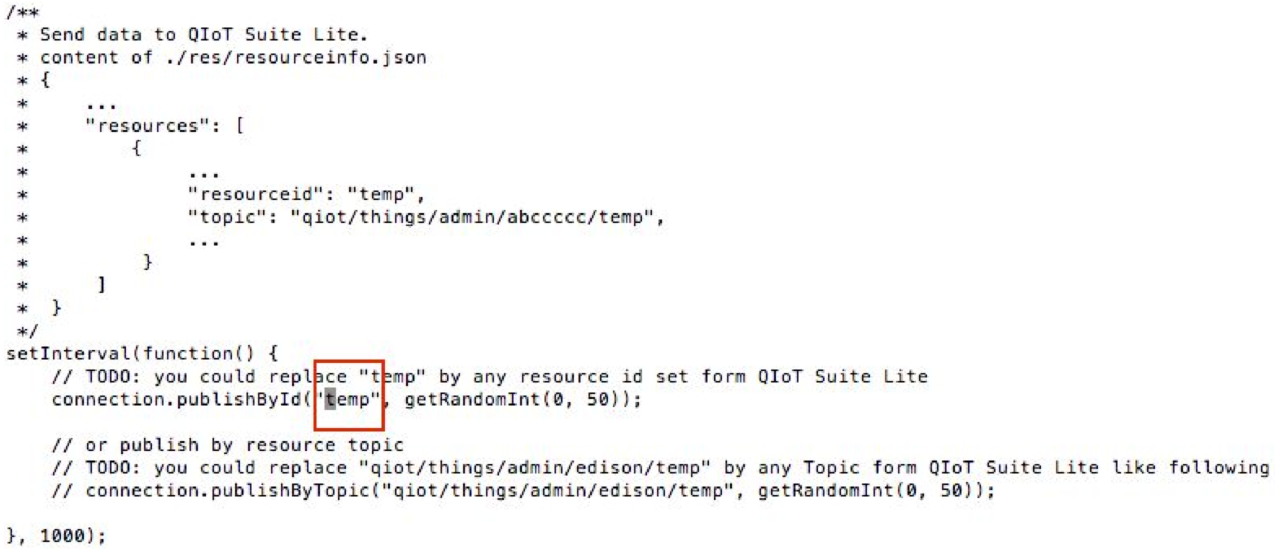
- On your Intel Edison, open Terminal application.
- Go to the directory where QIoT Suite Lite uploads sample codes to.

```
root@Edison:~ $ cd /home/{{user}}/bundle
root@Edison:~ $ npm install
```



```
root@edisontest:~/bundle# pwd
/home/root/bundle
root@edisontest:~/bundle# ls
coap-observe.js  coap.js  http-get.js  http.js  https-get.js  https.js  lib  mqtt-subscribe.js  mqtt.js  node_modules  package.json  res  ssl
root@edisontest:~/bundle# npm install
npm WARN package.json qiot@1.0.0 No description
npm WARN package.json qiot@1.0.0 No repository field.
npm WARN package.json qiot@1.0.0 No README data
```

- Edit the sample code for resourceid (topic name).



```
/**
 * Send data to QIoT Suite Lite.
 * content of ./res/resourceinfo.json
 * {
 *     ...
 *     "resources": [
 *         {
 *             ...
 *             "resourceid": "temp",
 *             "topic": "qiot/things/admin/abccccc/temp",
 *             ...
 *         }
 *     ]
 *  }
 */
setInterval(function() {
    // TODO: you could replace "temp" by any resource id set form QIoT Suite Lite
    connection.publishById("temp", getRandomInt(0, 50));

    // or publish by resource topic
    // TODO: you could replace "qiot/things/admin/edison/temp" by any Topic form QIoT Suite Lite like following
    // connection.publishByTopic("qiot/things/admin/edison/temp", getRandomInt(0, 50));

}, 1000);
```

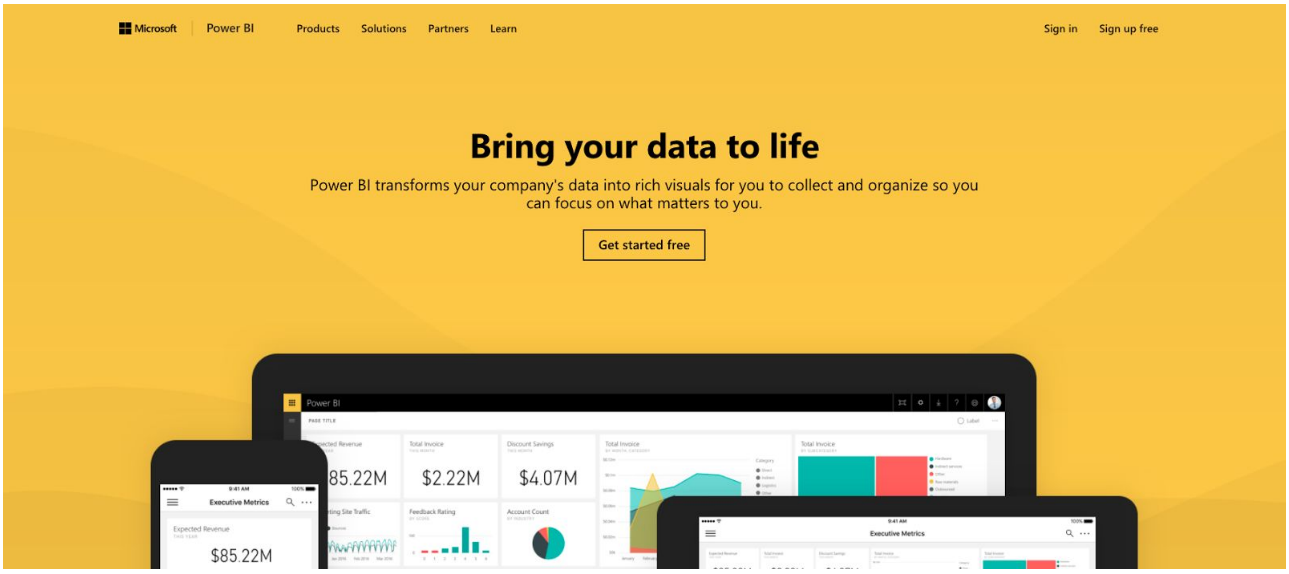
- Run the sample application.

```
root@Edison:~ $ node mqtt.js
```

Lesson 4: Integrate Power BI

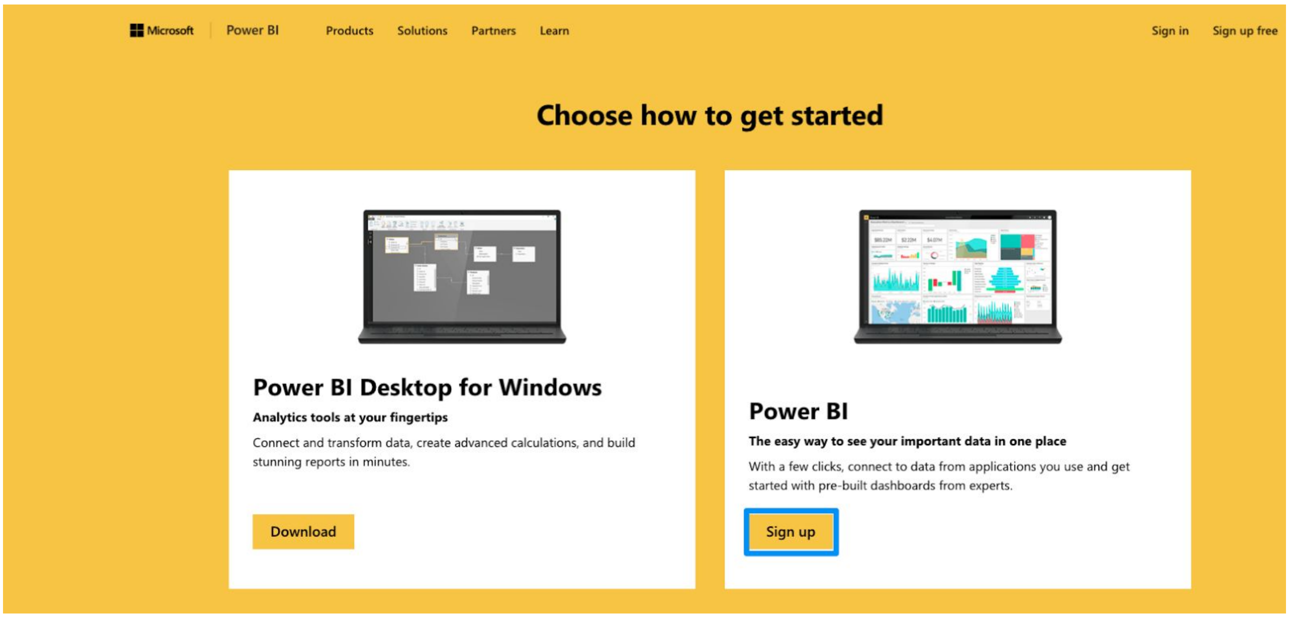
4.1 Get your first Power BI account

- Go to the offical website “https://powerbi.microsoft.com/en-us/” to sign up your free account.



Bring your data to life

Power BI transforms your company's data into rich visuals for you to collect and organize so you can focus on what matters to you.



Choose how to get started

Power BI Desktop for Windows

Analytics tools at your fingertips

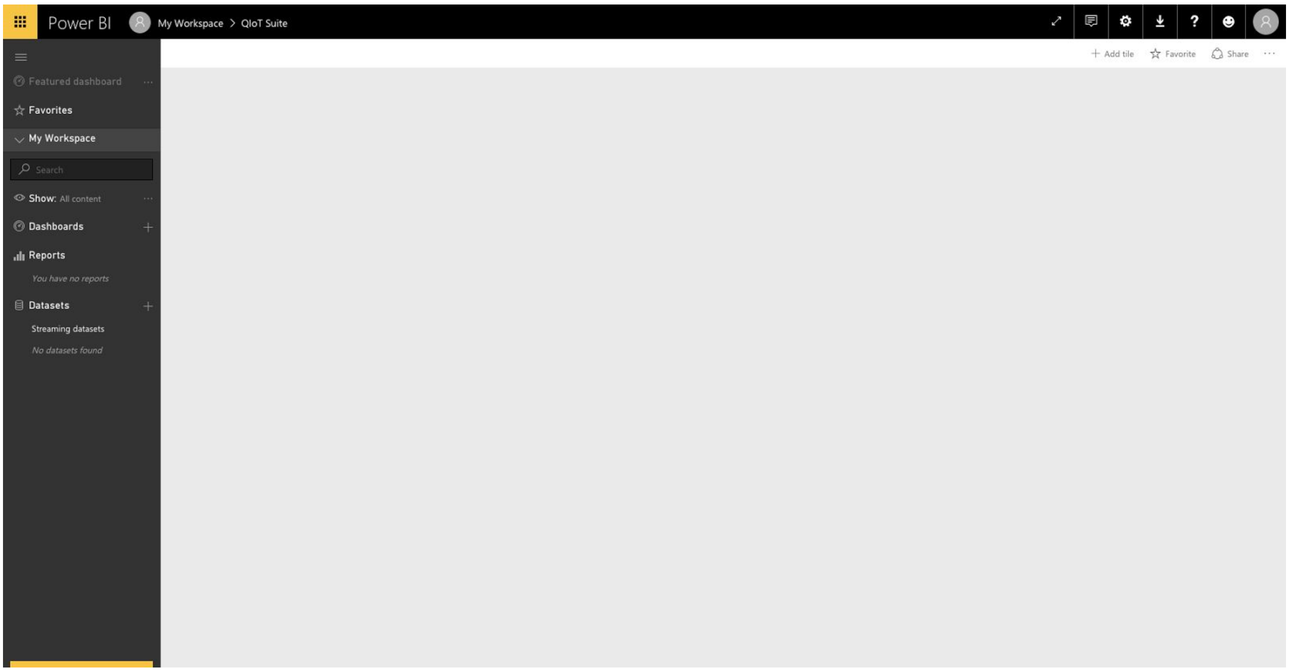
Connect and transform data, create advanced calculations, and build stunning reports in minutes.

Power BI

The easy way to see your important data in one place

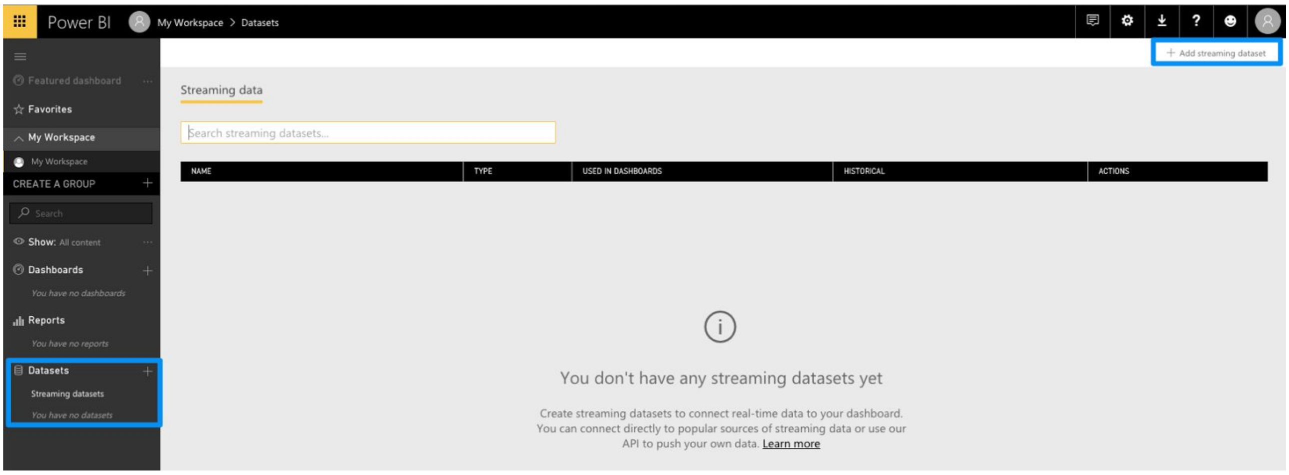
With a few clicks, connect to data from applications you use and get started with pre-built dashboards from experts.

- After a sequence of registration, the page will lead you to below page



4.2 Setup your streaming dataset API

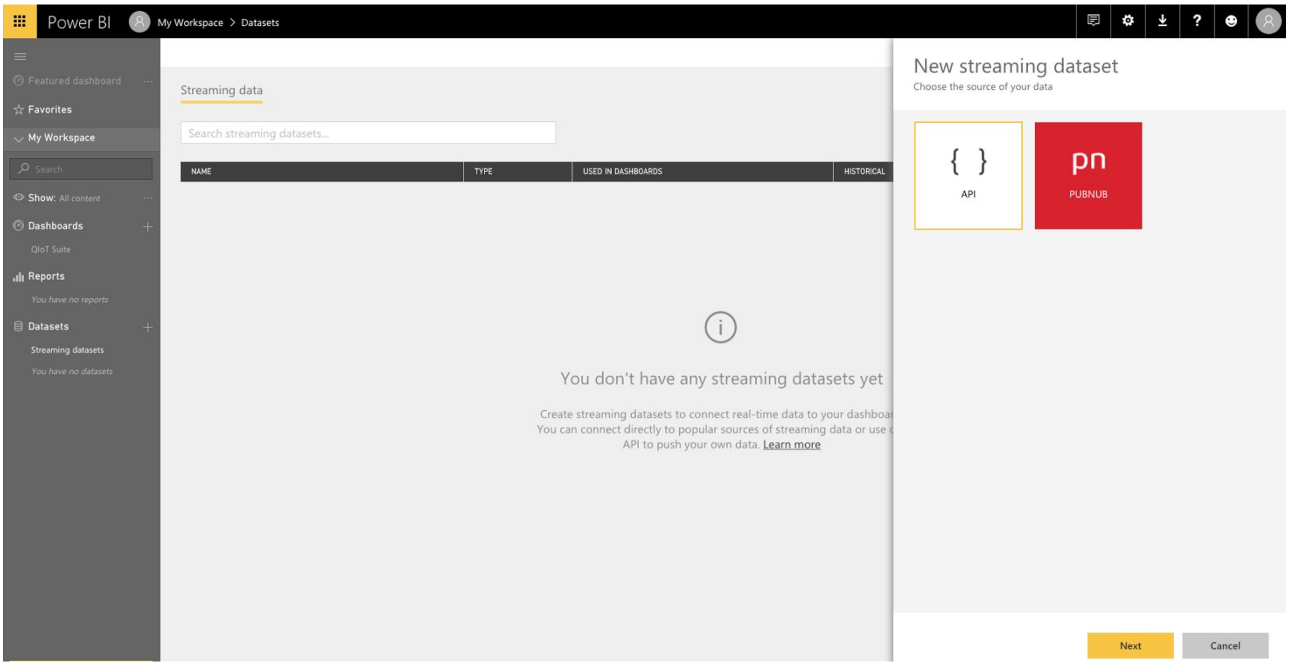
- Go to “**Datasets -> Streaming datasets**” section at left menu, and click the “ + **Add streaming dataset**” button at right-top corner.



You don't have any streaming datasets yet

Create streaming datasets to connect real-time data to your dashboard. You can connect directly to popular sources of streaming data or use our API to push your own data. Learn more

- Select “**API**” as your source of iot data, and click the “**Next**” button.



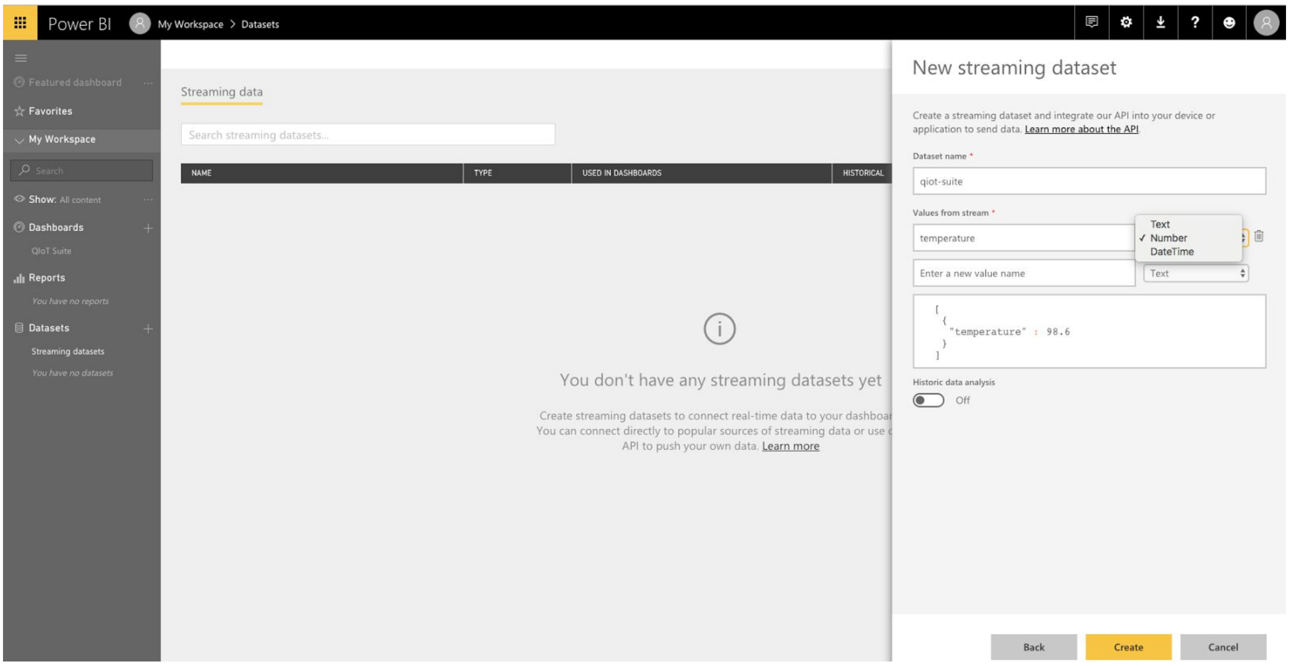
New streaming dataset

Choose the source of your data

You don't have any streaming datasets yet

Create streaming datasets to connect real-time data to your dashboar
You can connect directly to popular sources of streaming data or use
API to push your own data. Learn more

- Define your values from stream, and you will get a result of JSON in textbox. We will use this JSON template later to push data to IoT application. Click the "**Create**" button to finish flow of push API creation.



New streaming dataset

Create a streaming dataset and integrate our API into your device or application to send data. Learn more about the API

Dataset name *

qiot-suite

Values from stream *

temperature | Number

Enter a new value name | Text

```
[
  {
    "temperature" : 98.6
  }
]
```

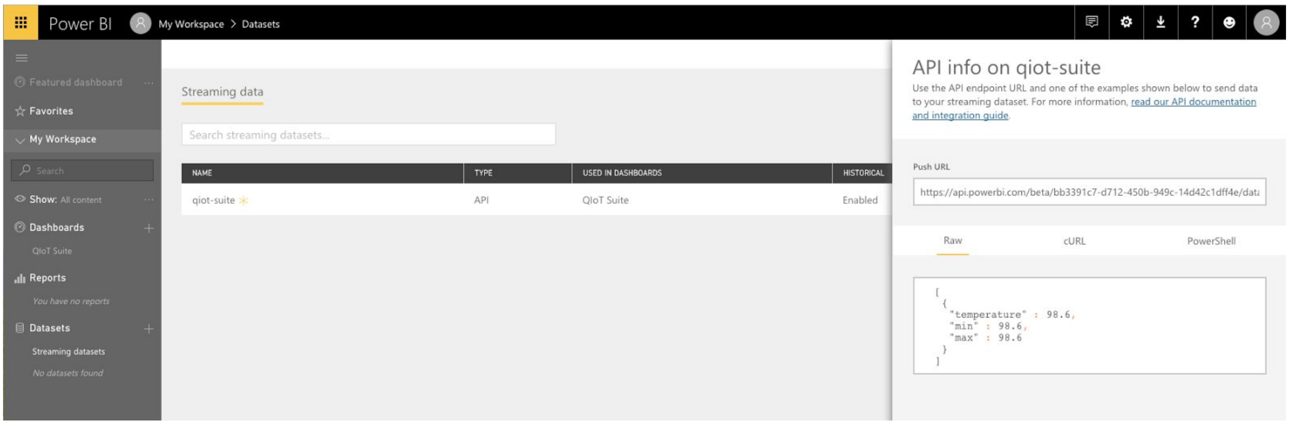
Historic data analysis

Off

You don't have any streaming datasets yet

Create streaming datasets to connect real-time data to your dashboar
You can connect directly to popular sources of streaming data or use
API to push your own data. Learn more

- Once you successfully create your data stream, you get REST API URL which IoT application can call using POST request to push your live data to streaming data dataset you created.



NAME	TYPE	USED IN DASHBOARDS	HISTORICAL
qiot-suite	API	QIoT Suite	Enabled

API info on qiot-suite

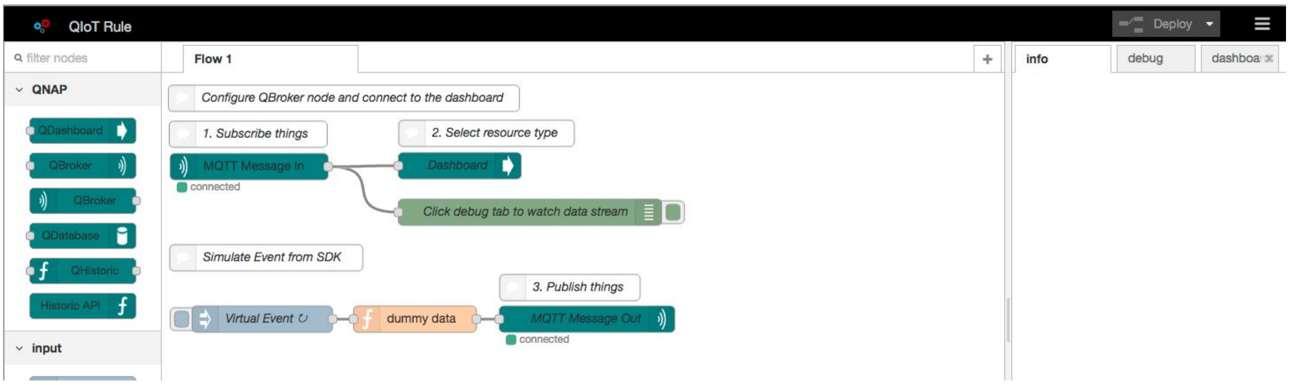
Use the API endpoint URL and one of the examples shown below to send data to your streaming dataset. For more information, read our API documentation and integration guide.

Push URL: https://api.powerbi.com/beta/bb3391c7-d712-450b-949c-14d42c1dff4e/data

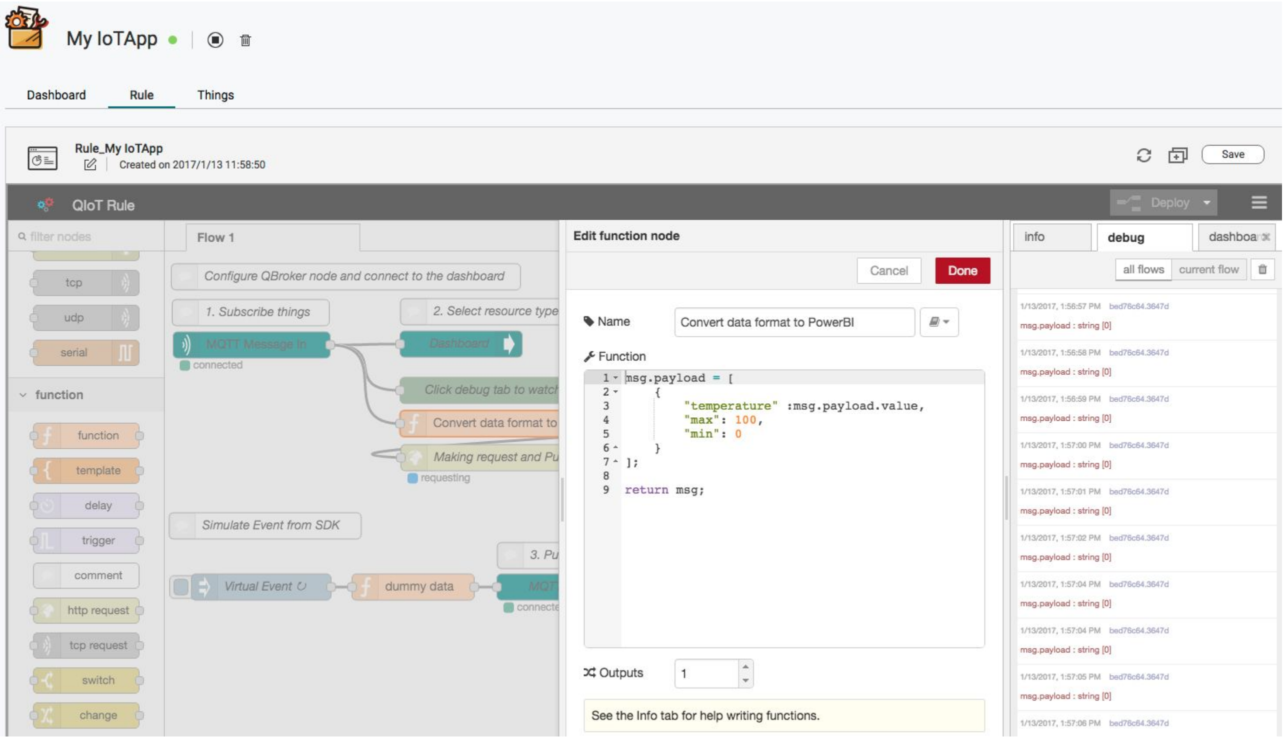
```
[
 {
  "temperature" : 98.6,
  "min" : 98.6,
  "max" : 98.6
 }
]
```

4.3 Configure Node-RED’s nodes in IoT application

- Create IoT application in QIoT suite. The following one is your first node-red flow, and then you can start to create your own IoT flow. more node-red information can be found in “https://nodered.org/”.



- Before you start to push live data to Power BI. We need a “function” node to convert IoT data to streaming data dataset. Here you can replace msg.payload to your JSON dataset.



Name: Convert data format to PowerBI

Function

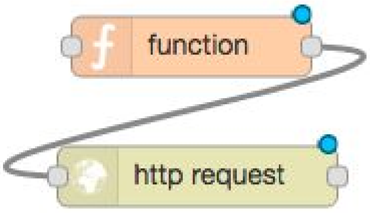
```
msg.payload = [
    {
        "temperature" :msg.payload.value,
        "max": 100,
        "min": 0
    }
];

return msg;
```

Outputs: 1

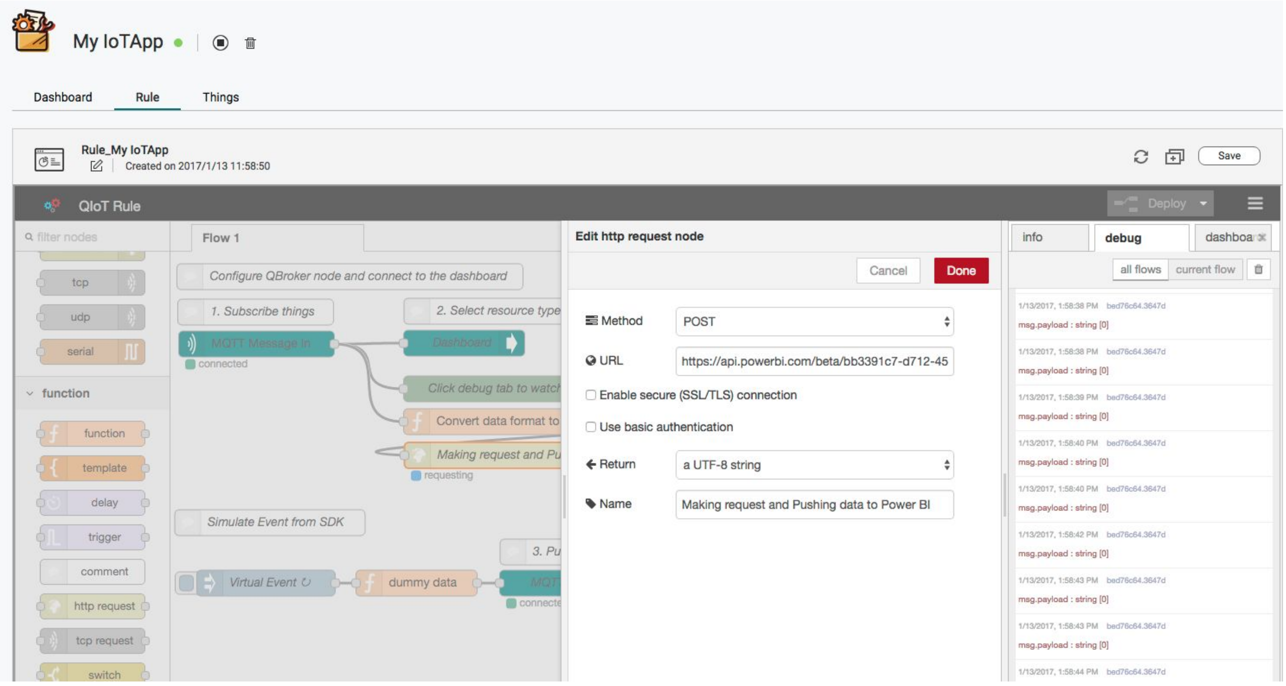
See the Info tab for help writing functions.

```
1/13/2017, 1:56:57 PM   bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:56:58 PM   bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:56:59 PM   bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:57:00 PM   bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:57:01 PM   bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:57:02 PM   bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:57:04 PM   bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:57:04 PM   bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:57:05 PM   bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:57:06 PM   bed76c64.3647d
```

- We need a “http request” node to help us to push live data to Power BI. Just drop and drag “http request” node and connect to tail of “function“ node.



- Copy and paste REST API URL that you got from Power BI console, and set http method to POST. Finally, don’t forget to press “**Deploy**” button to save changes.



Edit http request node

Cancel Done

Method: POST

URL: https://api.powerbi.com/beta/bb3391c7-d712-45

□ Enable secure (SSL/TLS) connection

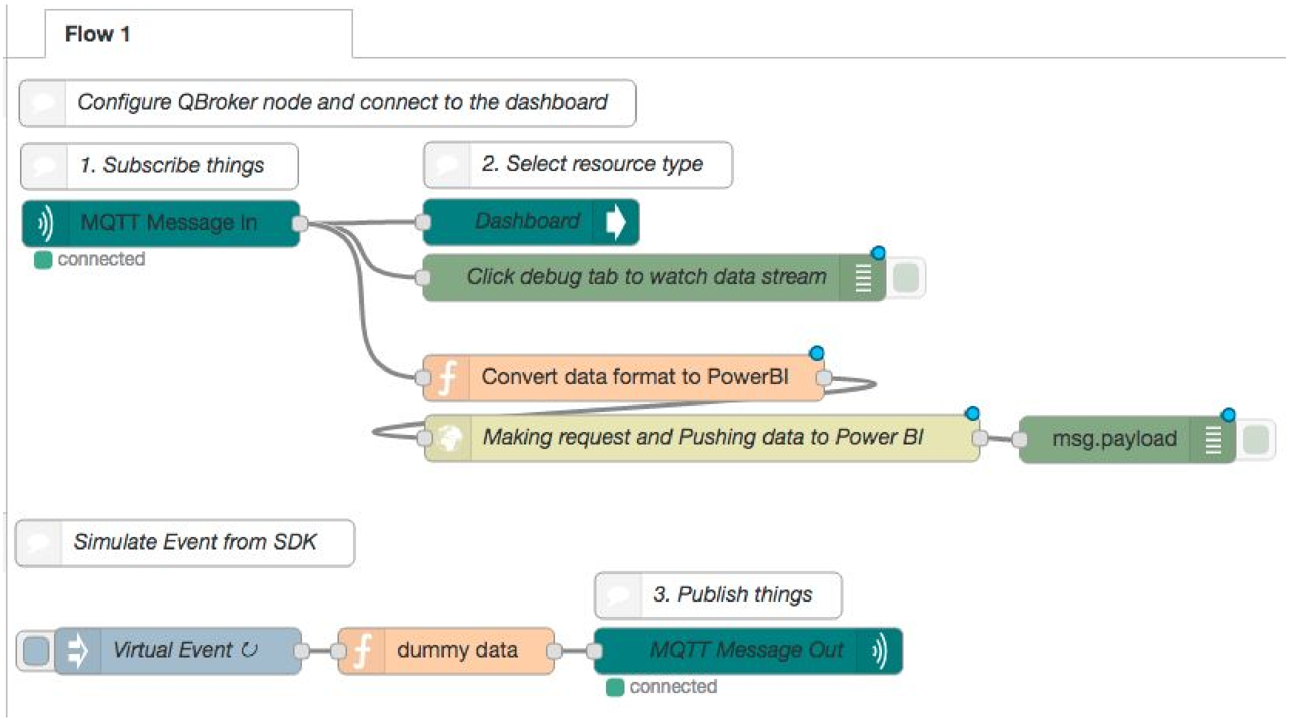
□ Use basic authentication

Return: a UTF-8 string

Name: Making request and Pushing data to Power BI

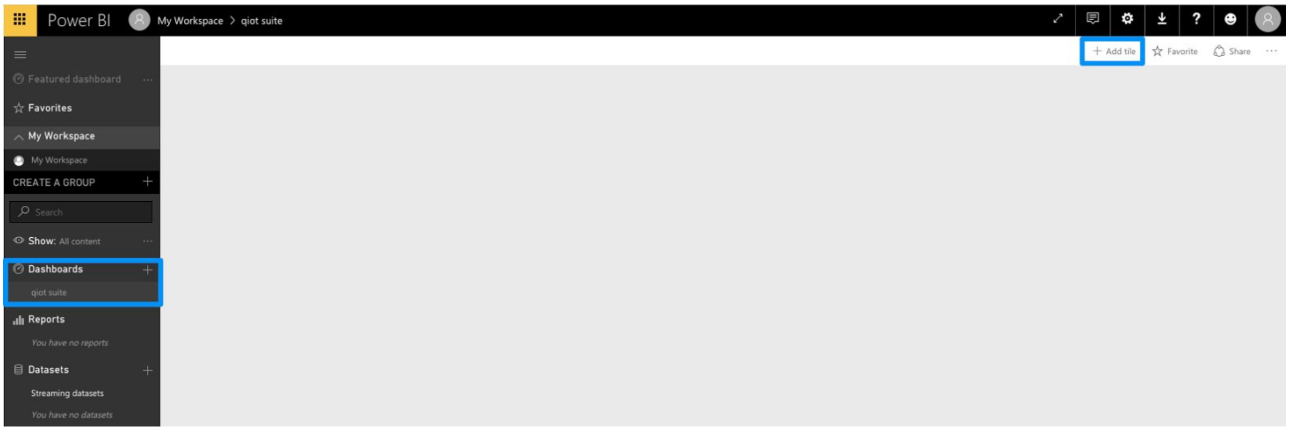
```
1/13/2017, 1:58:38 PM  bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:58:38 PM  bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:58:39 PM  bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:58:40 PM  bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:58:40 PM  bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:58:42 PM  bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:58:43 PM  bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:58:43 PM  bed76c64.3647d
msg.payload : string [0]
1/13/2017, 1:58:44 PM  bed76c64.3647d
```

- Finally, your node-red flow will look like below one.

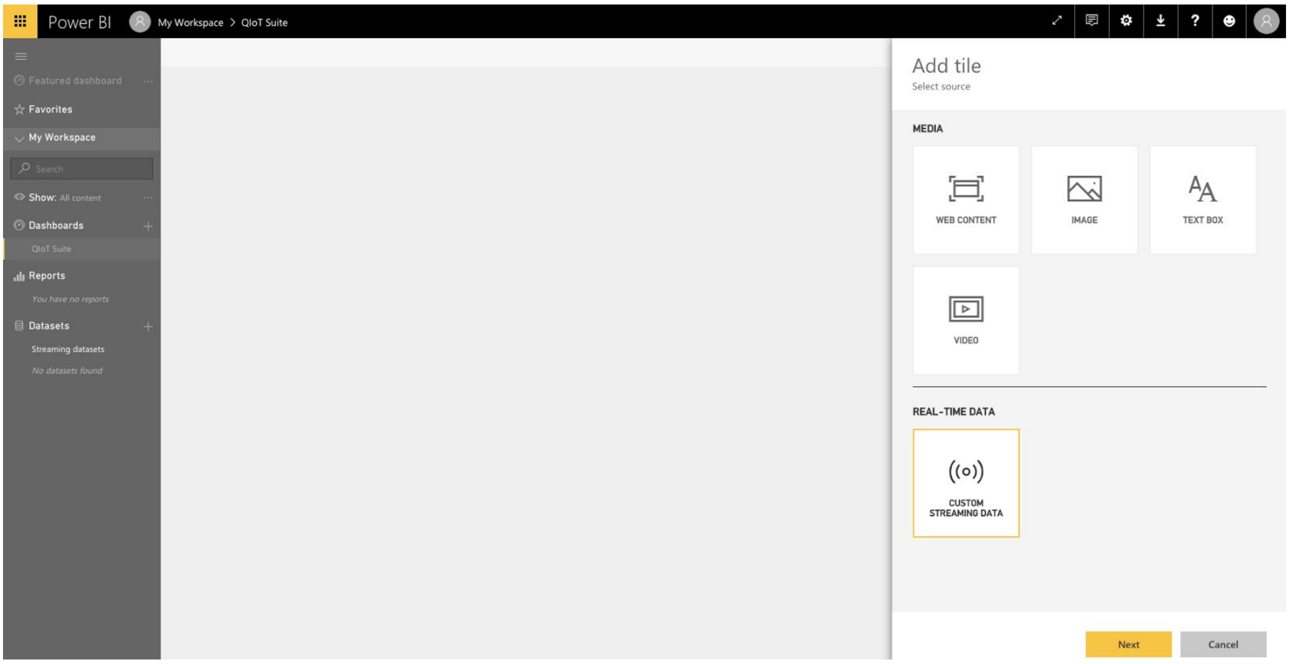


4.4 Add tile to display real-time data

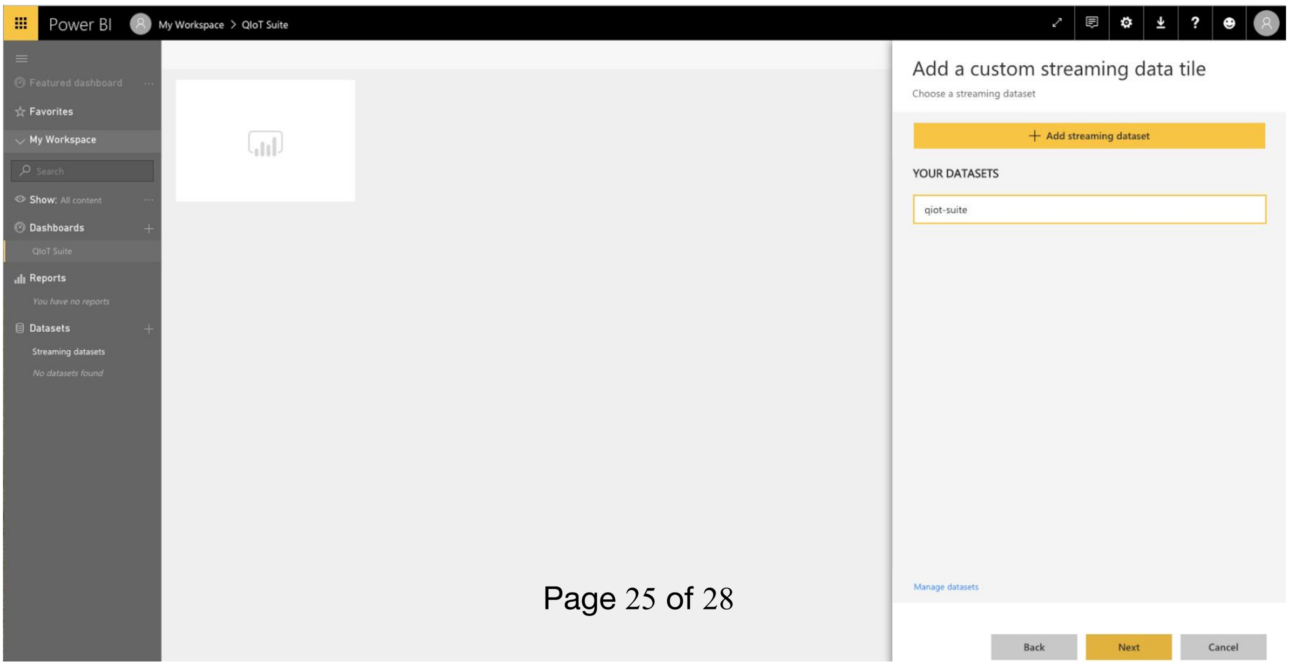
- Go to “**Dashboards +**” to create your first dashboard, then click the “**+ Add tile**” button to configure a widget.



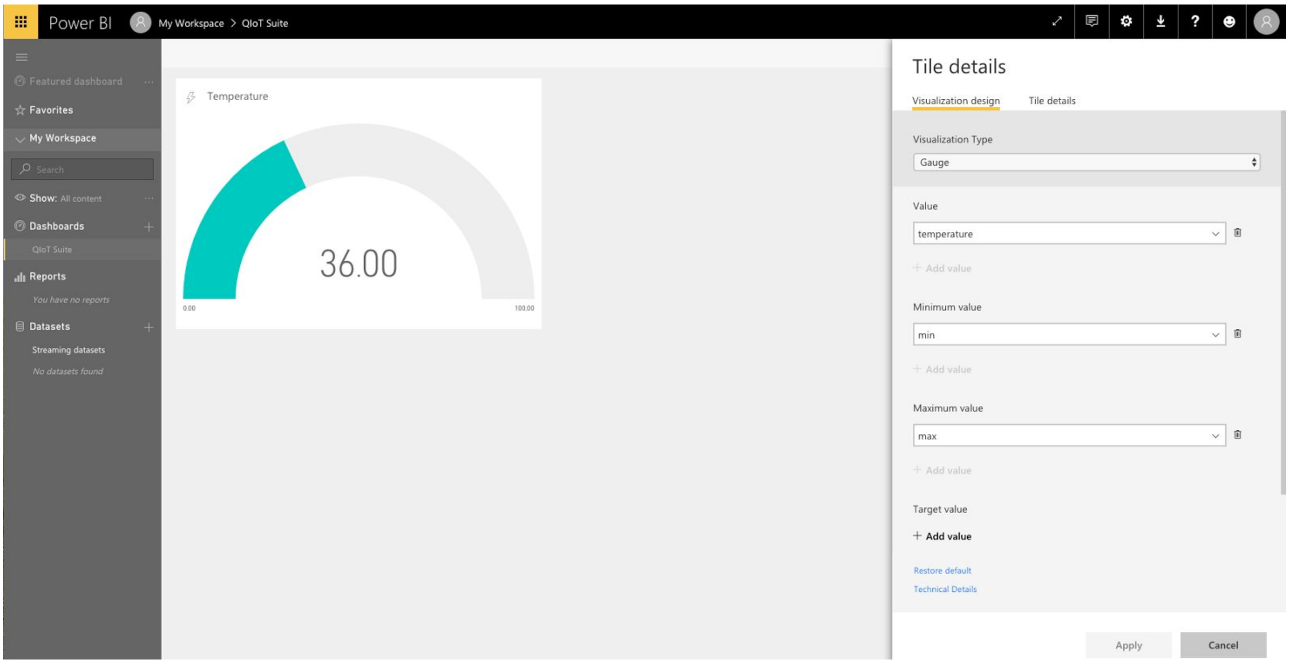
- Select “**CUSTOM STREAMING DATA**” and then select the “**Next**” button.



- Select datasets and then click the “**Next**” button.



- You have a streaming dataset to work with, you can get a real time gauge that looks like as following.



Visualization Type: Gauge

Value: temperature

Minimum value: min

Maximum value: max

Appendix

QNAP QIoT Startkit Sample Code Introduction

- GitHub: Sample Code
- Sample Code Structure

```
qnap-qiot-sdks/
   nodejs/                       # nodejs program language
      device/
         intel-edison/           # intel-edison/raspberrypi...
            examples/
               lib/              # QIoT command Lib
               res/              # QIoT resourceinfo.json folder
               ssl/              # QIoT certificate files folder.
               mqtt.js           # sample code - mqtt/mqtts publish
               http.js           # sample code - http post
               https.js          # sample code - https post
               coap.js           # sample code - coap postt
               mqtt-subscribe.js # sample code - mqtt/mqtts subscribe
               http-get.js       # sample code - http get
               https-get.js      # sample code - https get
               coap-observe.js   # sample code - coap get
              packagae.json      # npm packages document
   python/                       # python program language
      device/
         intel-edison/
            examples/
               lib/
               res/
               ssl/
               mqtt.js
               http.js
               ...
```

- content of resourceinfo.json

protocol	resourceinfo.json content
mqtts	{ "host": ["172.17.28.28" # nas ip], "myqnapcloudHost": "Not Available", # myqnapcloudHost

	"port": 8883, # mqtts port
	"clientId": "adfa_1491561635", # thing Id
	"username": "00477f86-425b-49de-8590-xx", # username
	"password": "r:2825dedfb012969e1dfb6adb8", # password
	"resources": [# resource des
	  {
	    "resourcename": "adf", # resource name
	    "resourceid": "dfadf", # resource id
	    "resourcetypename": "Temperature", # resource type
	    "datatype": "Float", # data type
	    "unit": "°C", # data unit
	    "description": "adfa", # resource des
	    "topic": "qiot/things/admin/adfa/dfadf" # topic name
	  }
	],
	"caCert": "/v1/media/ca-crt.pem", # certificate file
	"clientCert": "/v1/media/xx-04-07_10-40-35/xx_certificate.pem",
	"privateCert": "/v1/media/xx-04-07_10-40-35/xx_privatekey.pem"
	}

https	{
	  "accesstoken": "r:2825dedfb012969e1dfb6adb8", # password
	  "myqnapcloudHost": "Not Available",
	  "clientId": "adfa_1491562164",
	  "host": [
	    "172.17.28.28"
	  ],
	  "requesterid": "00477f86-425b-49de-8590-xx", # username
	  "port": 3443, # https port
	  ...
	}

CoAP	{
	  "myqnapcloudHost": "Not Available",
	  "clientId": "adfa_1491562176",
	  "host": [
	    "172.17.28.28"
	  ],
	  "r": "00477f86-425b-49de-8590-1282c65b4348", # username
	  "t": "r:2825dedfb012969e1dfb6adb80a419df", # password
	  "port": 5683, # coap port
	  ...
	}