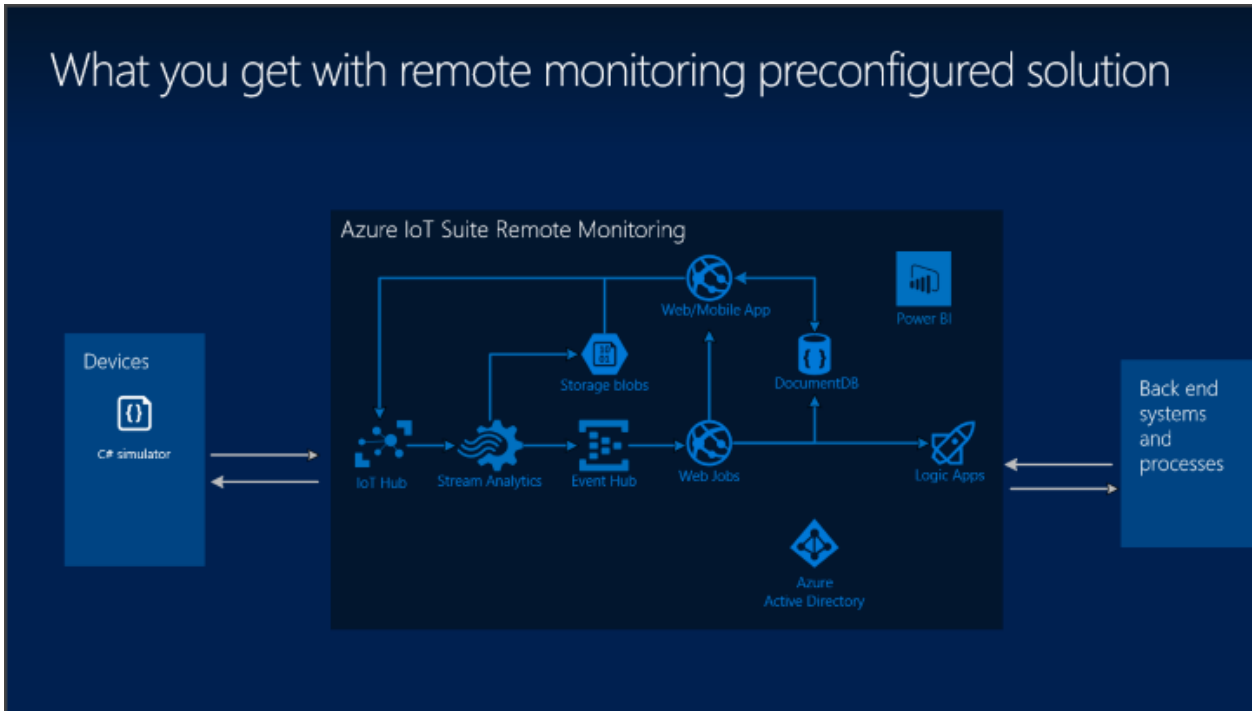
# IoT application demonstration

David Chen , Dec./2015 Rev 1.0

# 1.0 Purpose :

Understand how an IoT structure and how implement IoT client to works with Cloud . The following instructions describe the steps for connecting an mbed-enabled Freescale FRDM-K64F device to Azure IoT Hub.



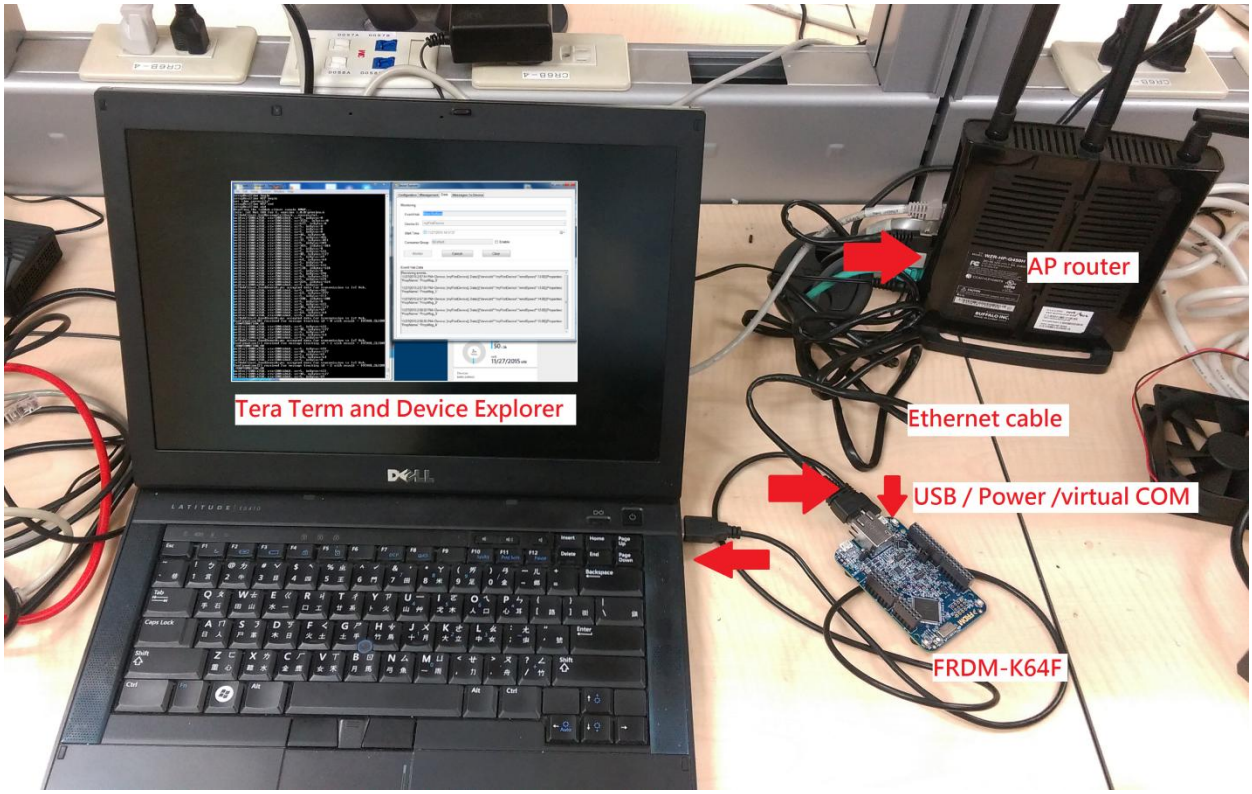

## 1.1 Demonstration :

1. IoT client (FRDM-K64F) report data to Cloud (Microsoft Azure)
2. IoT client receive data from Cloud
3. Could computing IoT client data and take action

## 1.2 Tool Requirement :

1. FRDM-K64F ( http://www.freescale.com/FRDM-K64F )
2. Device Explorer ( http://aka.ms/iot-hub-how-to-use-device-explorer )
3. Visual Studio 2015
4. SSH client ( PuTTY or Tera Term )
5. mbed ( http://www.mbed.com )
6. Microsoft Azure ( https://azure.microsoft.com )

## 1.3 Hardware setup:




# 2.0 Setup the demo

## 2.1 : Create an Microsoft Azure IoT Hub

### 2.1.1 IoT Hub

***Azure IoT Hub is a fully managed service that enables reliable and secure bi-directional communications between millions of IoT devices and an application back end.***

***More IoT Hub information please refer to below Microsoft link .***

- *IoT Hub documentation*

  https://azure.microsoft.com/en-us/documentation/services/iot-hub/

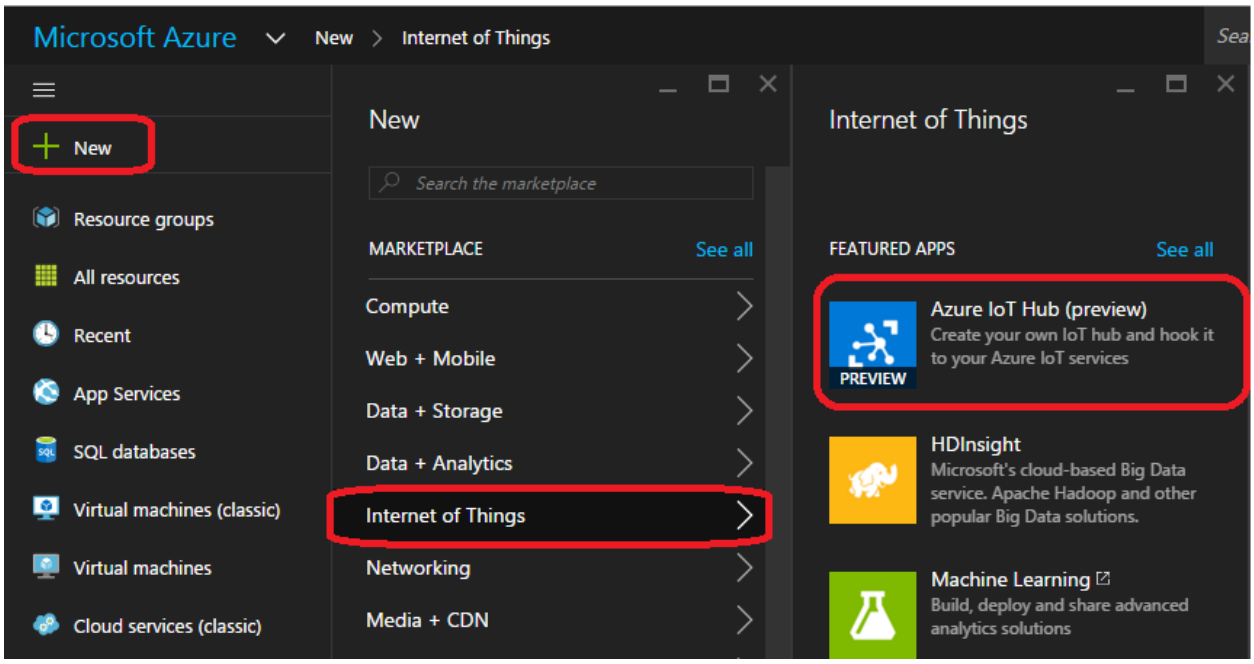
- *Azure IoT Hub developer guide*

  https://azure.microsoft.com/en-us/documentation/articles/iot-hub-devguide/

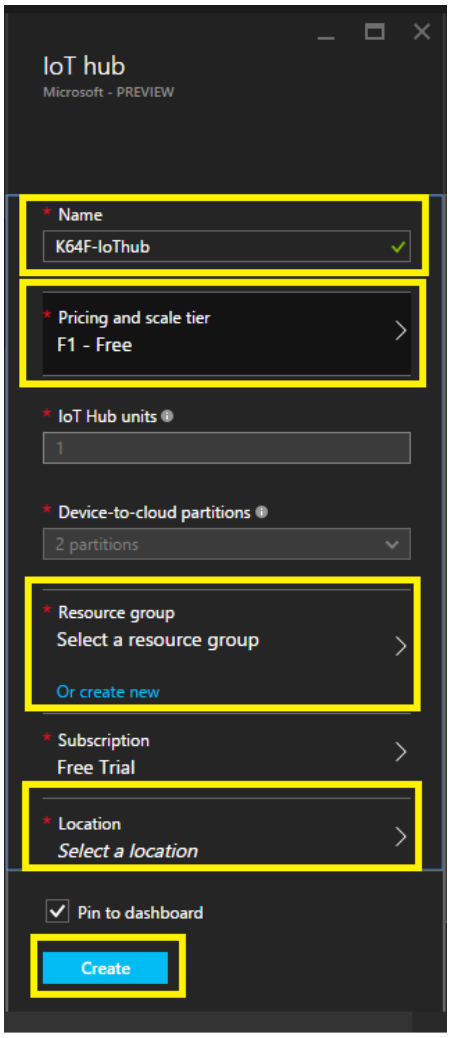
### 2.1.2 Step of create IoT Hub

*Step 1 :* Log on to the Azure portal.

*Step 2:* In the jumpbar, click **New**, then click **Internet of Things**, and then click **IoT Hub**.

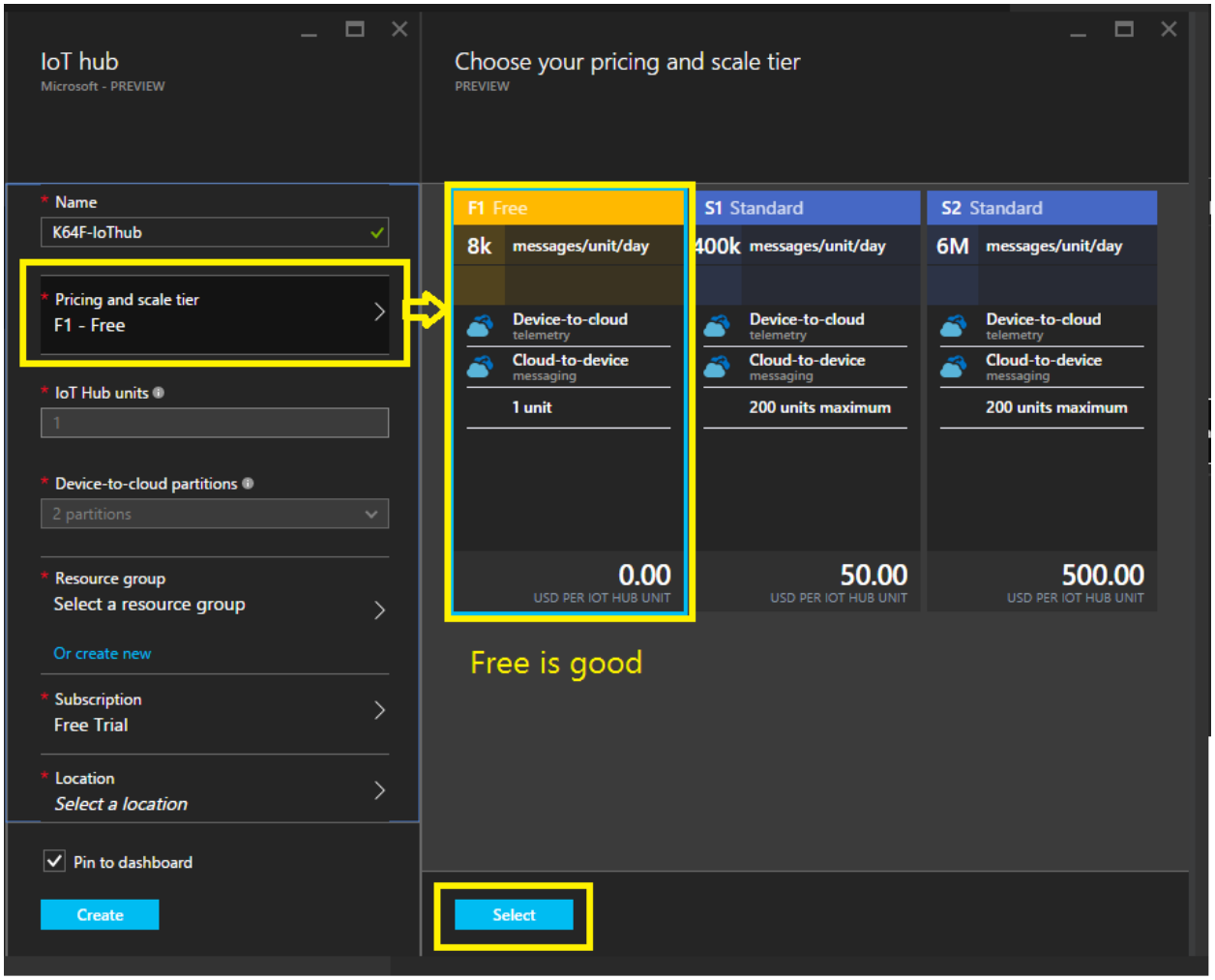



*Step 3:* In the **New IoT Hub** blade, specify the desired configuration for the IoT Hub.

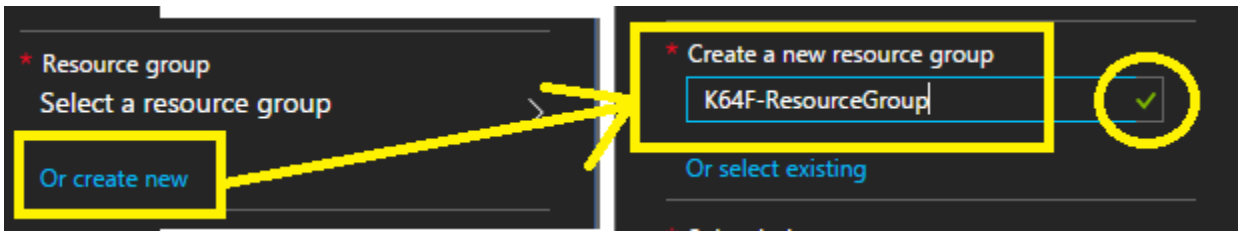



***In the** Name **box, enter a name to identify your IoT hub. When the** Name **is validated, a green check mark appears in the** Name **box.***

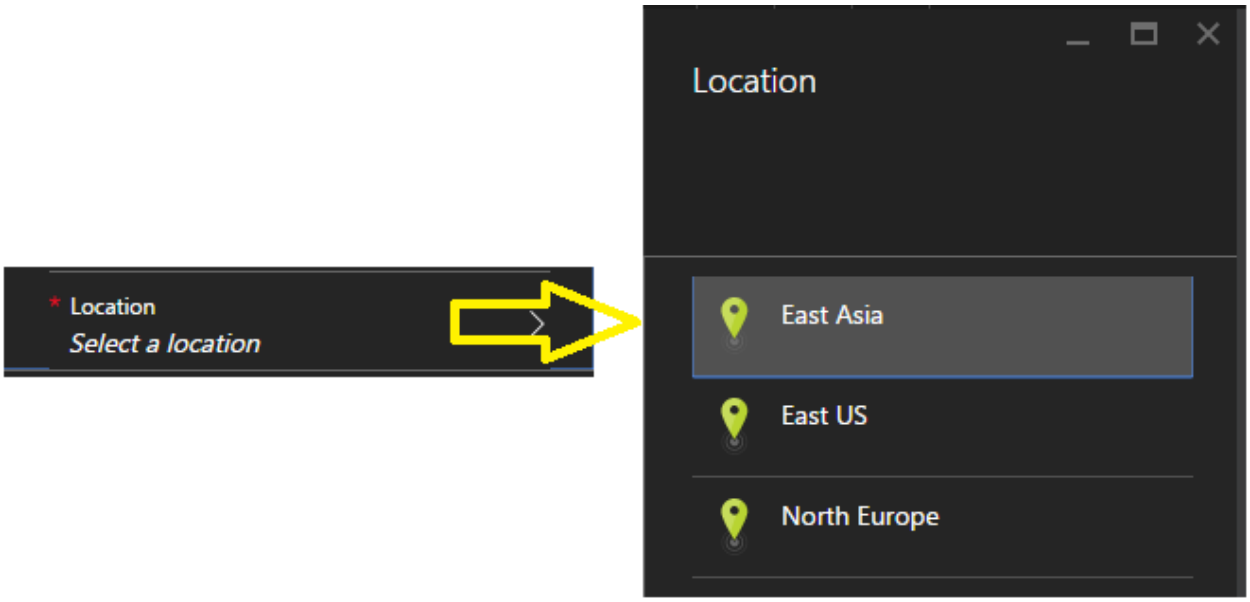

*Change the* Pricing and scale tier *as desired. This demo does not require a specific tier.*



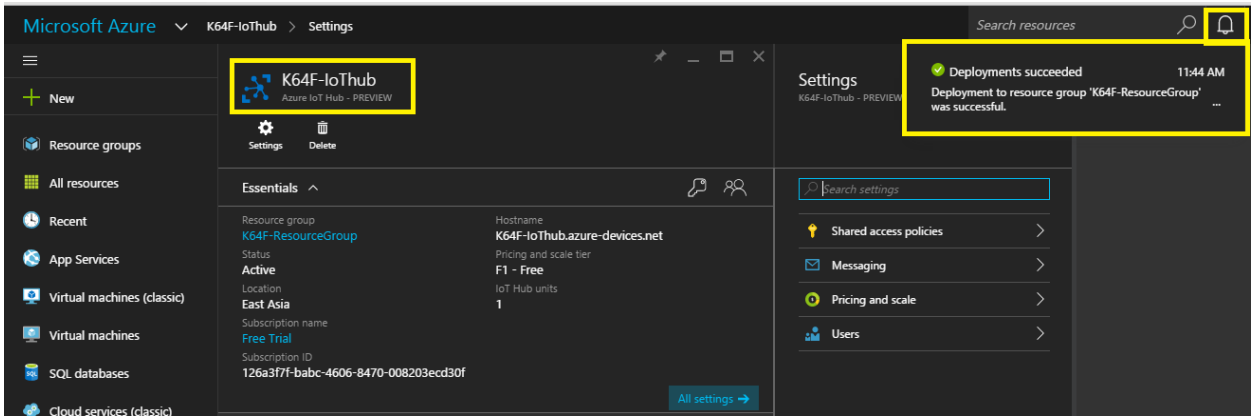

*In the* Resource group *box, create a new resource group, or select and existing one. For more information, see* ***Using resource groups to manage your Azure resources****.*



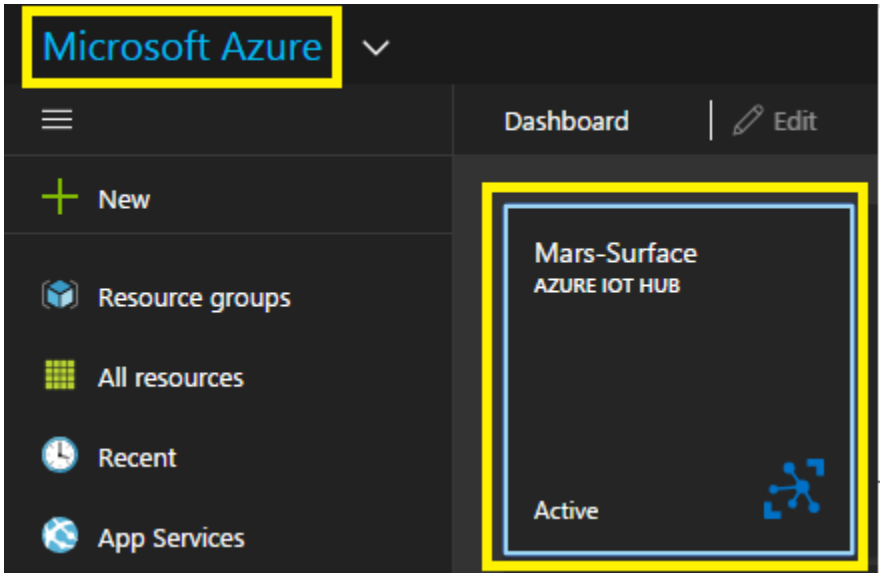

*Use* Location *to specify the geographic location in which to host your IoT hub.*

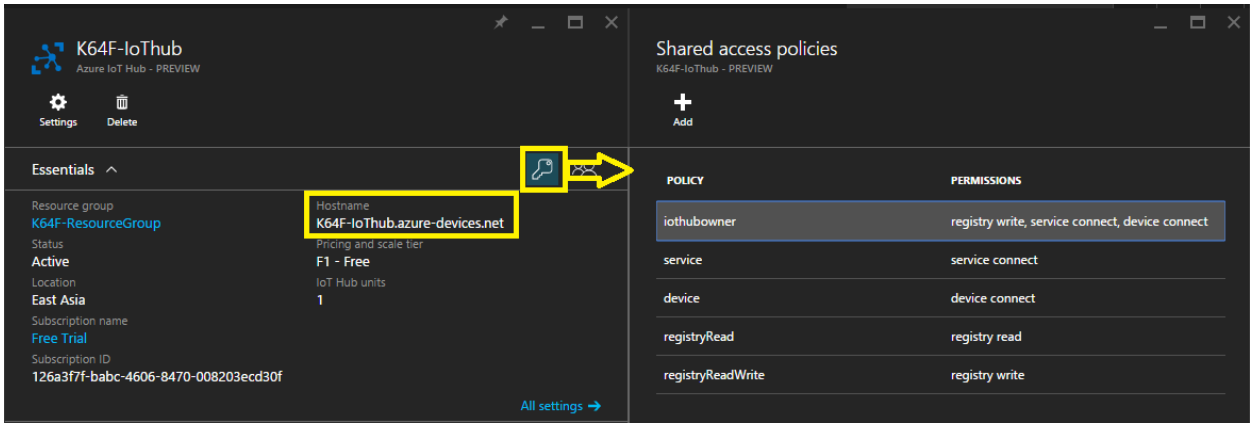



*Step 4 :* Once the new IoT hub options are configured, click **Create**. It can take a few minutes for the IoT hub to be created. To check the status, you can monitor the progress on the Startboard. Or, you can monitor your progress from the Notifications section.

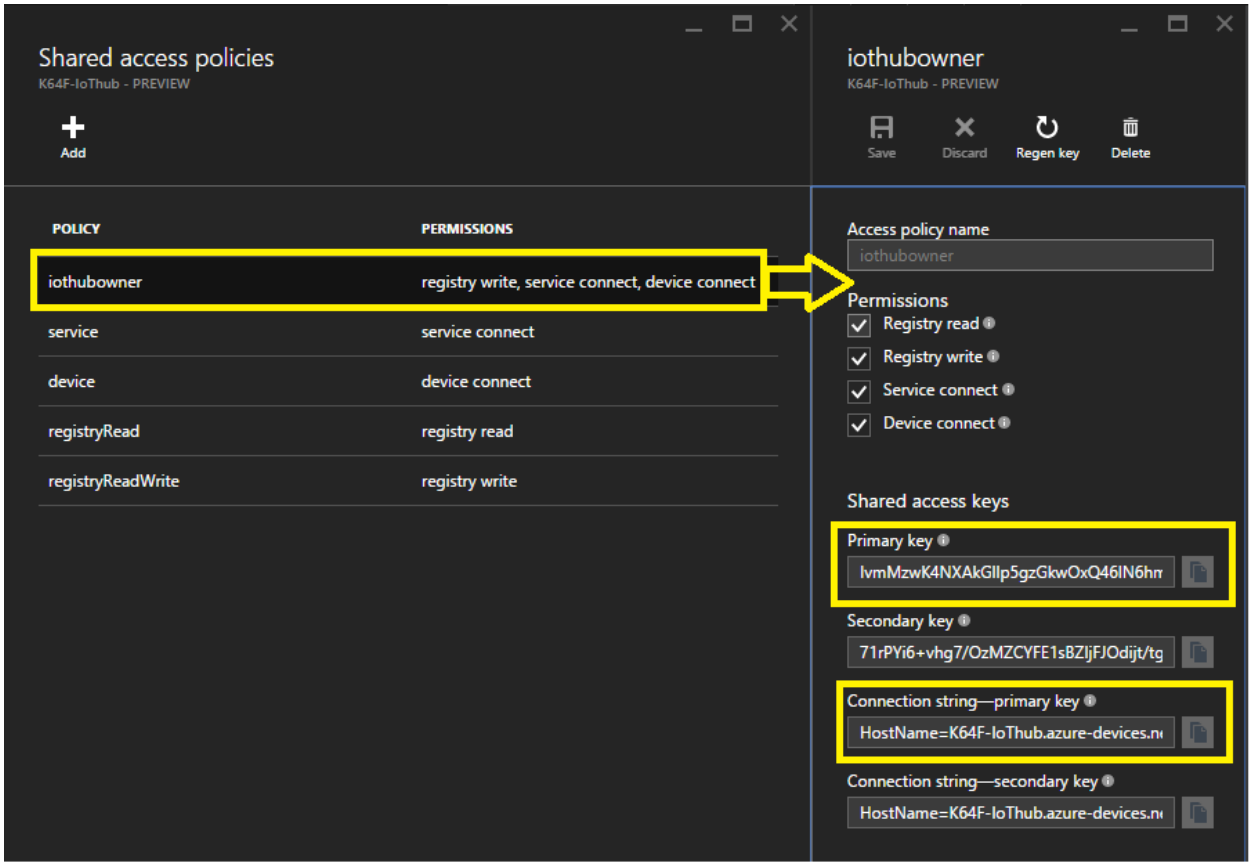



*Step 5 :* After the IoT hub has been created successfully, open the blade of the new IoT hub, take note of the **Hostname** , and select the **Key** icon on the top.






*Step 6 :* Select the Shared access policy called **iothubowner**, then copy and take note of the **Primary key** and **Connection string** on the right blade.




IoT hub is now created, and we have the **Hostname** ,**Primary Key** and **Connection string** that we need to configure IoT client .

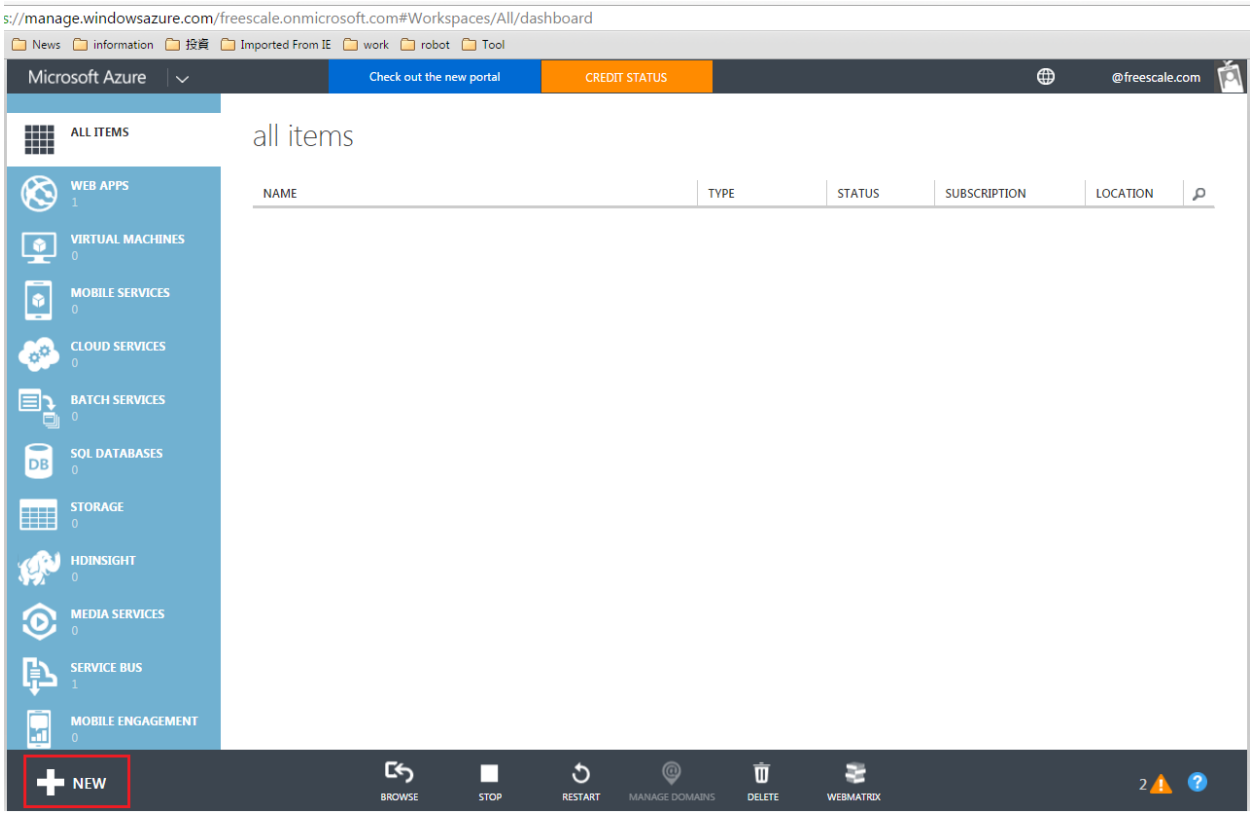
## 2.2 Create an Microsoft Azure Event Hub

### 2.2.1 Event Hub

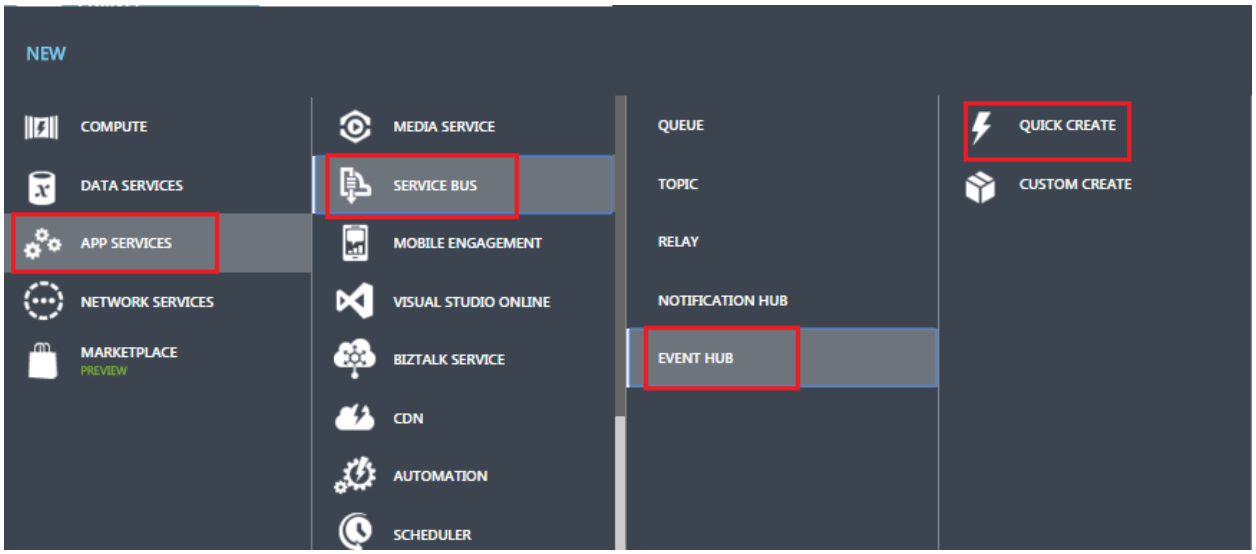
### 2.2.2 : Step of  Event Hub

*Step 1 :* Log on to the Azure Management Portal.

*Step 2:* In the lower left corner of the page, click on the **+ NEW** button.

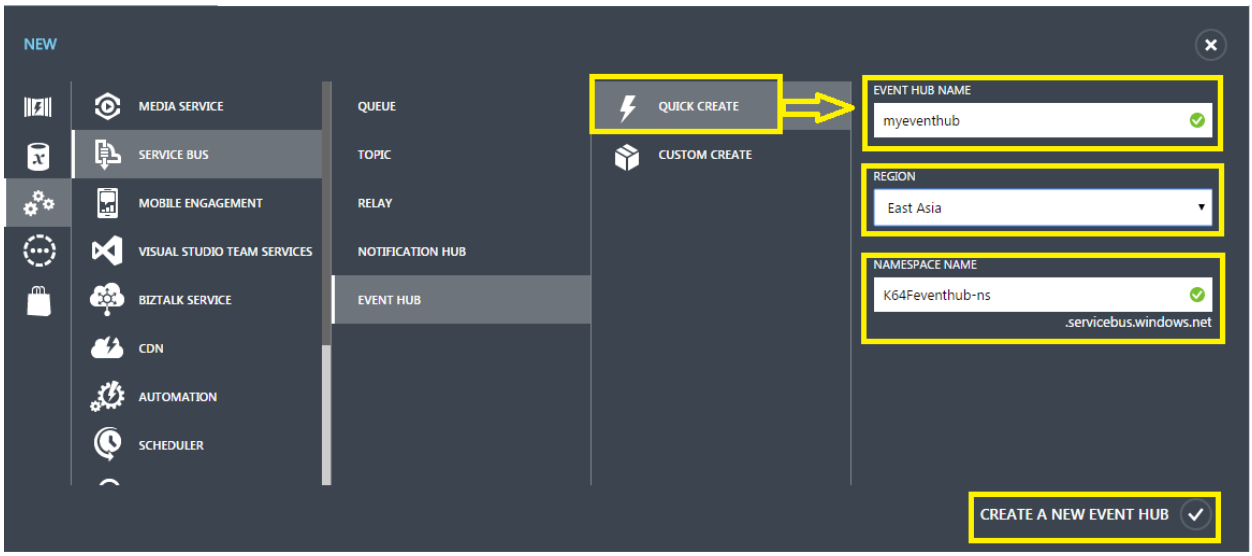



*Step 3 :* Select **App Services**, **Service Bus**, **Event Hub**, **Quick Create**

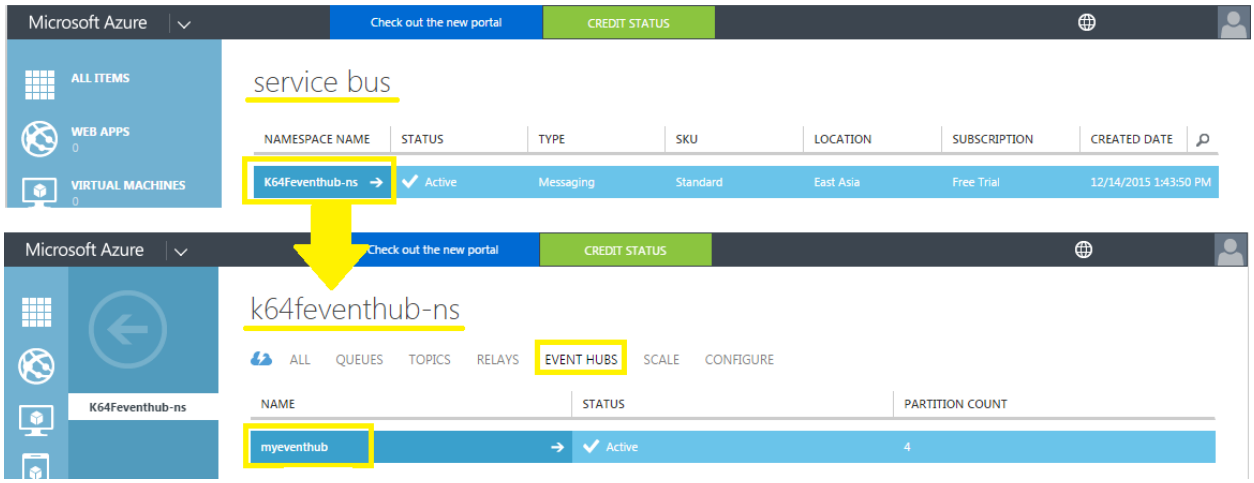



*Step 4 :* Enter the following settings for the Event Hub (use a name of your choice for the event hub and the namespace):

- Event Hub Name: "*myeventhubname*"
- Region: your choice
- Namespace Name: "*k64Feventhub-ns*"

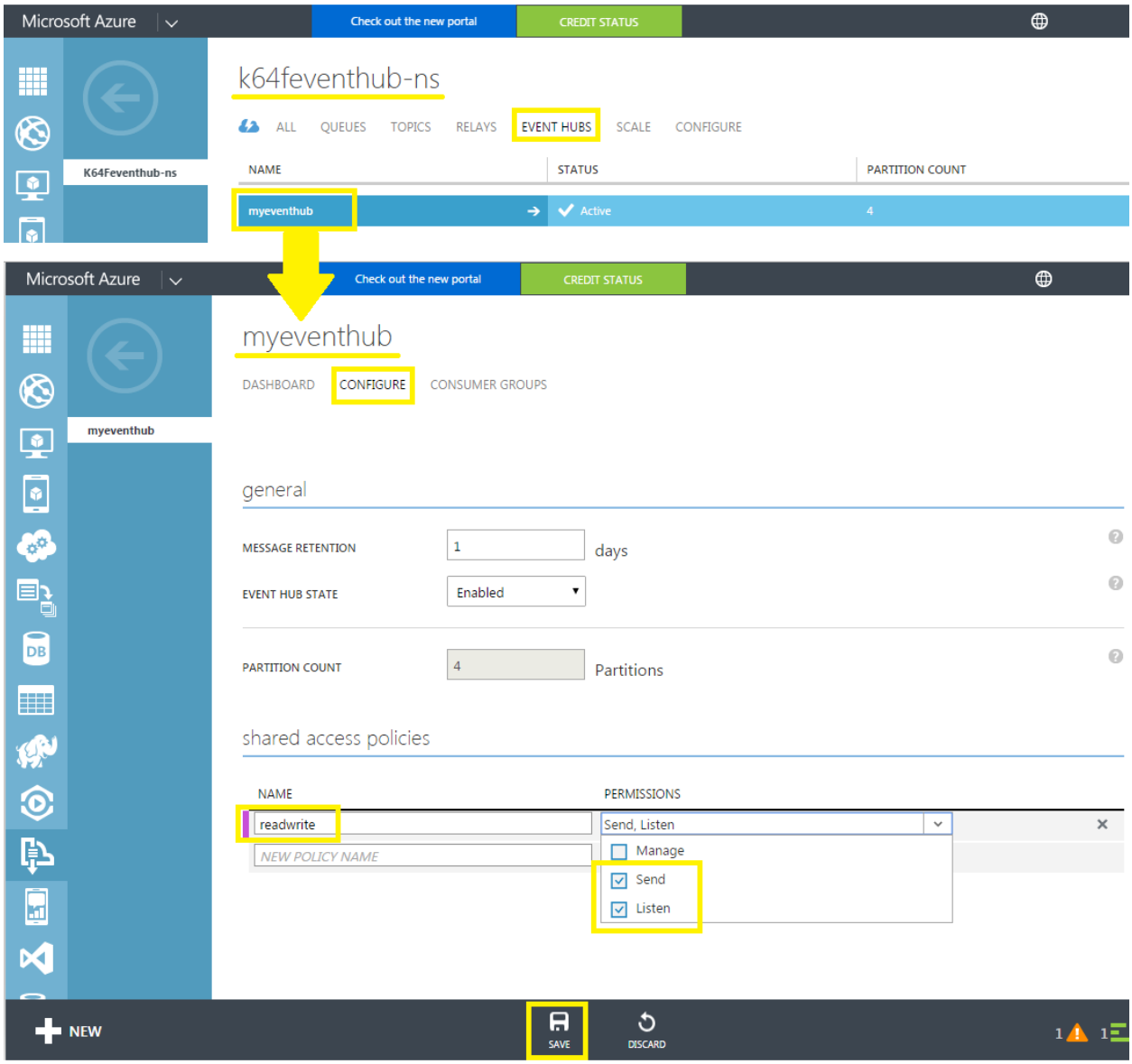



*Step 5 :* Select the *K64Feventhub-ns* ( *namespace* ) and go in the **Event Hub** tab

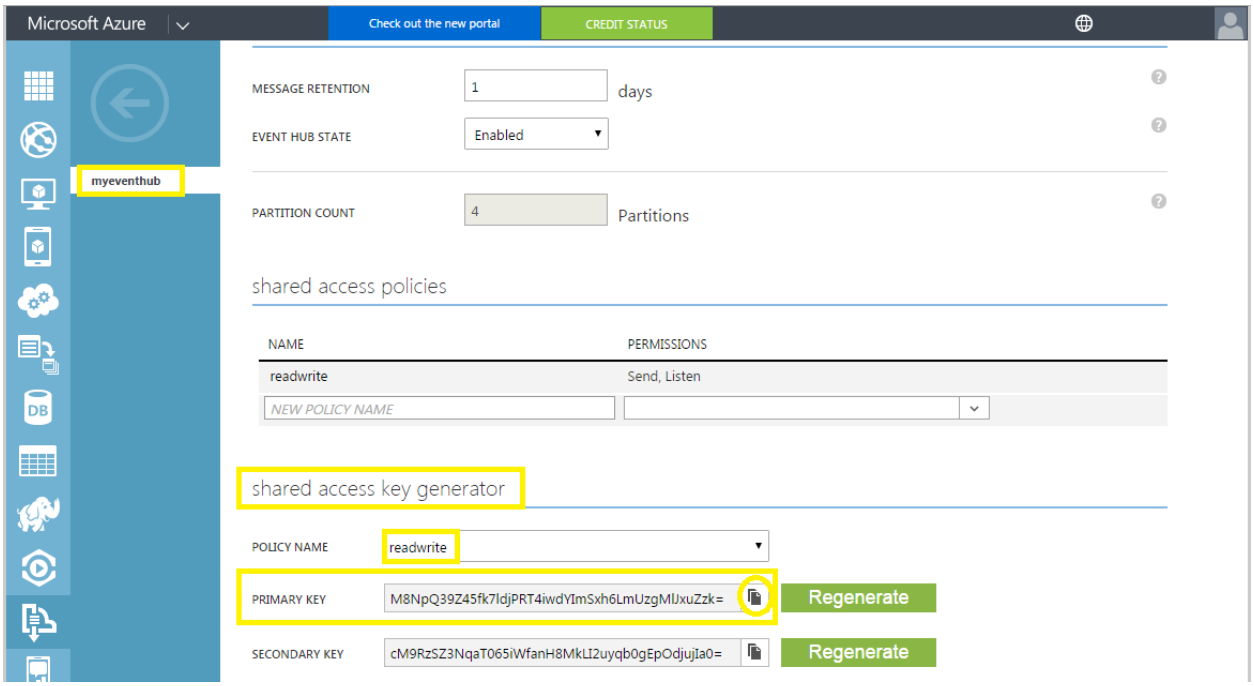



*Step 6 :* Select the *myeventhubname* event hub and go in the **Configure** tab , in the **Shared Access Policies** section, add a new policy:

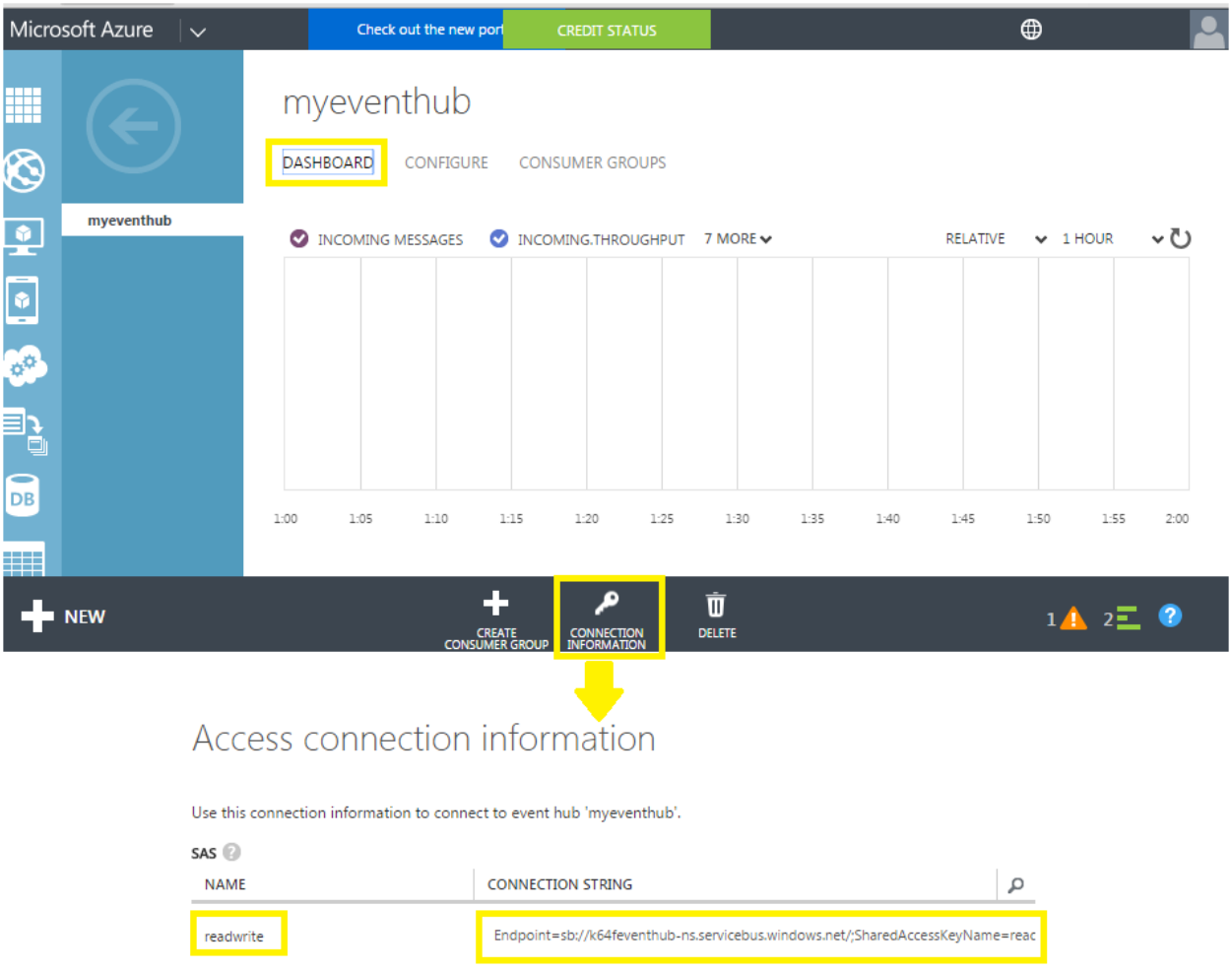
- Name = "readwrite"
- Permissions = Send, Listen




*Step 7 :* Click **Save**, then you will find there are two shared policy access key generated . Write down the “Policy Name” and “Primary Key “ .




*Step 8 :* Then go to the evnet hub **Dashboard** tab and click on **Connection Information** at the bottom . Write down the connection string for the readwrite policy name.



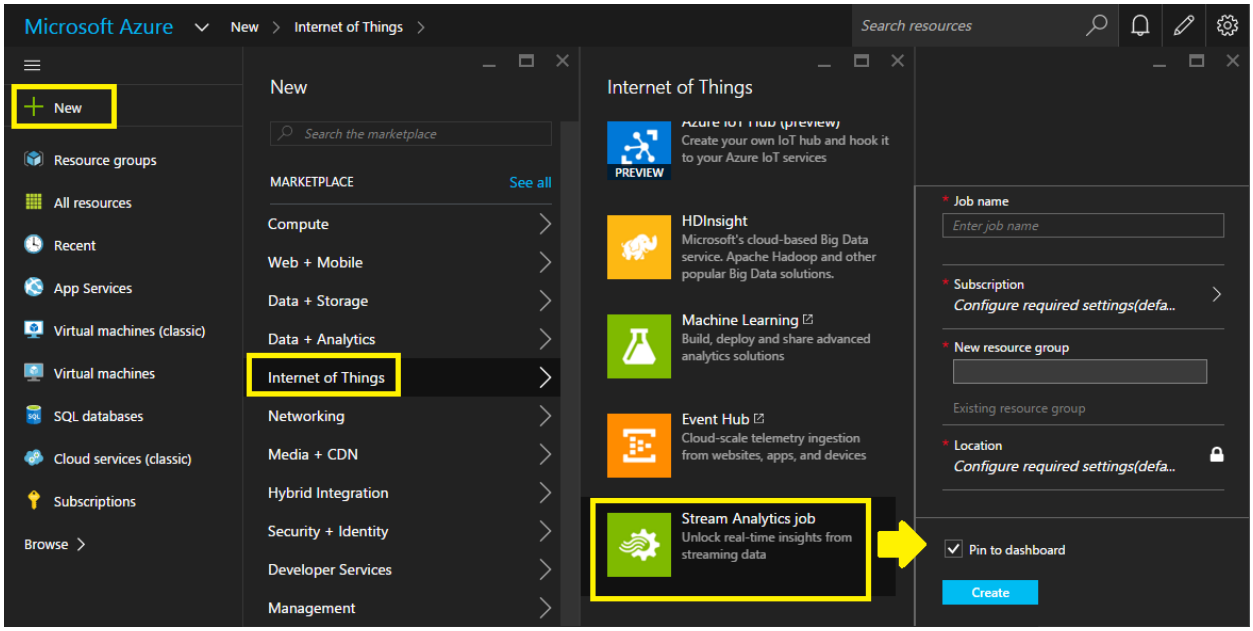
## 2.3 Create a Stream Analytics job

### 2.3.1 Stream Analytics

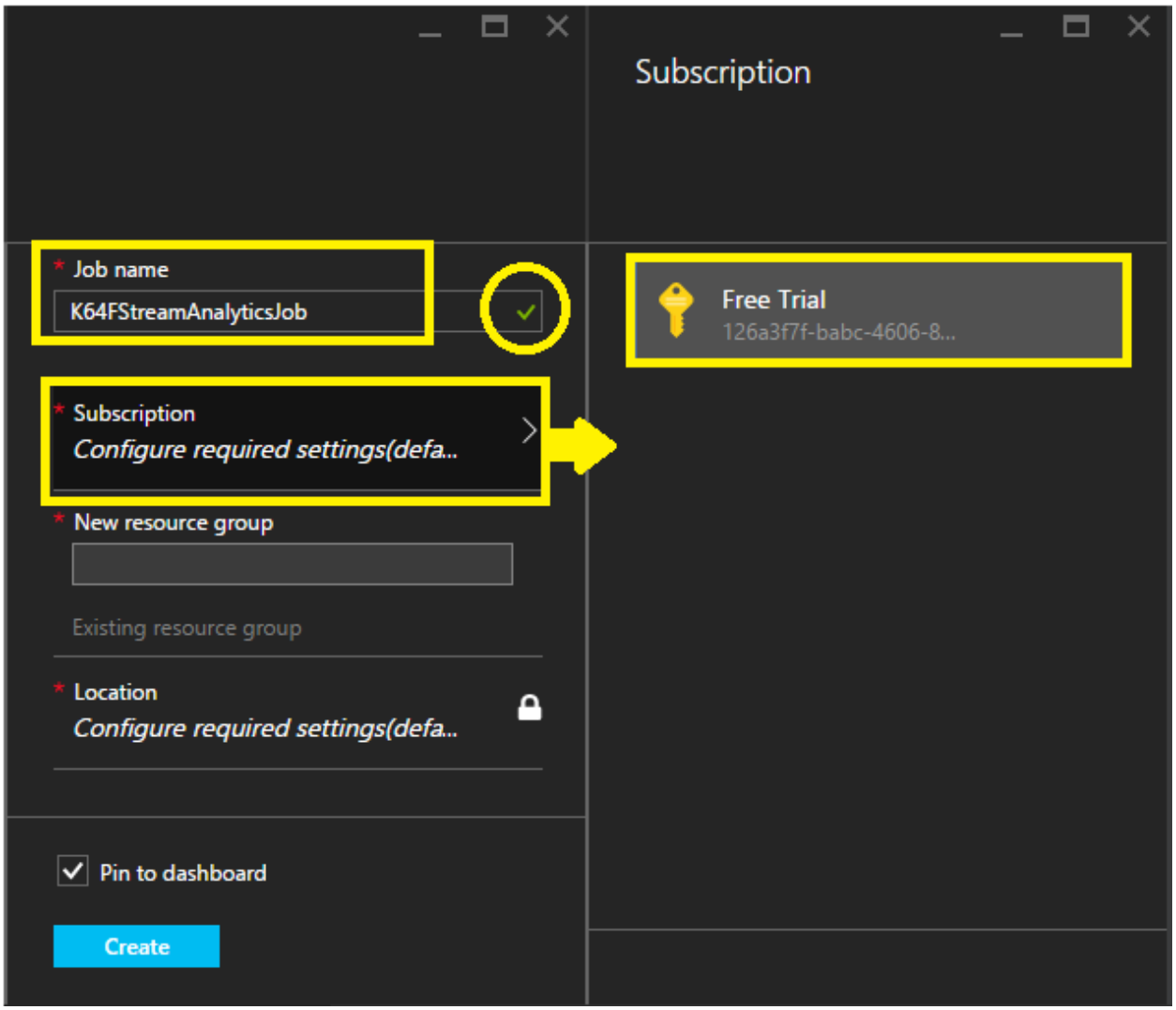
### 2.3.2 Step of create a Stream Analytics job

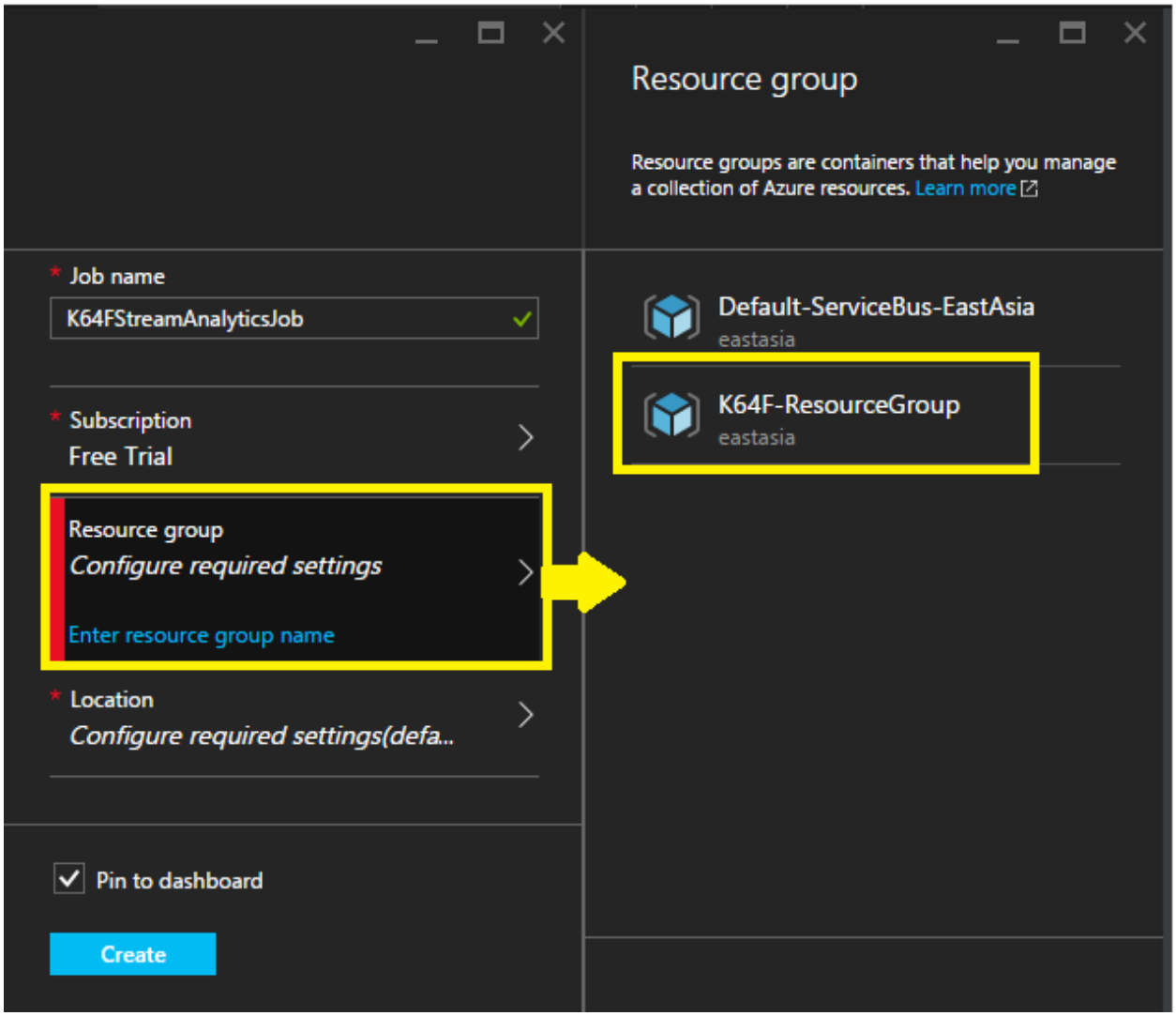
*Step 1 :* Log on to the [Azure Preview Portal](Azure Preview Portal).

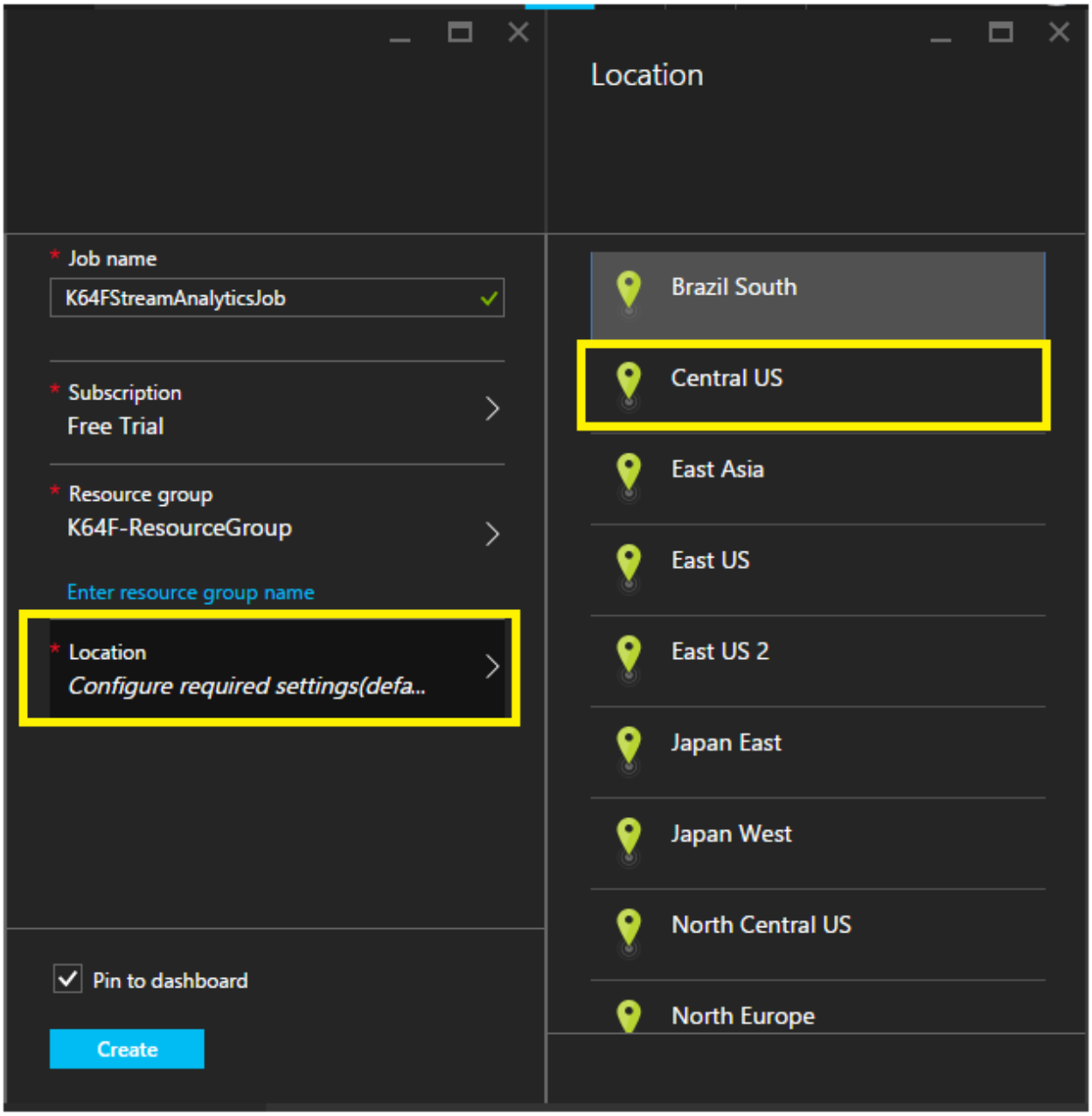
*Step 2 :* In the jumpbar, click **New**, then click **Internet of Things**, and then click **Stream Analytics job**.



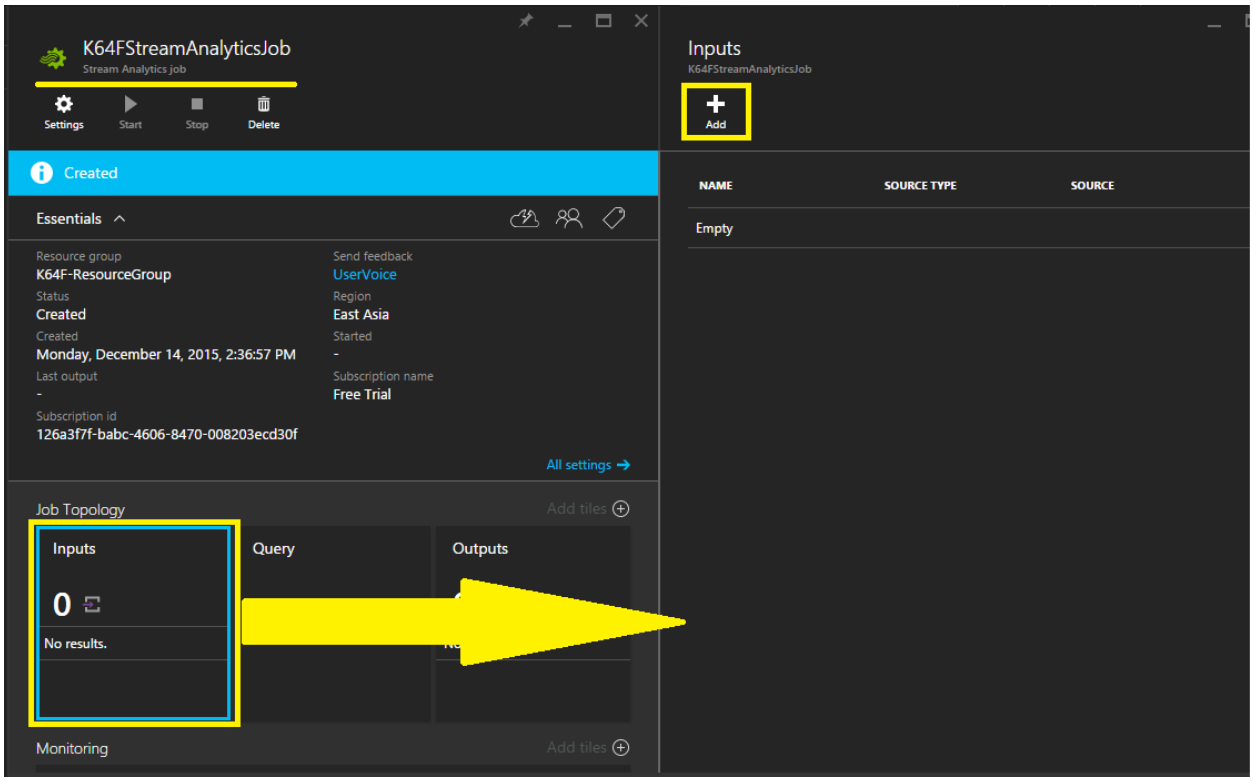

*Step 3 :* Enter a name for the job, a prefered region, choose your subscription. At this stage you are also offered to create a new or to use an existing resource group.


Subscription
* Job name
K64FStreamAnalyticsJob
* Subscription
Configure required settings(defa...
* New resource group
Existing resource group
* Location
Configure required settings(defa...
Pin to dashboard
Create
Free Trial
126a3f7f-babc-4606-8...


Resource group
Resource groups are containers that help you manage a collection of Azure resources. Learn more
* Job name
K64FStreamAnalyticsJob
* Subscription
Free Trial
Resource group
Configure required settings
Enter resource group name
* Location
Configure required settings(defa...
Pin to dashboard
Create
Default-ServiceBus-EastAsia
eastasia
K64F-ResourceGroup
eastasia

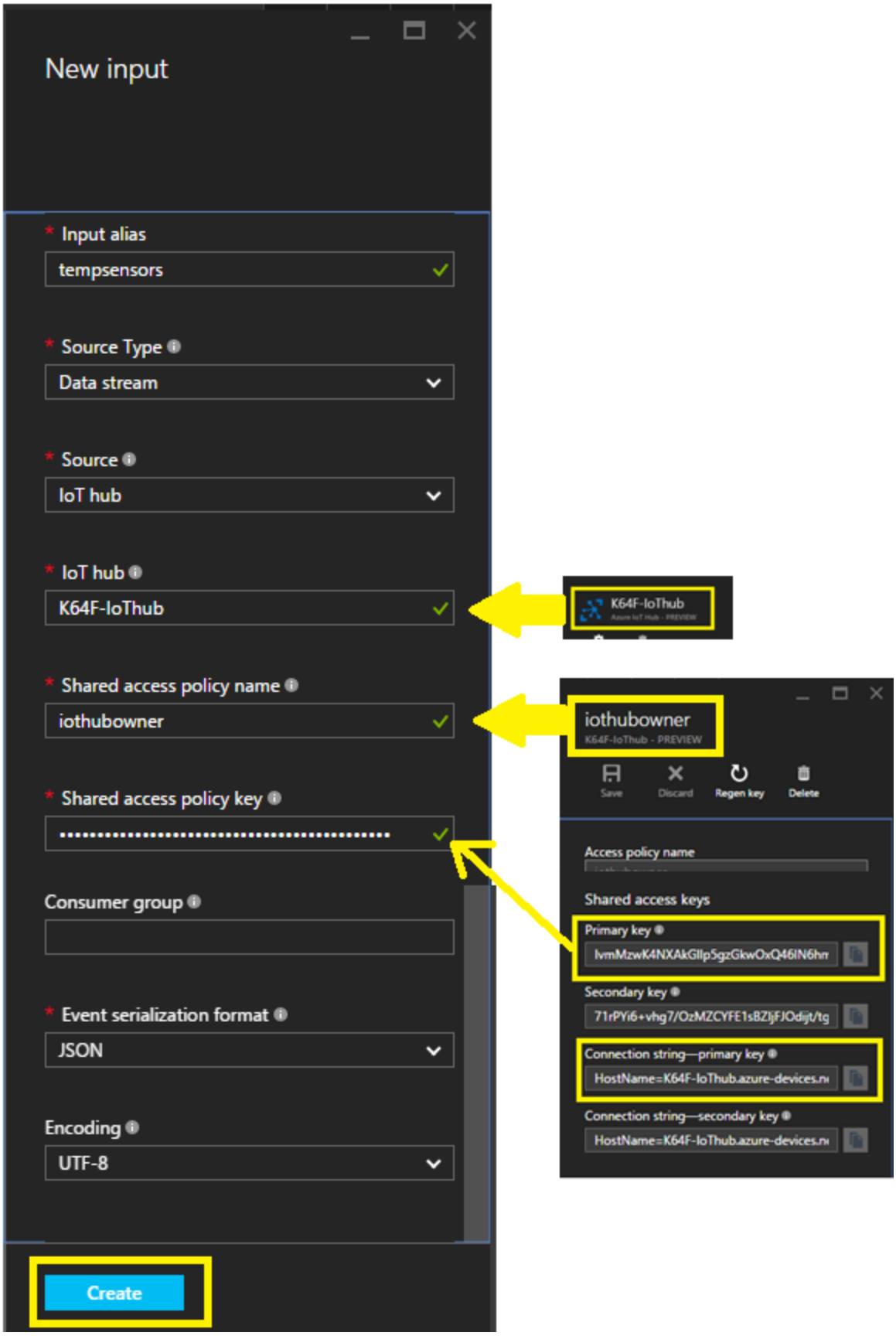



*Step 4 :* After the job is created , click on the **Inputs** tile in the **job topology** section. In the **Inputs blade**, click on **Add**




*Step 5 :* Enter the following settings:

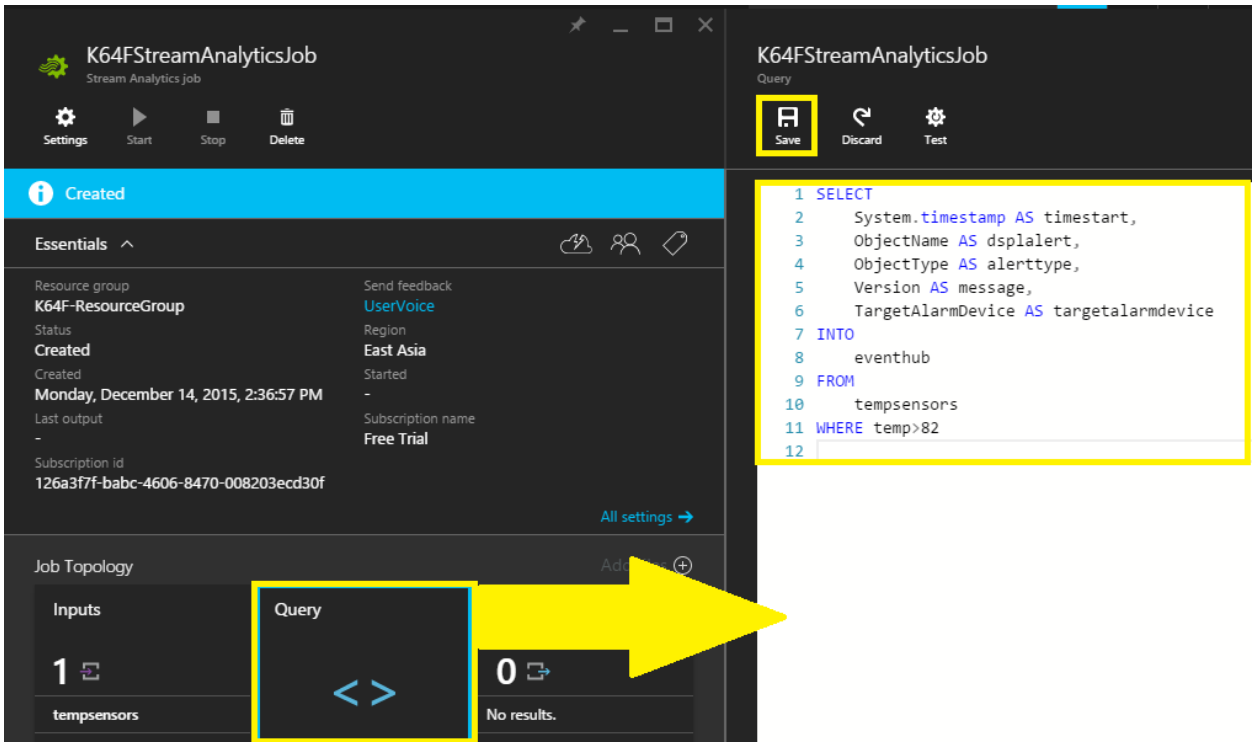
- Input Alias = "tempsensors"
- Type = "Data Stream"
- Source = "IoT Hub"
- IoT Hub = "*K64F-IoThub*" ( The name for the IoT Hub that we create at chapter 2.1.2 step 2 )
- Shared Access Policy Name = "iothubowner"
- Shared Access Policy Key = "**iothubowner Primary Key**" (That's the key you wrote down when creating the IoT Hub , pls refer to chapter 2.1.2 step 6 )
- IoT Hub Consumer Group = "" (leave it to the default empty value)
- Event serialization format = "JSON"
- Encoding = "UTF-8"



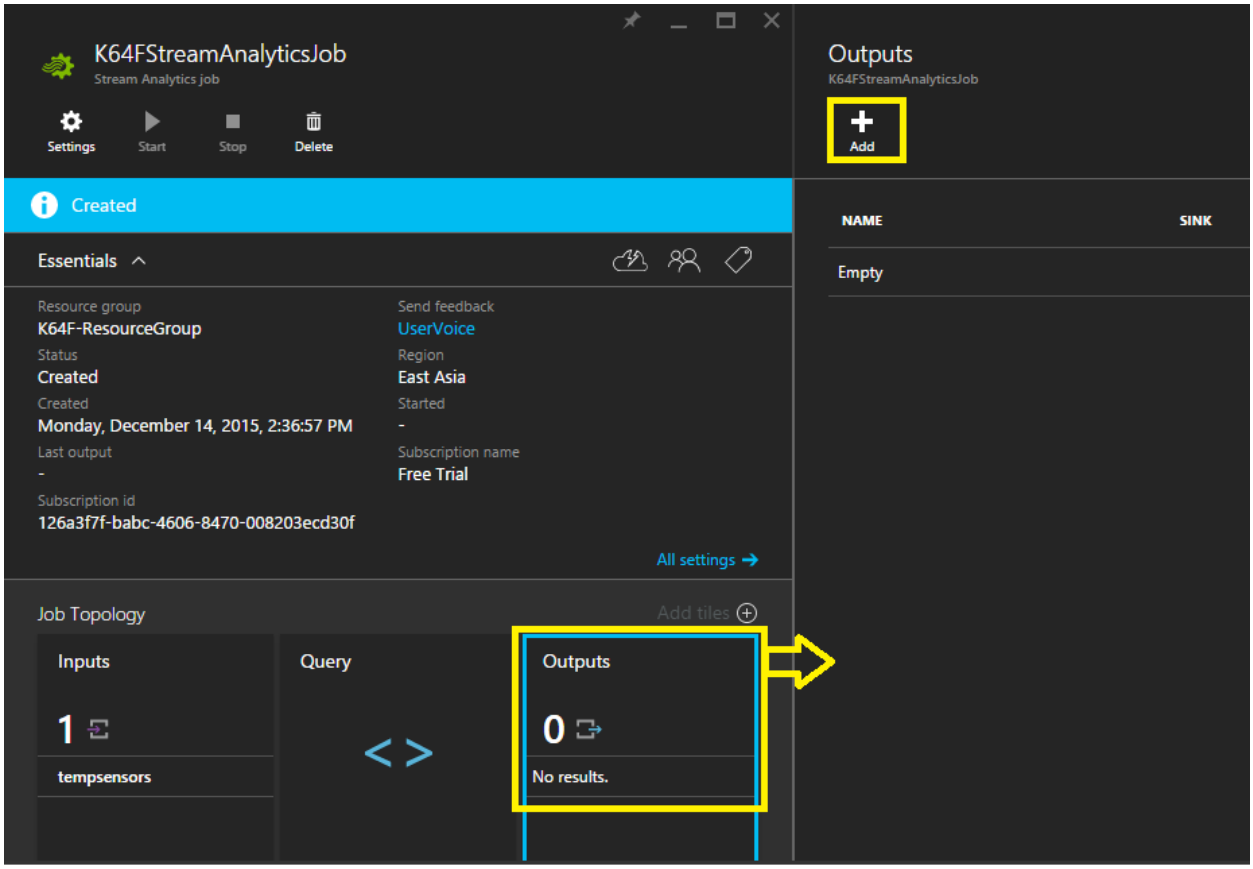

*Step 6 :* Back to the Stream Analytics Job blade, click on the **Query** tile. In the Query settings blade, type in the below query and click **Save**

```
SELECT
    System.timestamp AS timestart,
```

```
    ObjectName AS dsplalert,
    ObjectType AS alerttype,
    Version AS message,
    TargetAlarmDevice AS targetalarmdevice
INTO
    eventhub
FROM
    tempsensors
WHERE temp>82
```

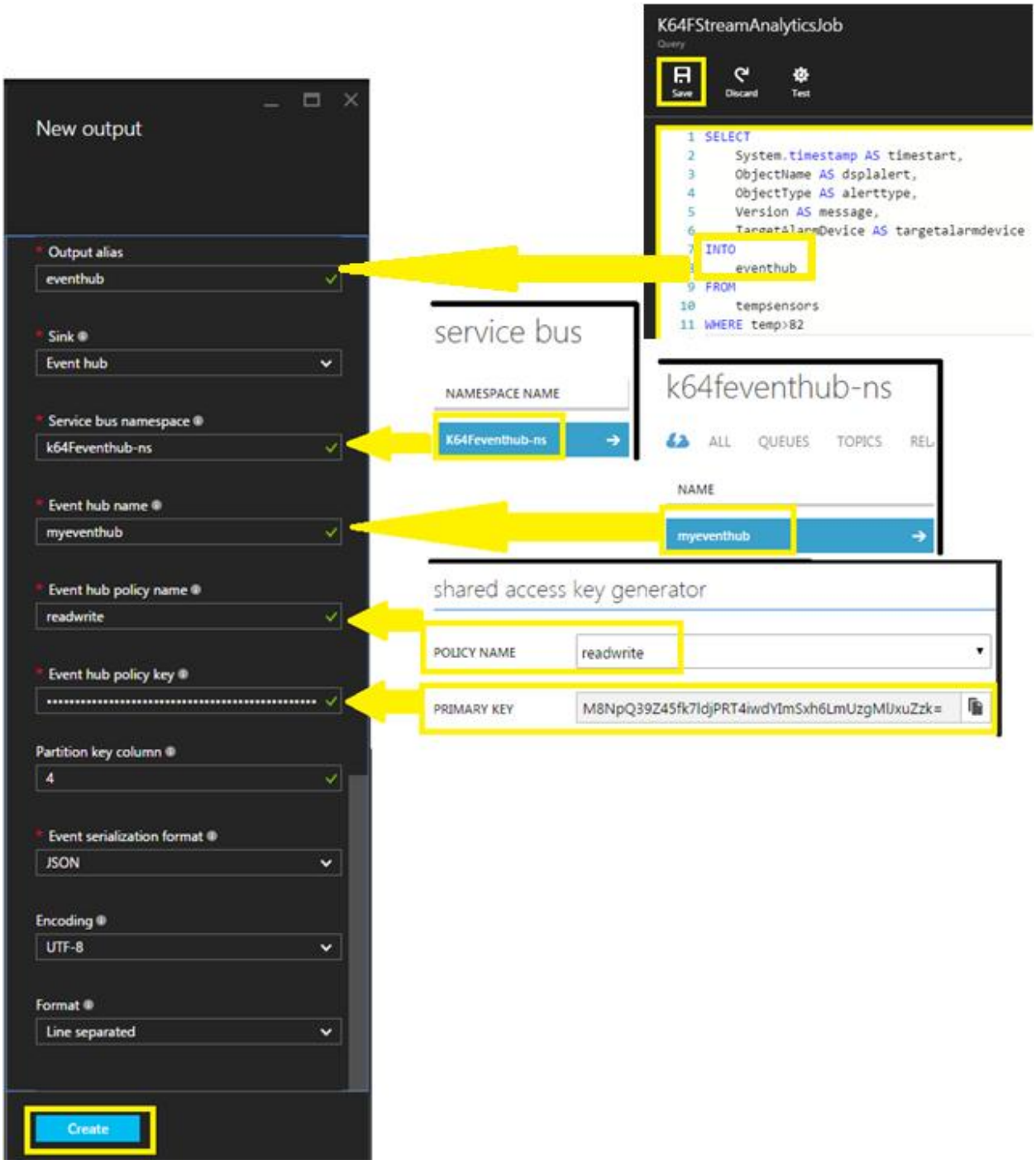



*Step 7 :* Back to the Stream Analytics Job blade, click on the **Outputs** tile and in the Outputs blade, click on **Add**

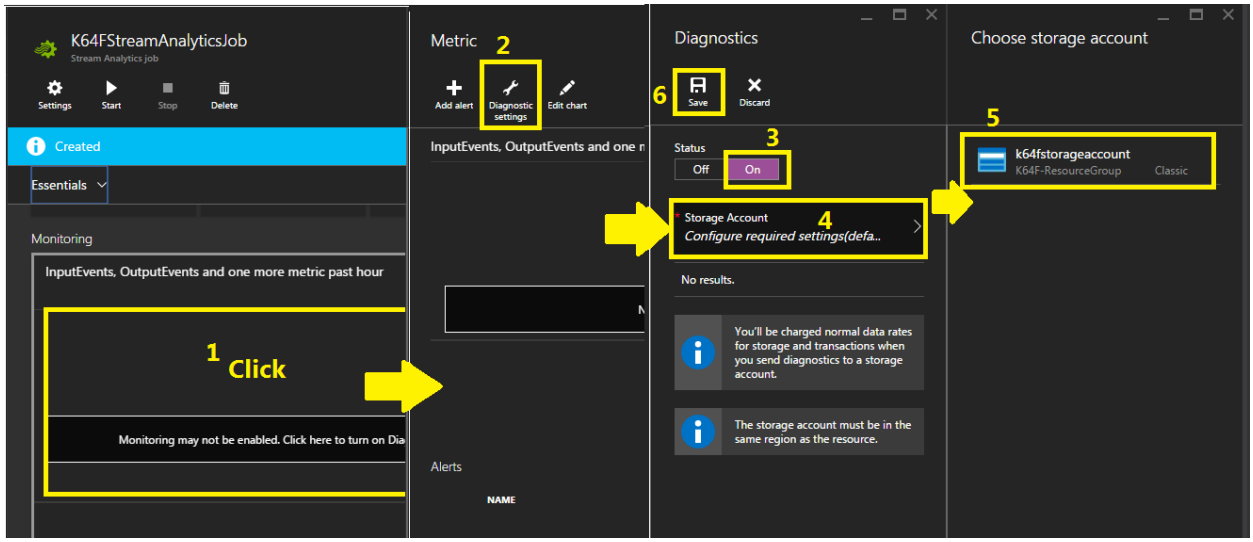



*Step 8 :* Enter the following settings then click on **create**:

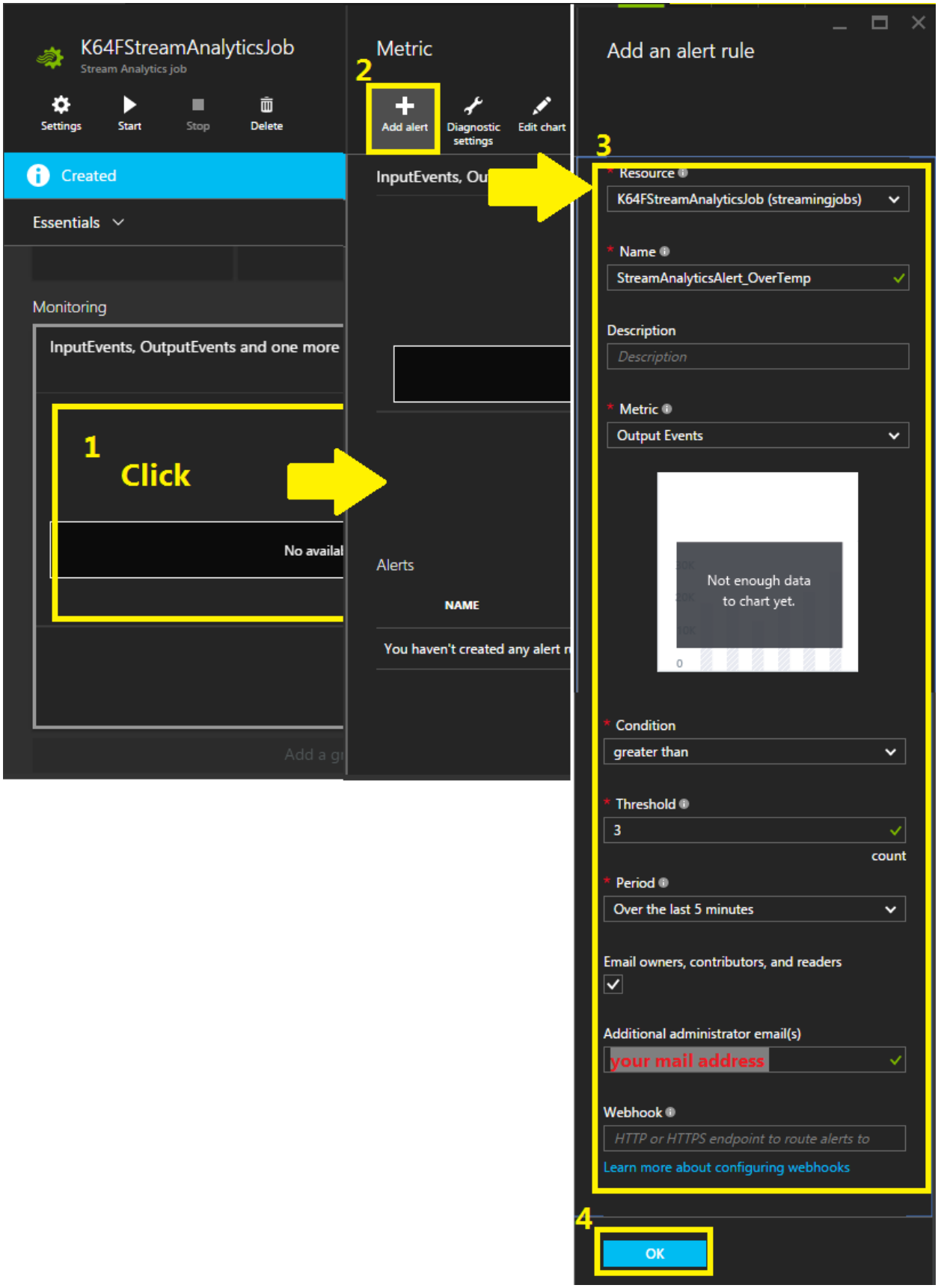
- Output Alias = "eventhub" ( the name is used at chapter 2.3.2 step 6 )
- Source = "Event Hub"
- Service Bus Namespace = "*K64Feventhub-ns" ( please refer to chapter2.2.2 step 4 )*
- Event Hub Name = "*myeventhub*" (please refer to chapter 2.2.2 step 4 )
- Event Hub Policy Name = "readwrite" (please refer to chapter 2.2.2 step 6 )
- Event Hub Policy Key = "*Primary Key for readwrite Policy name*" ( please refer to chapter 2.2.2 step 7 )
- Partition Key Column = "4"
- Event Serialization format = "JSON"
- Encoding = "UTF-8"
- Format = "Line separated"



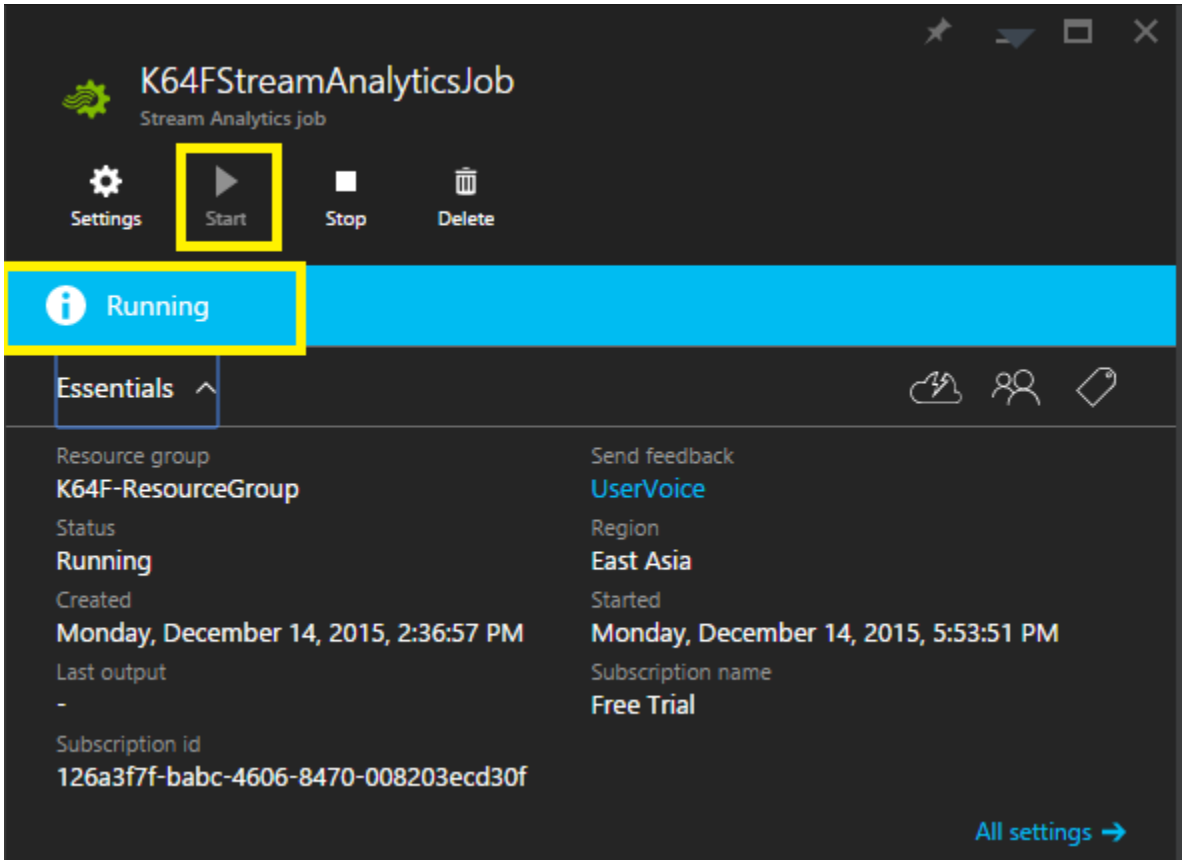

*Step 9 :* Configuration and Enable Monitoring .




*Step 10 :* Setup alert condition and action




*Step 11 :* Back in the Stream Analytics blade, start the job by clicking on the **Start** button at the top




## 2.4 Create a storage account

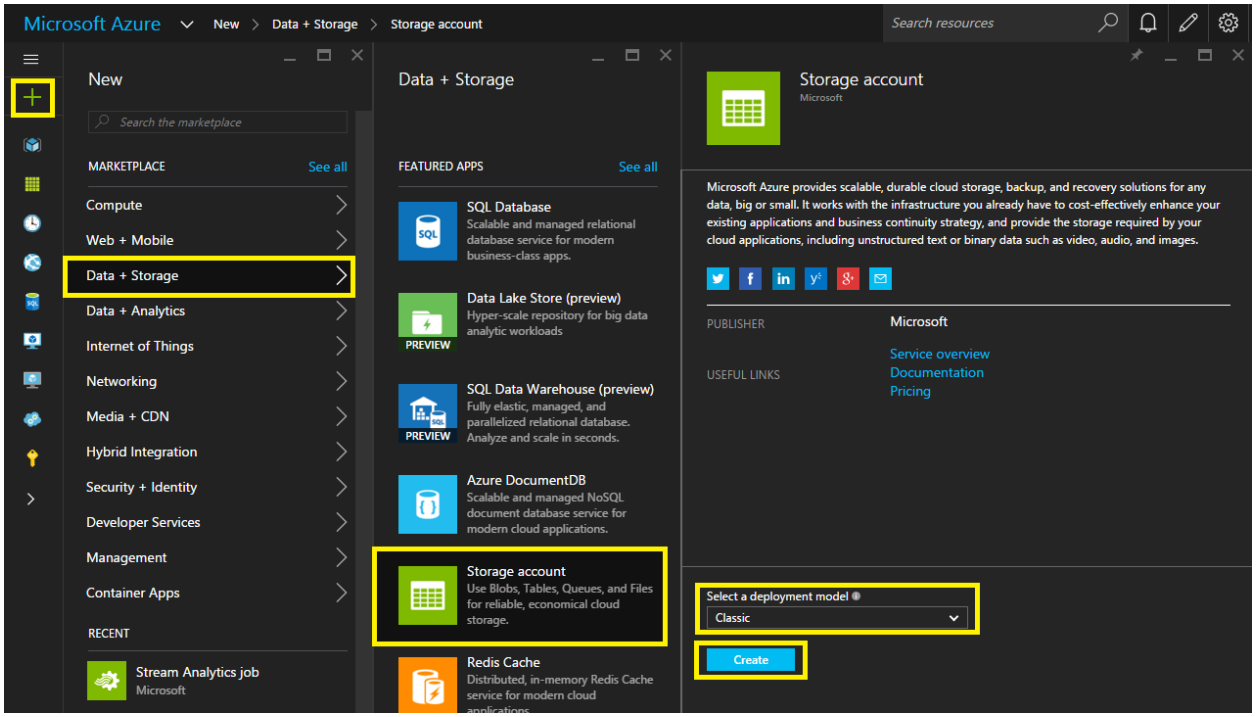
### 2.4.1 Storage account

### 2.4.2 Step of create a storage account

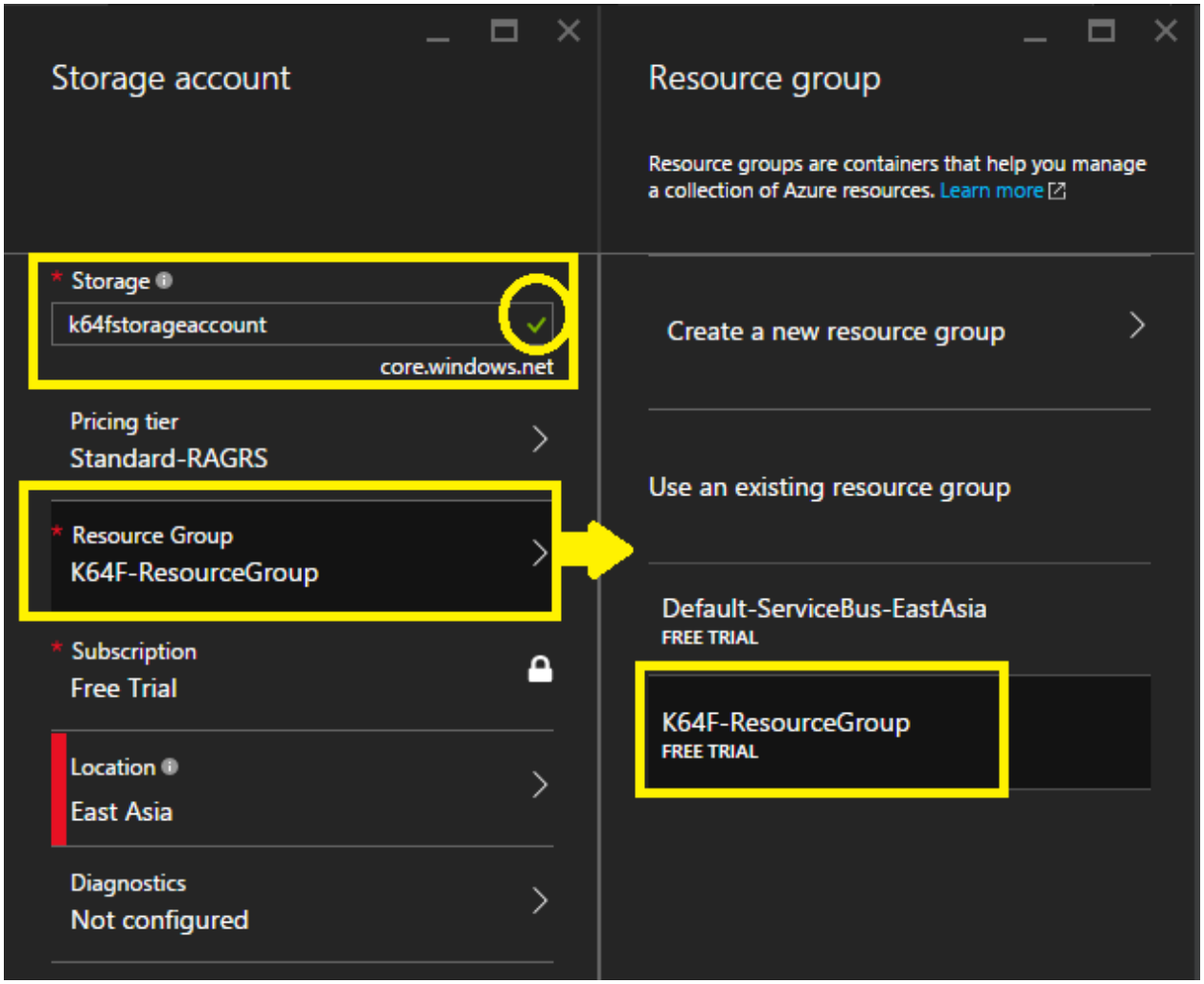
*Step 1 :* Log on to the [Azure Preview Portal](Azure Preview Portal).

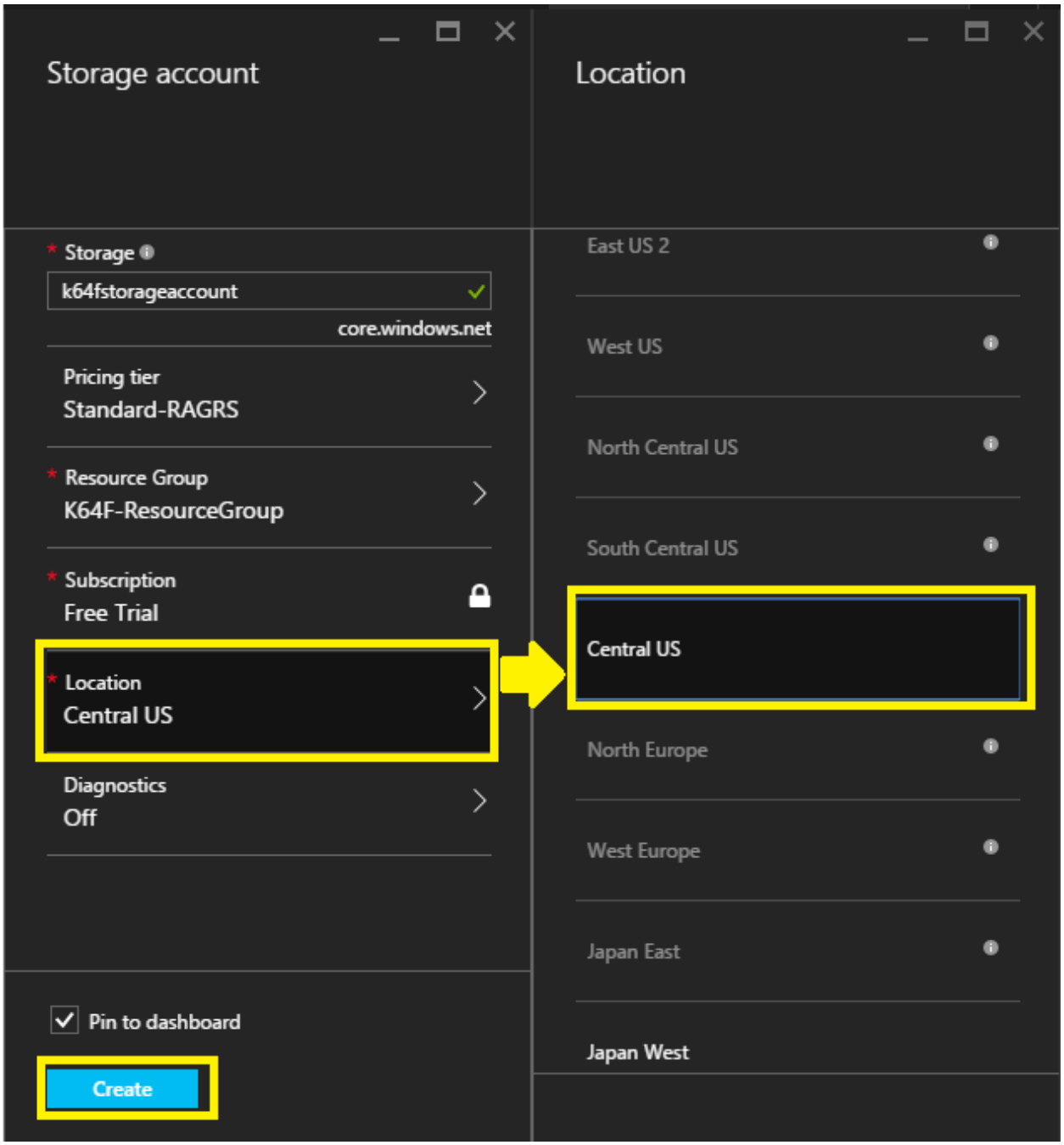
*Step 2 :* In the jumpbar, click **New** and select **Data + Storage** then**Storage Account**

*Step 3 :* Choose **Classic** for the deployment model and click on**create**

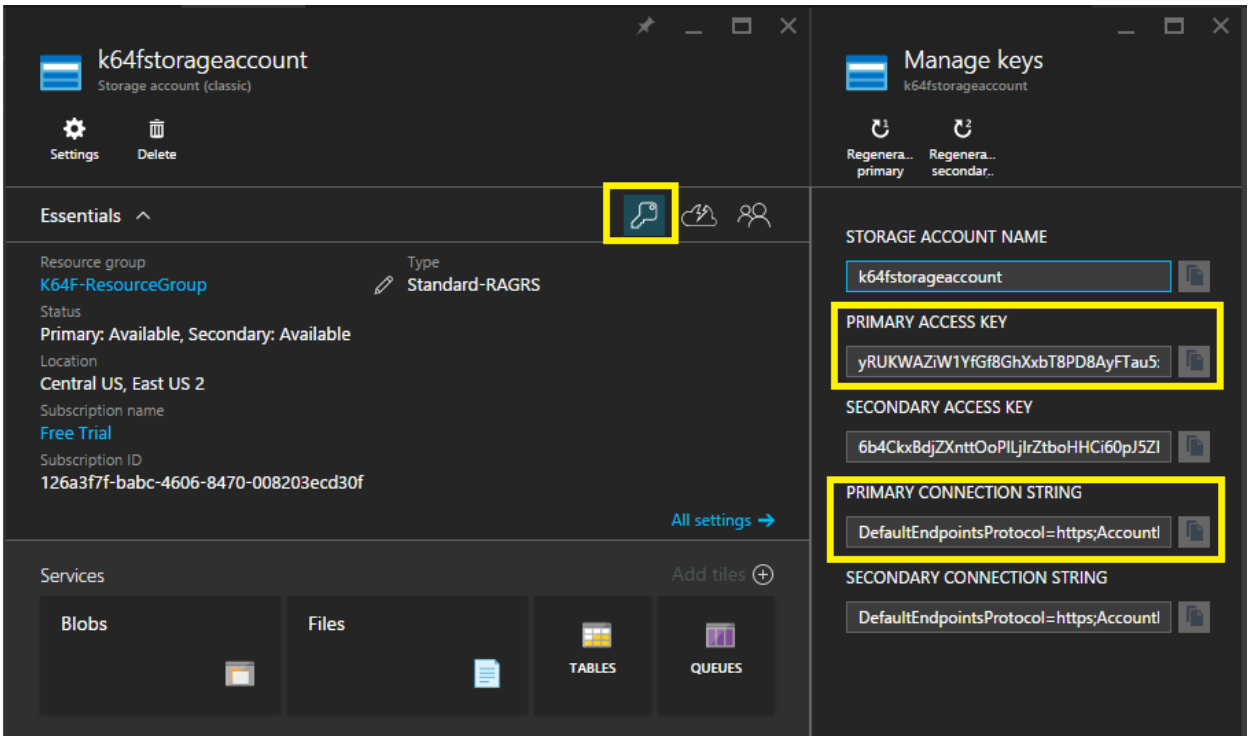



*Step 4 :* Enter the name of your choice (i.e. "*mystorageaccountname*" for the account name and select your resource group, subscription,... then click on "Create"






*Step 5 :* Once the account is created, find it in the resources blade and write down the primary connection string for it to configure the worker role




## 2.5 Create a new device identity in the IoT Hub

***To connect your device to the IoT Hub instance, you need to generate a unique identity and connection string. IoT Hub does that for you. Here we use the [Device Explorer tool](#) (runs only on Windows for now) to create a new device identity .***

### 2.5.1 Step of create device identity

*Step 1 :* Getting Device Explorer

You can either download a pre-built version of Device Explorer or build it yourself.

link: Downloads

a) Download a pre-built version of the Device Explorer application

→Scroll down for SetupDeviceExplorer.msi

b) Build the Device Explorer application

→ Scroll down for Source code(zip) .

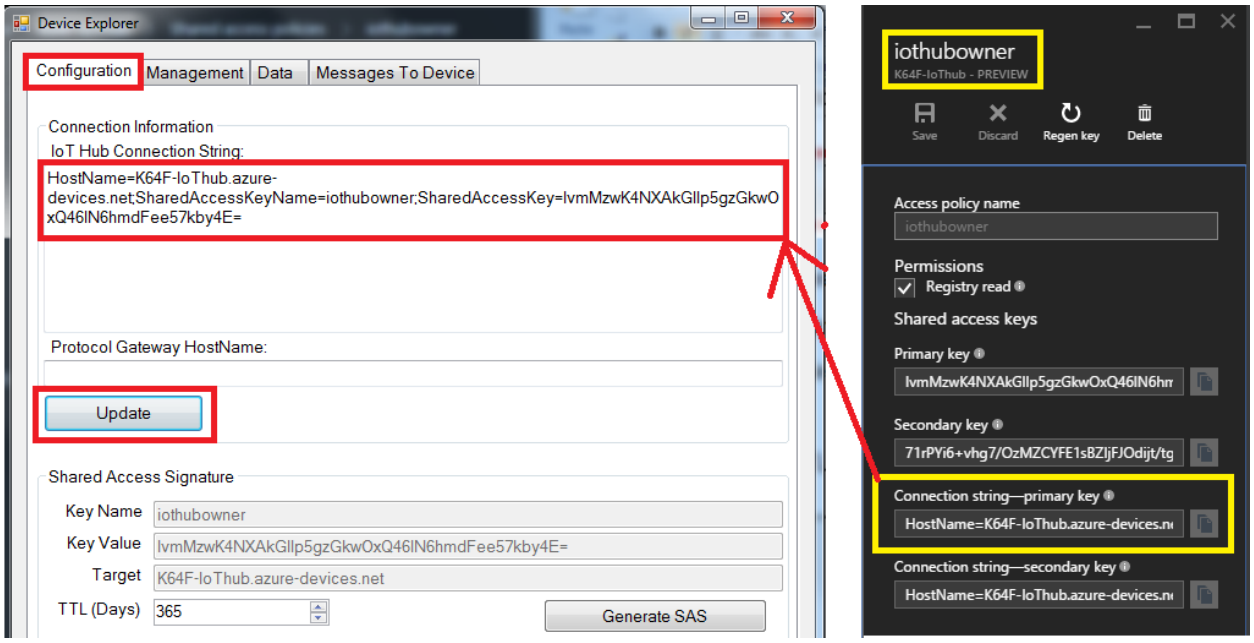
→ To open the **tools\DeviceExplorer\DeviceExplorer.sln** file in your local copy of this repository (azure-iot-sdks) in Visual Studio 2015. Then build and run the solution.

*Step 2 :* Configure an IoT Hub connection

Execute Device Explorer , In the **Configuration** tab, add the connection string for your IoT Hub.

The IoT Hub connection string is what we created at chapter 2.1.2 step 6 .

Then click **Update**.

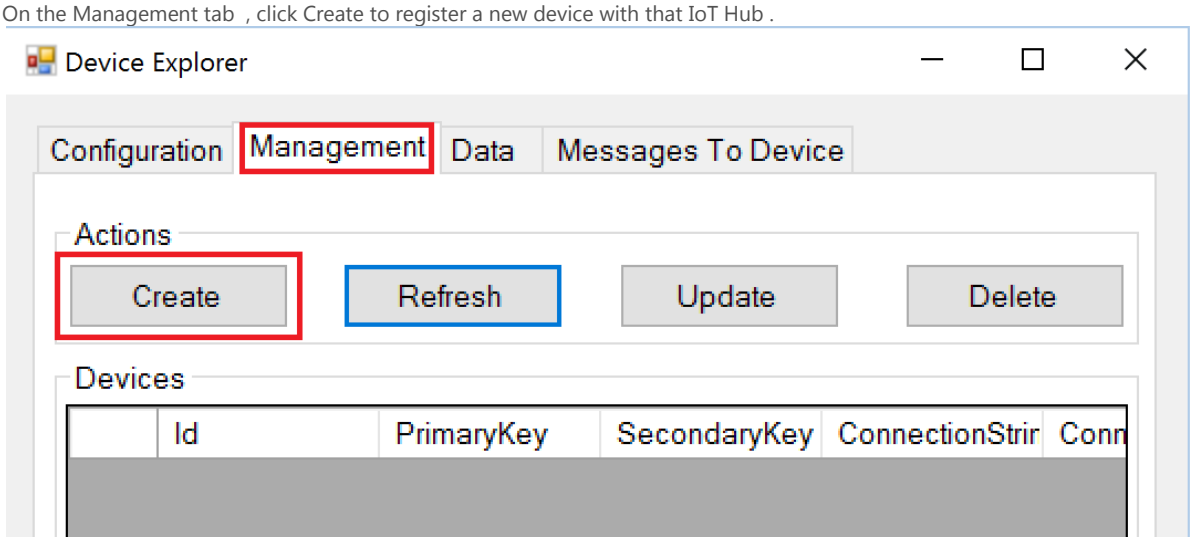



## *Step 3 :* Create device

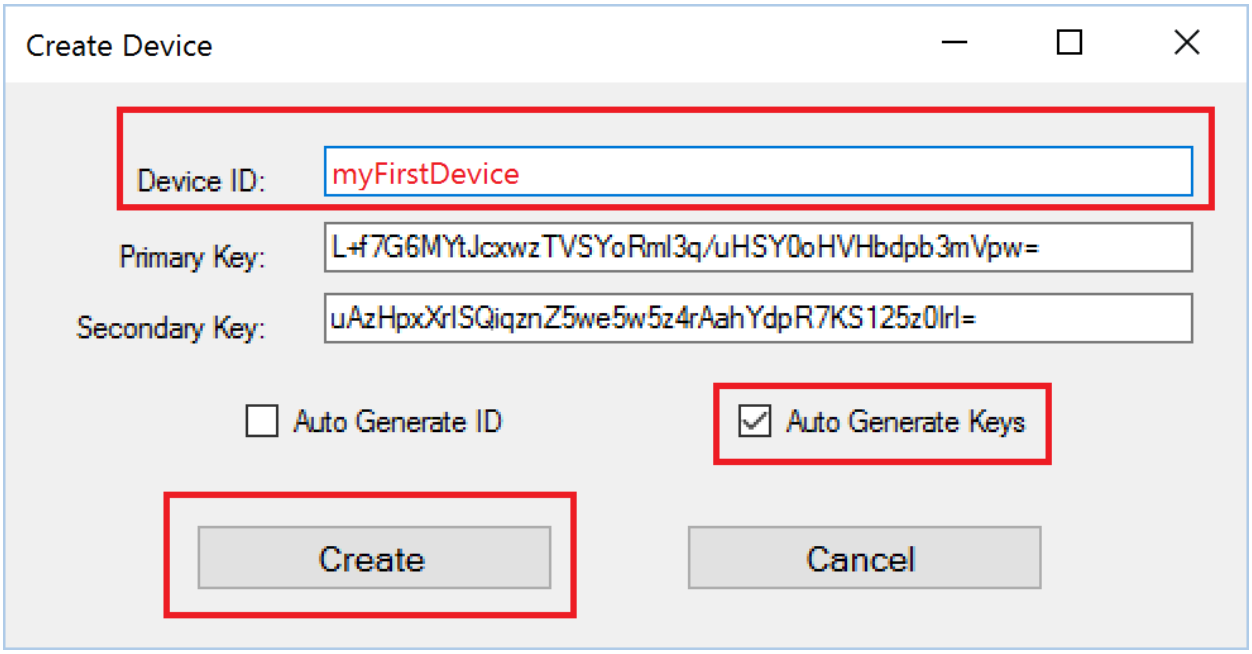
Creating a device adds device details to the device identity registry**. IoT Hub uses this information to generate <u>a device-specific connection string that enables the device to connect to the IoT hub</u>.**

a) Click the Management tab

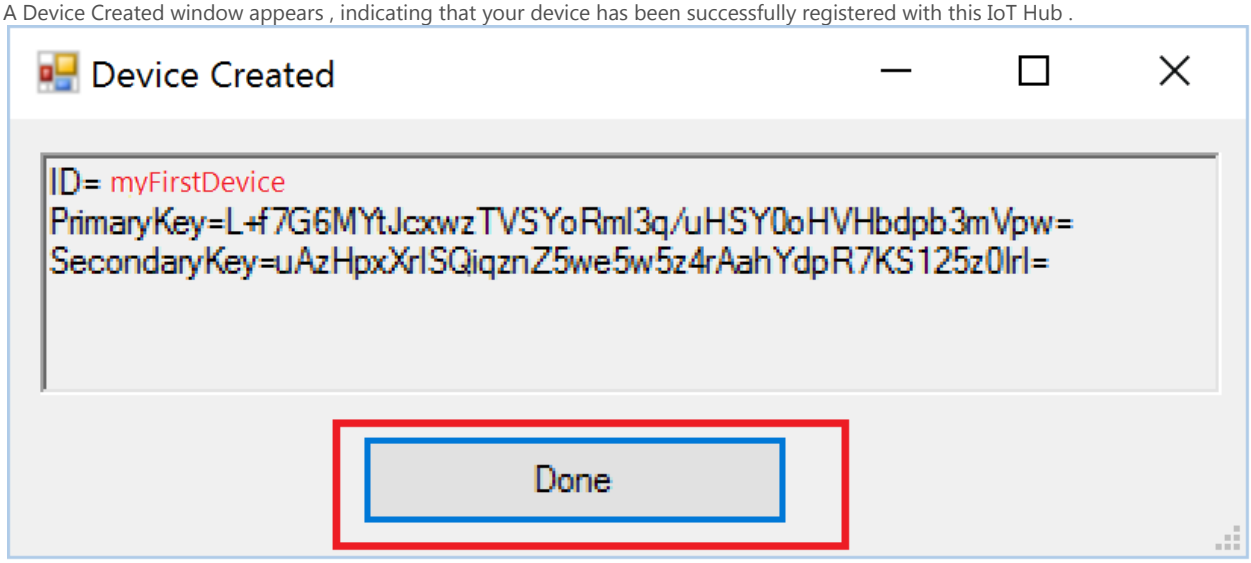
b) On the Management tab , click Create to register a new device with that IoT Hub .




c) The **Create Device** dialog appears . In the **Deivce ID** field , type a unique name for your device or select **Auto Generate ID** to generate a unique ID instead . In this demo please type **"myFirstDevice"** that will be used in FRDM-K64F platform . Then click **Create** .




d) A Device Created window appears , indicating that your device has been successfully registered with this IoT Hub .

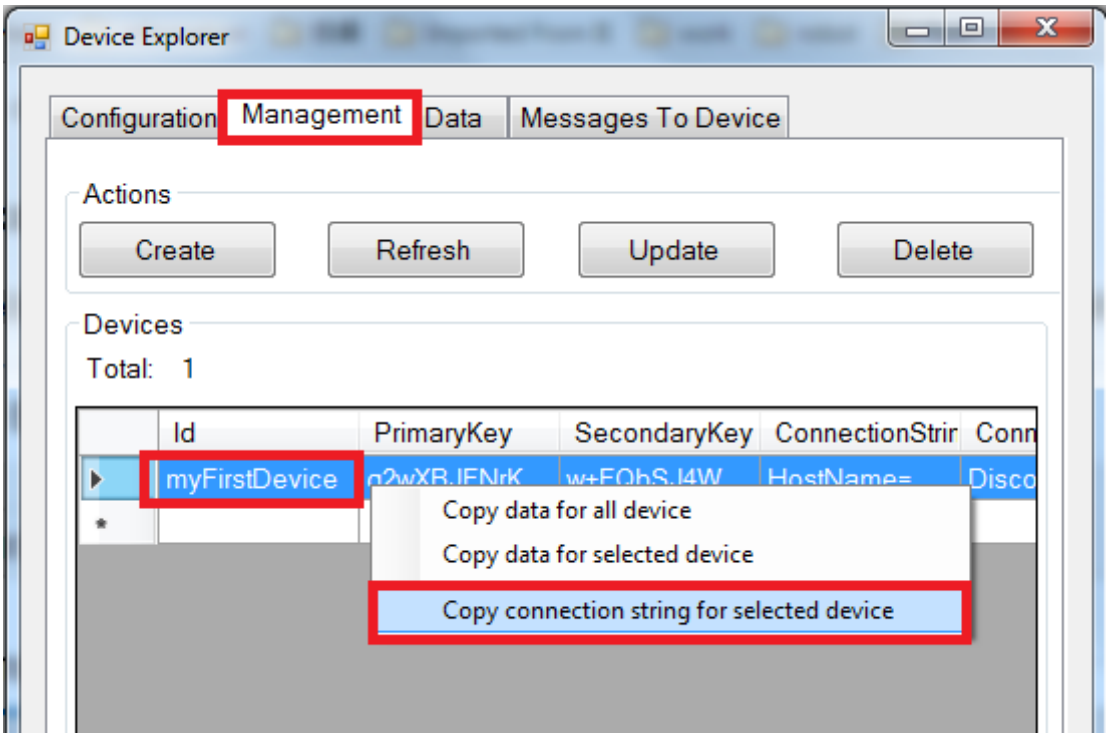



*Step 4 :* Get device connection string

Use the mouse right click for context menu for the selected device and select "Copy connection string for selected device" of options . Then please save the connection string that you copied .

Ex. Device connection string -->

**HostName=Mars-Surface.azure-devices.net;DeviceId=myFirstDevice;SharedAccessKey=q2wXBJFNrKQF6t4pPyirPQF3H3UjlnXv1e/Hasy38yU=**




## 2.6 Connect the device

1. Connect the board FRDM-K64F to your network using an Ethernet cable. This step is required, as the sample depends on Internet access.
2. Plug the device into your computer using a micro-USB cable. Be sure to attach the cable to the correct USB port on the device (the CMSIS-DAP USB one, see here to find which one it is).
3. Follow the instructions on the mbed handbook to setup the serial connection with your device from your development machine. If you are on Windows, install the Windows serial port drivers located here.

## 2.7 Program IoT Client -- FRDM-K64F

### 2.7.1 FRDM-K64F

The Freescale Freedom-K64F is an ultra-low-cost development platform for Kinetis K64, K63, and K24 MCUs.

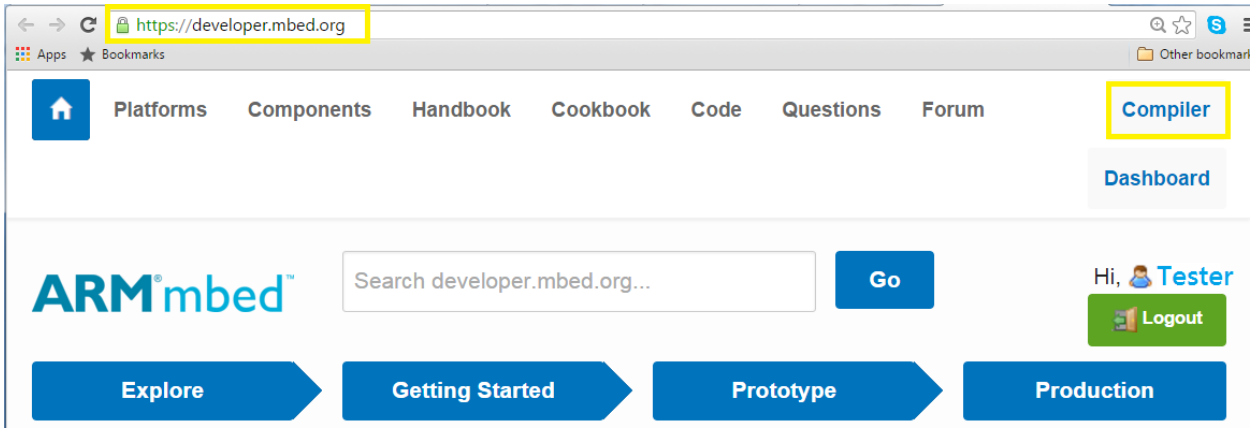
- Form-factor compatible with the Arduino R3 pin layout
- Peripherals enable rapid prototyping, including a 6-axis digital accelerometer and magnetometer to create full eCompass capabilities, a tri-colored LED and 2 user push-buttons for direct interaction, a microSD card slot, and connectivity using onboard Ethernet port and headers for use with Bluetooth® and 2.4 GHz radio add-on modules
- Microsoft has certified the ARM® mbed Enabled™ Freescale® FRDM-K64F development board for Internet of Things (IoT) products
    - ***More detail please refer to below Microsoft link .***

http://www.freescale.com/FRDM-K64F

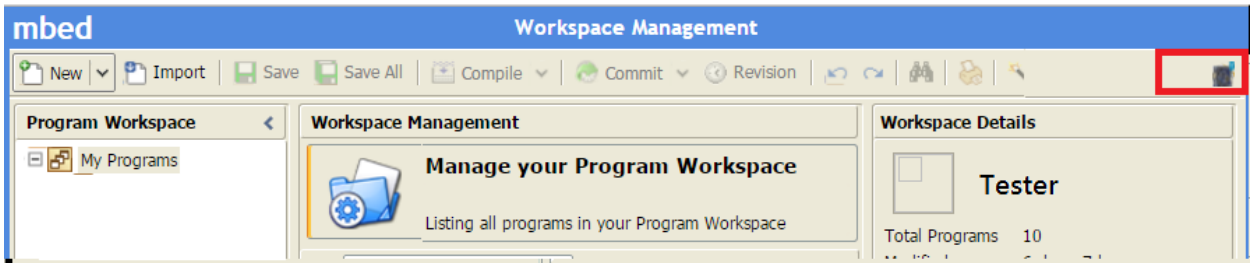
### 2.7.2 Create mbed project and import the sample code

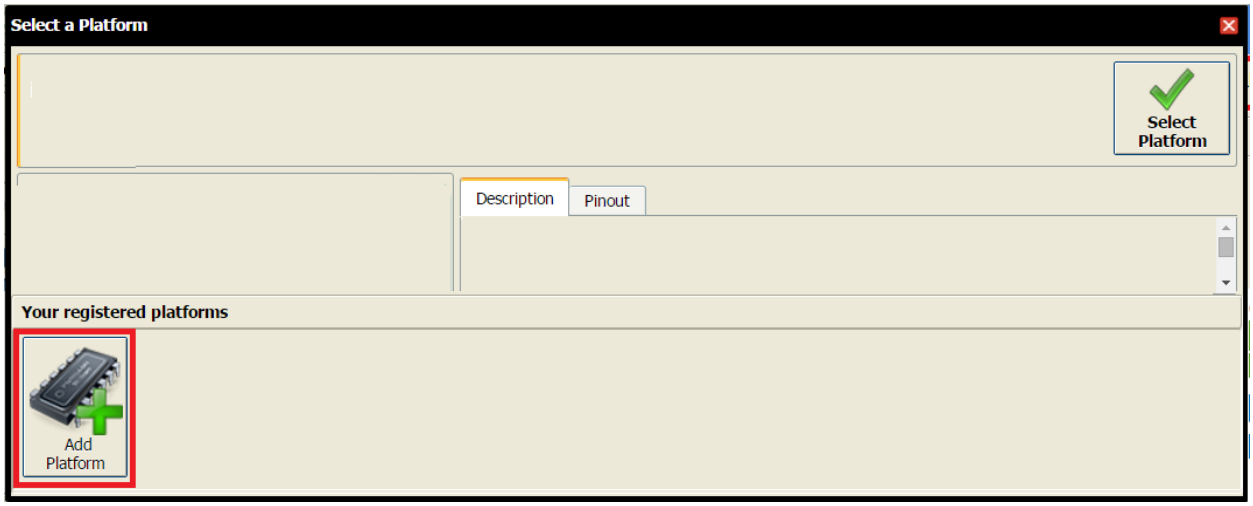
*Step 1 :* In your web browser, go to the mbed.org developer site.

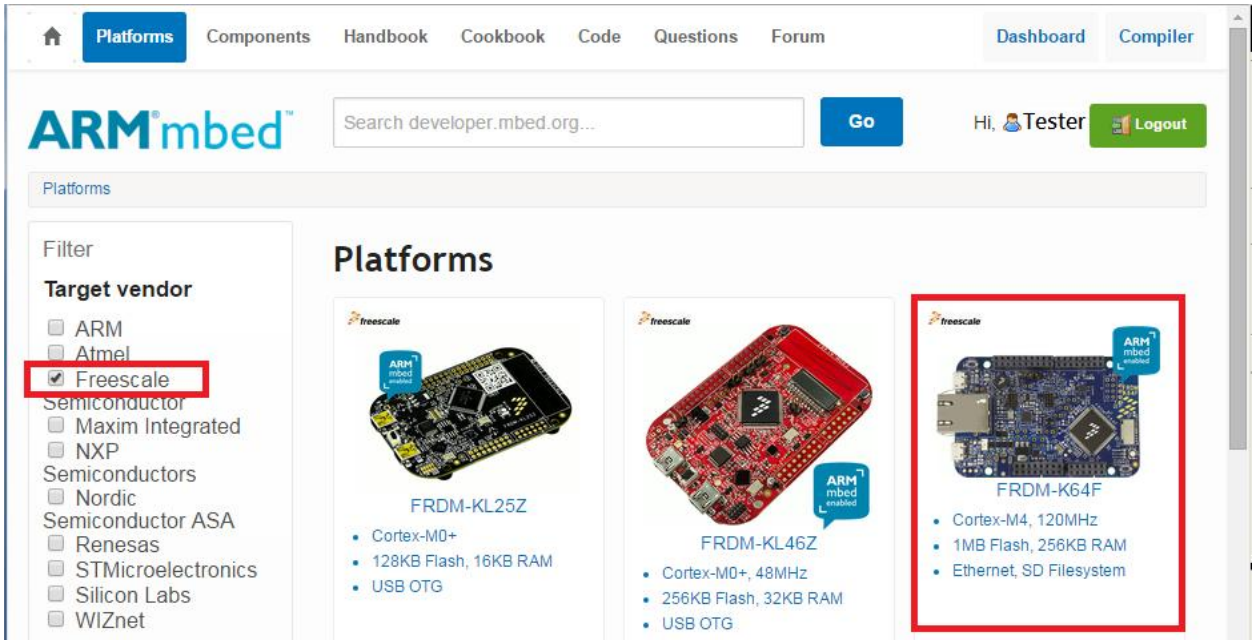
If you haven't signed up, you will see an option to create a new account (it's free). Otherwise, log in with your account credentials. Then click on **Compiler** in the upper right-hand corner of the page. This should bring you to the Workspace Management interface.

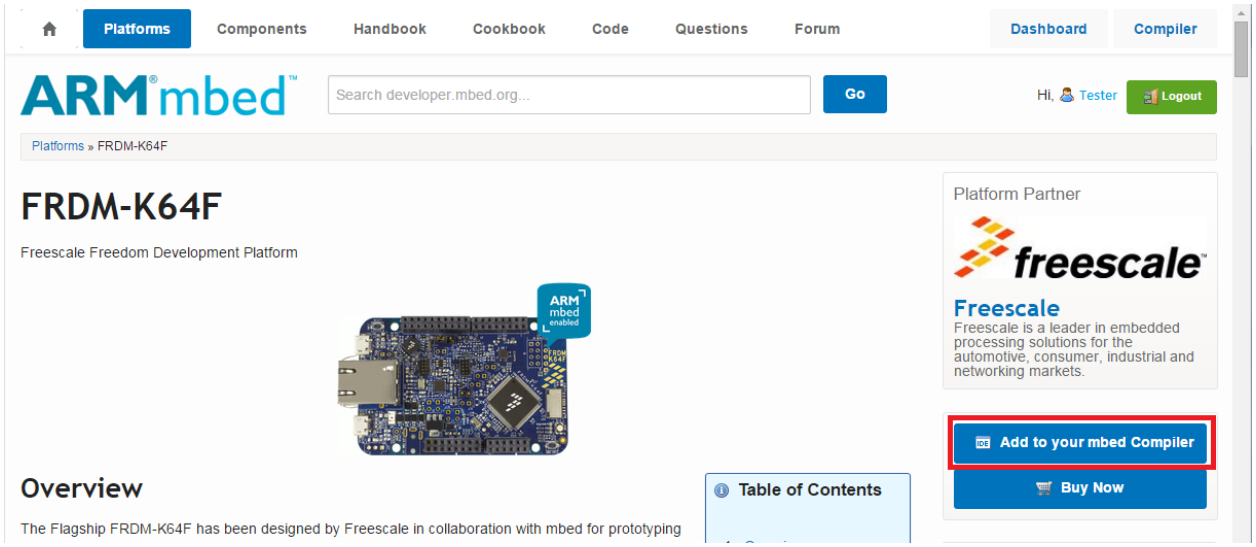


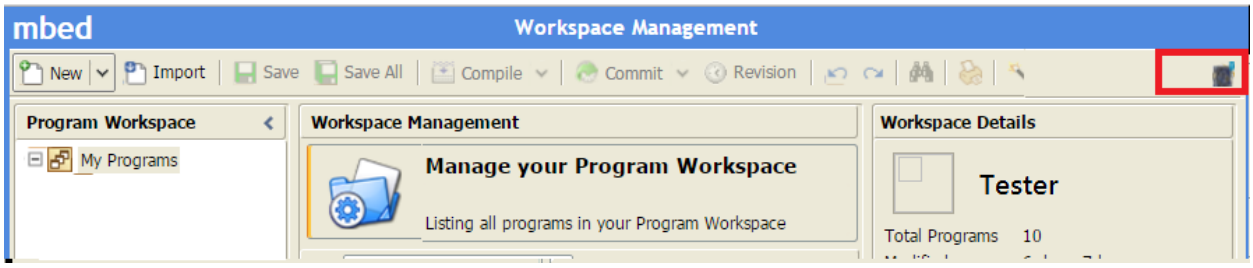

*Step2 :* Make sure the hardware platform you're using appears in the upper right-hand corner of the window, or click the icon in the right-hand corner to select your hardware platform.

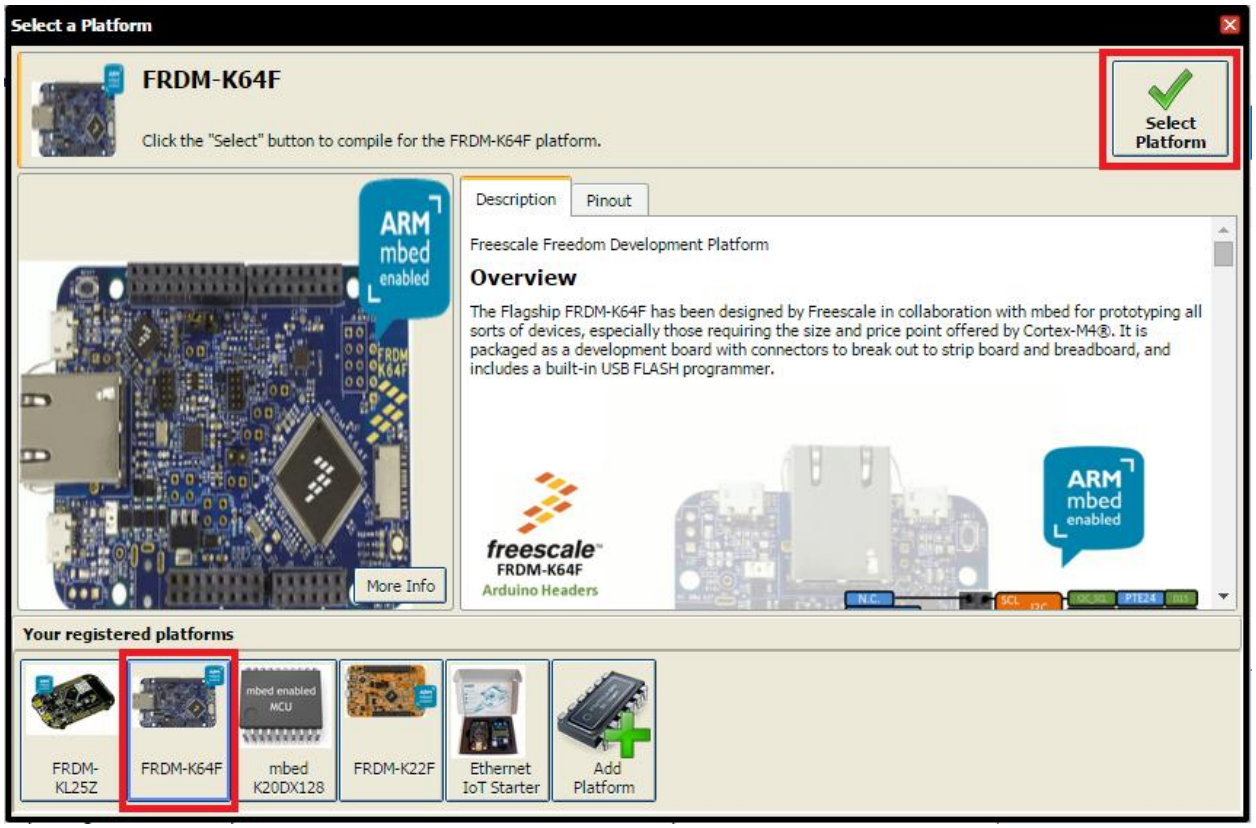


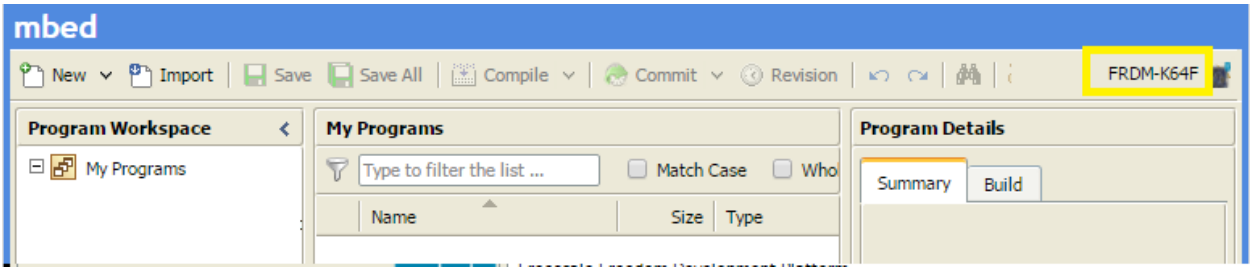




Platforms
Components
Handbook
Cookbook
Code
Questions
Forum
Dashboard
Compiler
ARM mbed
Search developer.mbed.org...
Go
Hi, Tester
Logout
Platforms
Filter
Target vendor
ARM
Atmel
Freescale Semiconductor
Maxim Integrated
NXP Semiconductors
Nordic Semiconductor ASA
Renesas
STMicroelectronics
Silicon Labs
WIZnet
Platforms
FRDM-KL25Z
Cortex-M0+
128KB Flash, 16KB RAM
USB OTG
FRDM-KL46Z
Cortex-M0+, 48MHz
256KB Flash, 32KB RAM
USB OTG
FRDM-K64F
Cortex-M4, 120MHz
1MB Flash, 256KB RAM
Ethernet, SD Filesystem

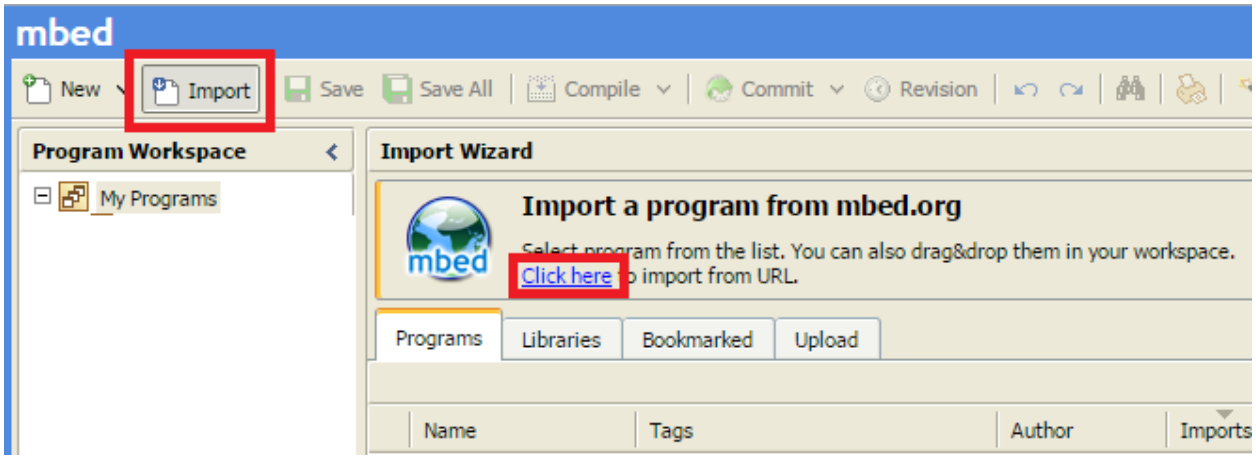

Platforms
Components
Handbook
Cookbook
Code
Questions
Forum
Dashboard
Compiler
ARM mbed
Search developer.mbed.org...
Go
Hi, Tester
Logout
Platforms » FRDM-K64F
FRDM-K64F
Freescale Freedom Development Platform
Platform Partner
freescale
Freescale
Freescale is a leader in embedded processing solutions for the automotive, consumer, industrial and networking markets.
Add to your mbed Compiler
Buy Now
Overview
The Flagship FRDM-K64F has been designed by Freescale in collaboration with mbed for prototyping
Table of Contents


mbed
Workspace Management
New
Import
Save
Save All
Compile
Commit
Revision
Program Workspace
My Programs
Workspace Management
Manage your Program Workspace
Listing all programs in your Program Workspace
Workspace Details
Tester
Total Programs 10

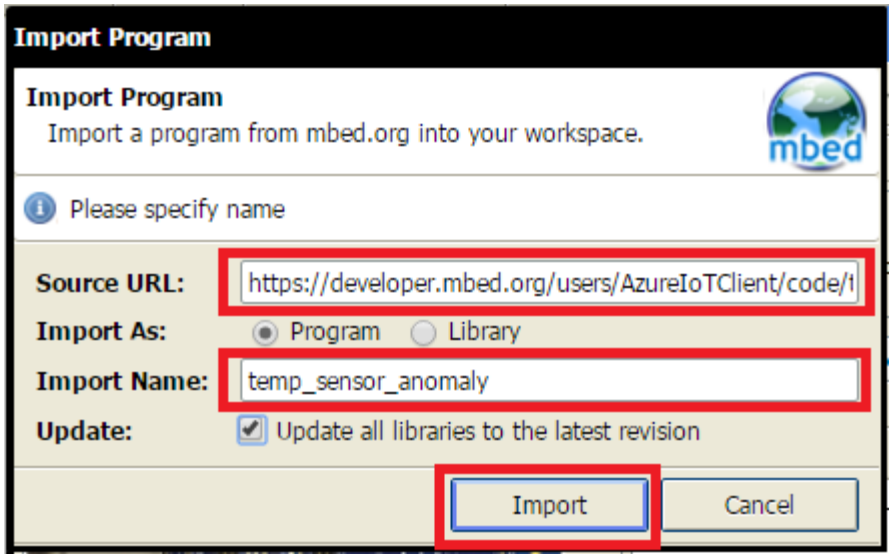


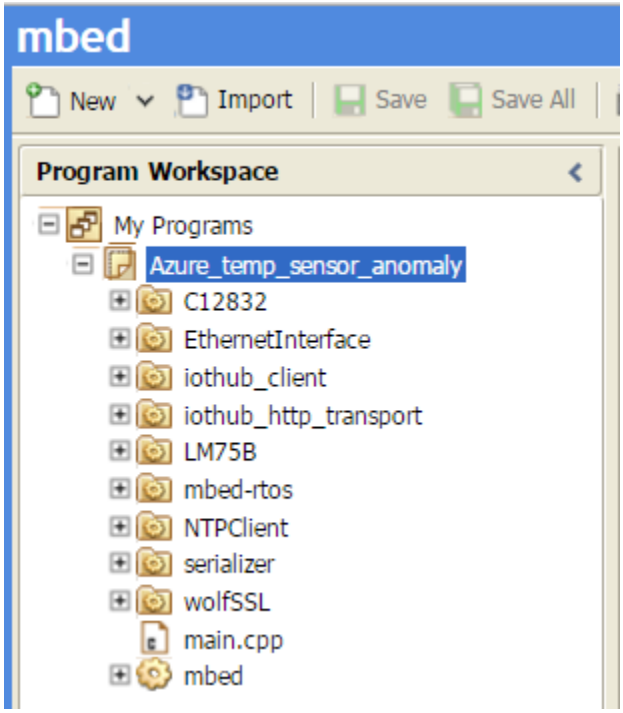

*Step3 :* Click **Import** on the main menu. Then click the **Click here** to import from URL link next to the mbed globe logo.




*Step4 :* In the popup window, enter the link for the sample code
https://developer.mbed.org/users/AzureIoTClient/code/temp_sensor_anomaly/



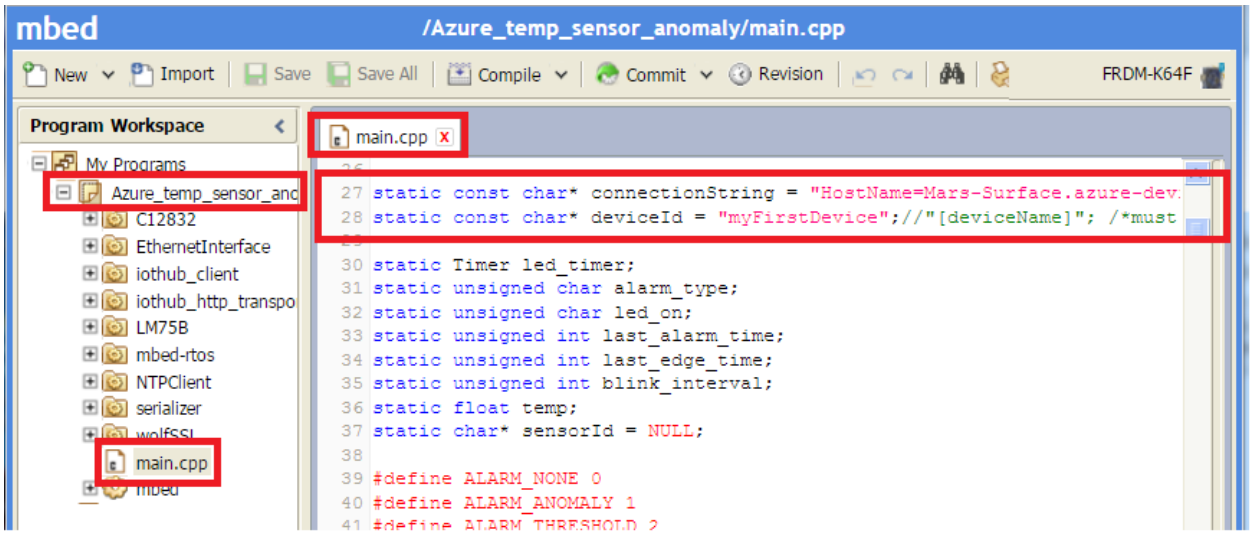

*Step 5 :* You can see in the mbed compiler that importing this project imported various libraries. Some are provided and maintained by the Azure IoT team (azureiot_common, iothub_client, iothub_http_transport, proton-c-mbed), while others are third party libraries available in the mbed libraries catalog .



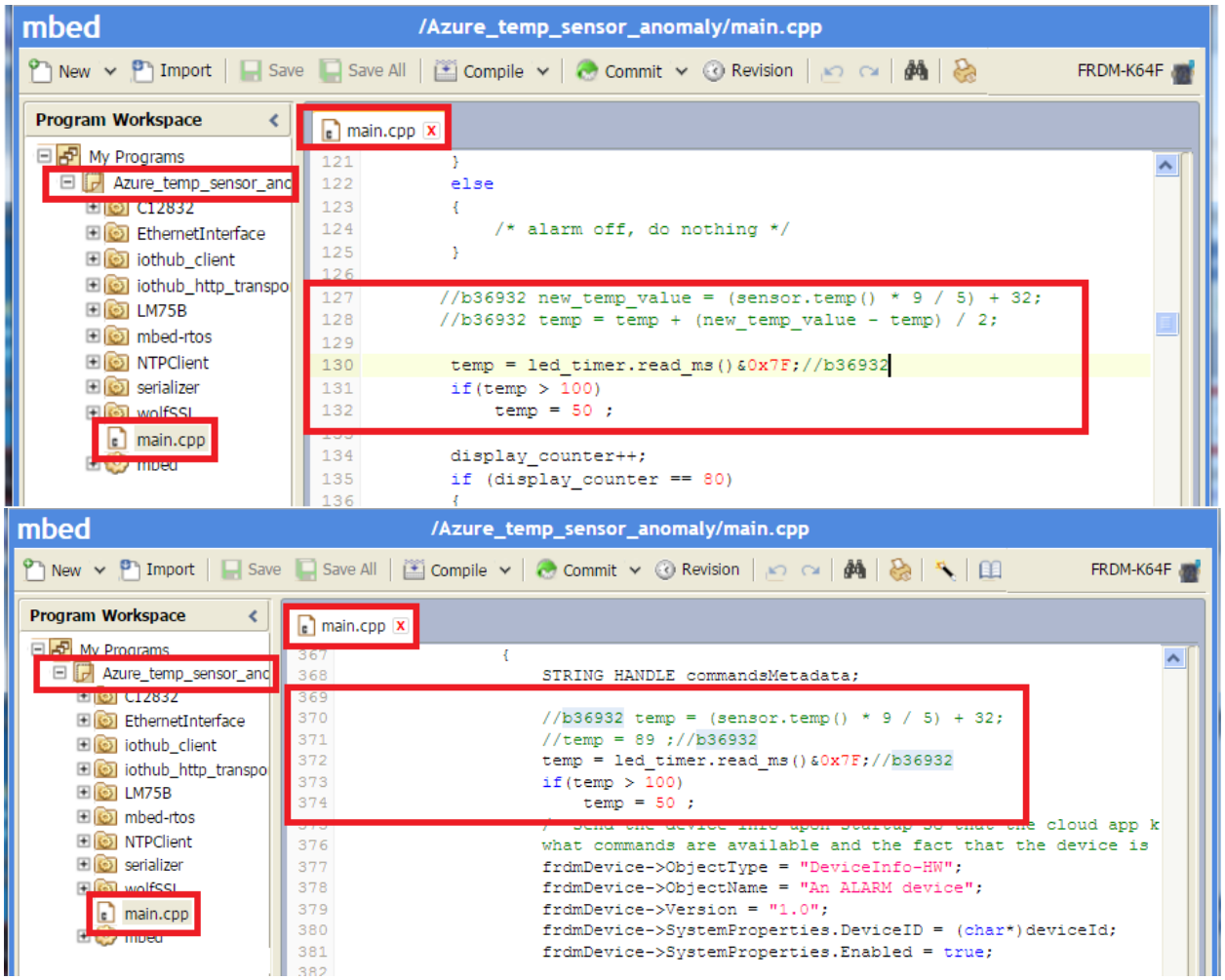

*Step 6 :* In the temp_sensor_anomaly\main.cpp file, find and replace values in the following lines of code with your

a) Device connection string that is what we created at Chapter 2.5.1 Step 4 .

b) device Id that is what we created at Chapter 2.5.1 Step 3 . (" myFirstDevice " is what we used on this demo)

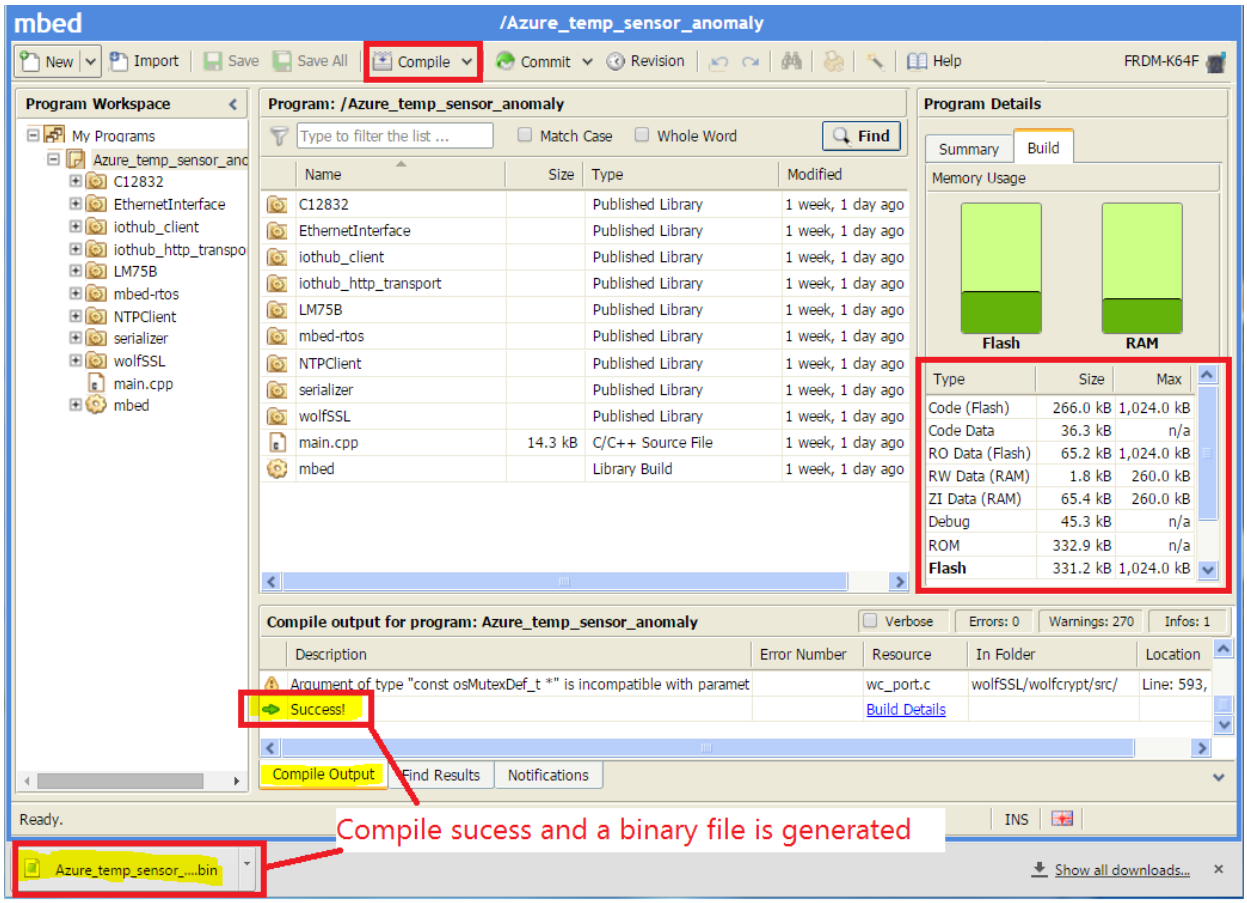


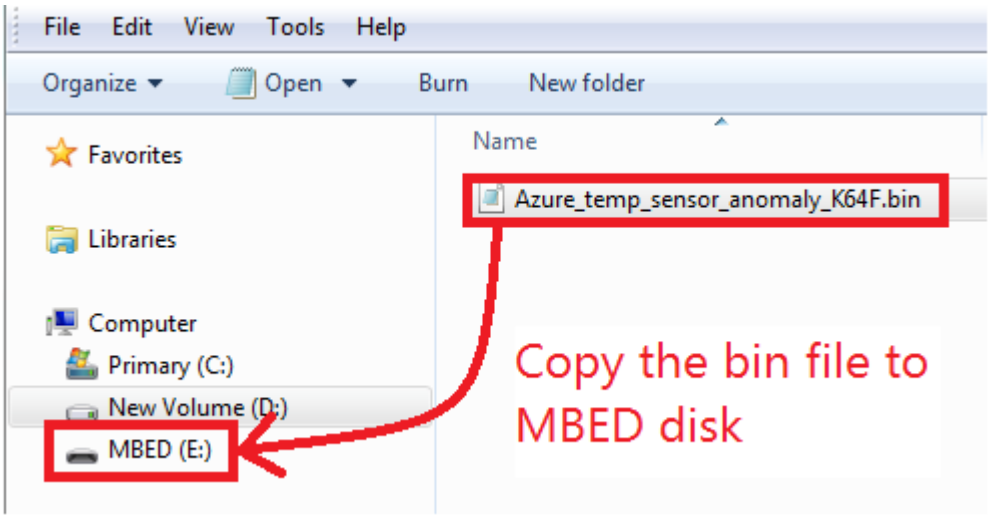

*Step7 :* In case we don't have "mbed Application shield" then we need to modify the software to generate a random number to instead temperature sensor .

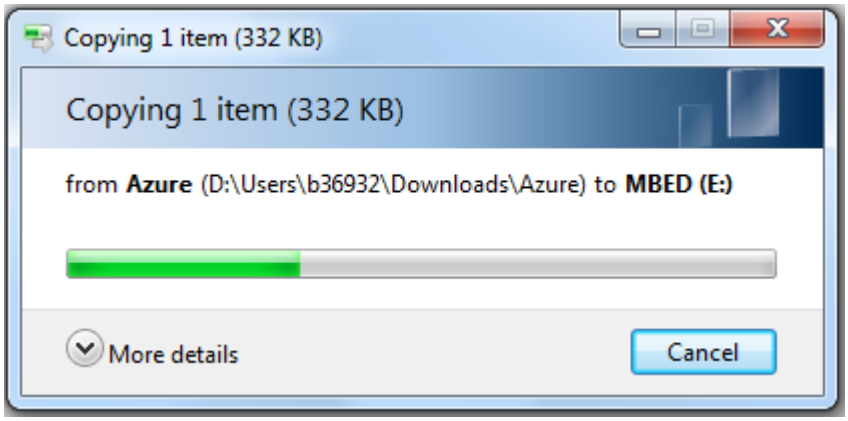



*Step 8 :* Click **Compile** to build the program. If the build generates errors, fix them before proceeding.

*Step 9 :* If the build is successful, a .bin file with the name of your project is generated. Copy the .bin file to the device.

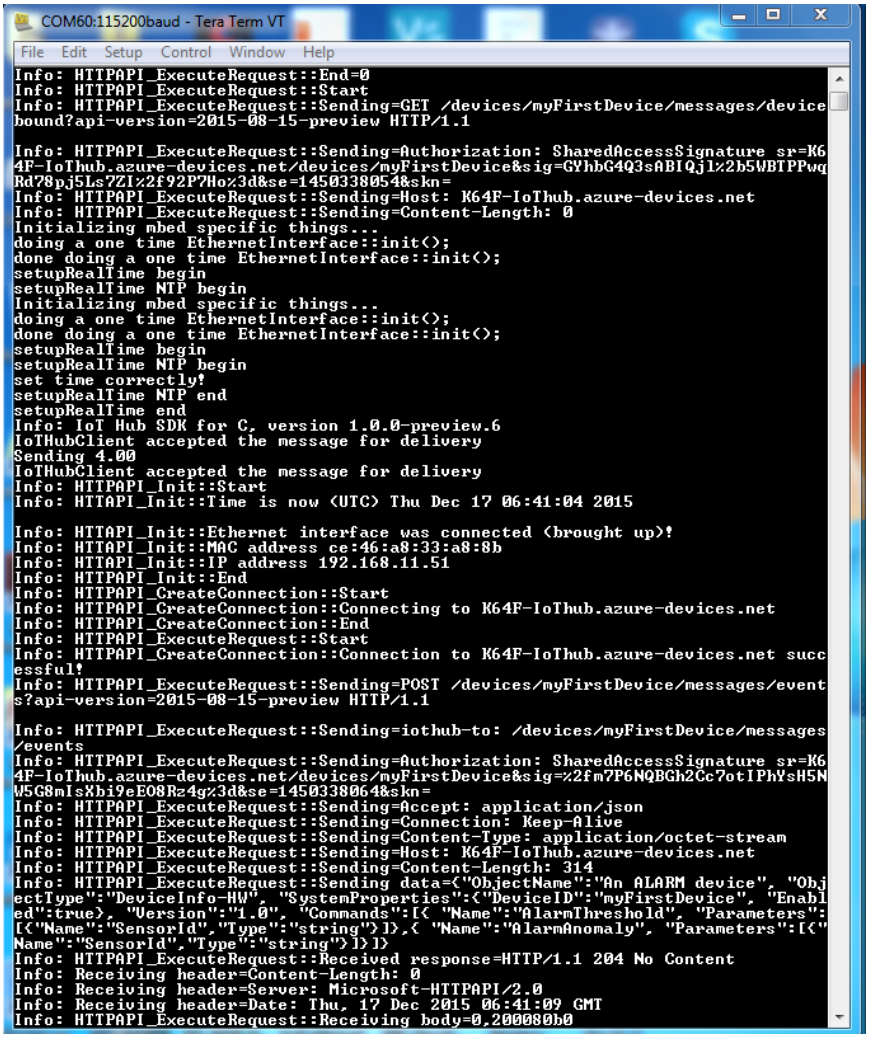

mbed
/Azure_temp_sensor_anomaly
New
Import
Save
Save All
Compile
Commit
Revision
Help
FRDM-K64F
Program Workspace
My Programs
Azure_temp_sensor_anc
C12832
EthernetInterface
iothub_client
iothub_http_transpo
LM75B
mbed-rtos
NTPClient
serializer
wolfSSL
main.cpp
mbed
Program: /Azure_temp_sensor_anomaly
Type to filter the list ...
Match Case
Whole Word
Find
Name
Size
Type
Modified
C12832 Published Library 1 week, 1 day ago
EthernetInterface Published Library 1 week, 1 day ago
iothub_client Published Library 1 week, 1 day ago
iothub_http_transport Published Library 1 week, 1 day ago
LM75B Published Library 1 week, 1 day ago
mbed-rtos Published Library 1 week, 1 day ago
NTPClient Published Library 1 week, 1 day ago
serializer Published Library 1 week, 1 day ago
wolfSSL Published Library 1 week, 1 day ago
main.cpp 14.3 kB C/C++ Source File 1 week, 1 day ago
mbed Library Build 1 week, 1 day ago
Program Details
Summary
Build
Memory Usage
Flash
RAM
Type Size Max
Code (Flash) 266.0 kB 1,024.0 kB
Code Data 36.3 kB n/a
RO Data (Flash) 65.2 kB 1,024.0 kB
RW Data (RAM) 1.8 kB 260.0 kB
ZI Data (RAM) 65.4 kB 260.0 kB
Debug 45.3 kB n/a
ROM 332.9 kB n/a
Flash 331.2 kB 1,024.0 kB
Compile output for program: Azure_temp_sensor_anomaly
Verbose
Errors: 0
Warnings: 270
Infos: 1
Description
Error Number
Resource
In Folder
Location
Argument of type "const osMutexDef_t *" is incompatible with paramet
wc_port.c
wolfSSL/wolfcrypt/src/
Line: 593,
Success!
Build Details
Compile Output
Find Results
Notifications
Ready.
INS
Compile sucess and a binary file is generated
Azure_temp_sensor_....bin
Show all downloads...


File
Edit
View
Tools
Help
Organize
Open
Burn
New folder
Favorites
Libraries
Computer
Primary (C:)
New Volume (D:)
MBED (E:)
Name
Azure_temp_sensor_anomaly_K64F.bin
Copy the bin file to MBED disk


Copying 1 item (332 KB)
Copying 1 item (332 KB)
from Azure (D:\Users\b36932\Downloads\Azure) to MBED (E:)
More details
Cancel

*Step 10 :* After firmware update , unplug and plug the FRDM-K64F board USB again . Then connect to the device using an serial terminal client application, such as PuTTY / Tera Term. Baud rate is 115200 .

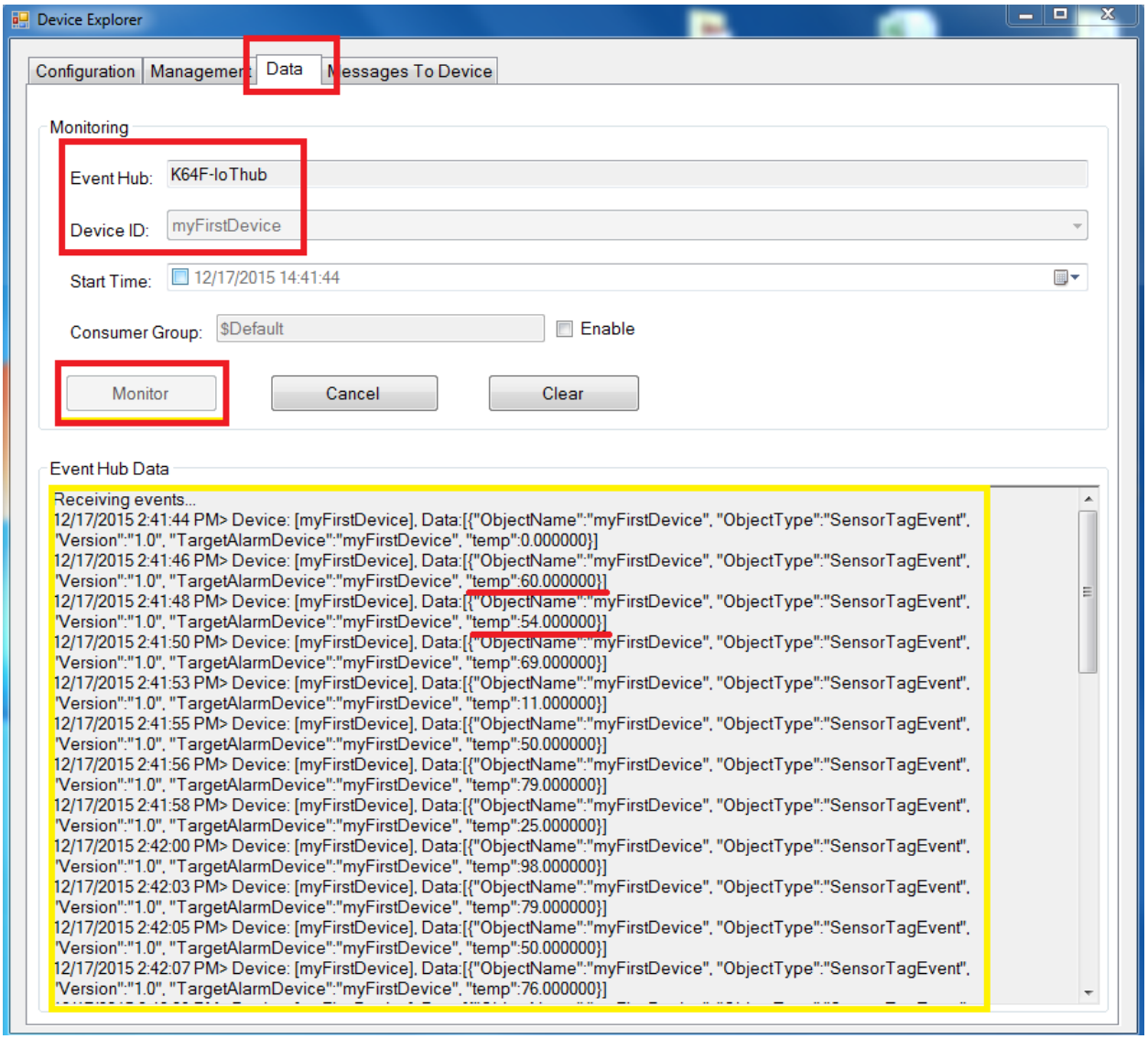
The program starts executing. You may have to reset the board (press on the board's reset button) .



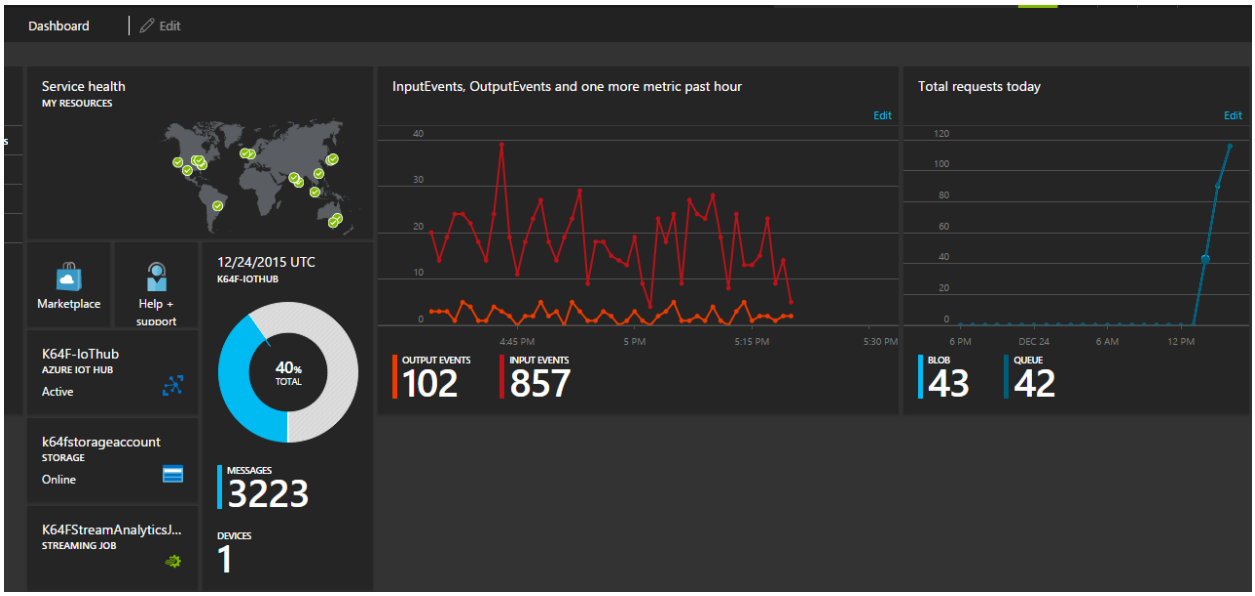

## 2.8 Monitor IoT Client information

### 2.8.1 Receive IoT Client data by “Device Explorer”

- In the Device Explorer **Data** tab, select the device name you created from the drop-down list of device IDs and leave the other fields with their default values for now.
- Click **Monitor**.




## 2.8.2 The data on Azure



[ALERT ACTIVATED] - Output Events GreaterThan 3 (Count) in the last 30 minutes...

FILE MESSAGE

Thu 12/24/2015 3:51 PM

Microsoft Azure Alerts <alerts-noreply@mail.windowsazure.com>

**[ALERT ACTIVATED] - Output Events GreaterThan 3 (Count) in the last 30 minutes**

To David Chen

Bing Maps Unsubscribe + Get more apps

Microsoft

Azure

Dear Customer,

'Output Events GreaterThan 3 (Count) in the last 30 minutes' was activated for Streaming Job: K64FStreamAnalyticsJob (k64f-resourcegroup)

**Microsoft Azure Alerts** [ALERT ACTIVATED] - Output Events GreaterThan 1 (Count) in the last 60 minutes

Microsoft

Azure

Dear Customer,

## ⚠ 'Output Events GreaterThan 3 (Count) in the last 30 minutes' was activated for Streaming Job: K64FStreamAnalyticsJob (k64f-resourcegroup)

You can view more details for this alert in the Microsoft Azure Management Portal.

**RULE NAME:** StreamAnalyticsAlert_OverTemp

**RULE DESCRIPTION:**

**SERVICE:** Streaming Job: K64FStreamAnalyticsJob (k64f-resourcegroup)

**METRIC:** Total Output Events

**ALERT ACTIVATED TIME (UTC):** 12/24/2015 7:51:22 AM

**SUBSCRIPTION NAME:** Free Trial