

# Azure IoT テクニカルワークショップ

Global Blackbelt Sales Microsoft Corporation

Sept-2018

# Table of Contents

このハンズオンについて	1
概要	1
必要な準備	1
ハンズオンで構築するIoTサービスのについて	1
演習: Cloud Shell (Bash)の設定	1
タスク1: Cloud Shellの初期設定	1
演習: Azure環境の準備	3
Azureポータルへのアクセス	3
リソースグループの作成	4
演習: IoHubとデバイスの接続	5
IoT Hubの作成	6
IoT Hubへのデバイス登録	14
IoT Hubデバイスクライアントの開発	17
IoT Hubでのデータ受信確認	19
演習: Time Series Insightsによるデータの可視化	20
Time Series Insightsの作成	20
Time Series Insightsの設定	24
Time Series Insightsでのデータ表示とカスタマイズ	28
演習: Stream Analyticsによるリアルタイムデータ加工	31
Stream Analyticsの作成	32
Stream Analyticsの入力設定	34
Stream Analyticsの出力設定（ストレージへの出力）	37
Stream Analyticsのクエリ設定（ストレージへの出力）	43
Stream Analyticsの出力設定（SQLデータベースへの出力）	47
Stream Analyticsのクエリ設定（SQLデータベースへの出力）	56
演習: Power BIでのデータ表示	59
Stream Analyticsの出力設定（PowerBIへの出力）	59
演習: Azure IoT Edge デバイスの作成	72
タスク1: Linux VMのデプロイ	72
タスク2: コンテナランタイムのインストール	75
タスク3: IoT Edgeデバイスの追加	77
タスク4: IoT Edgeランタイムのインストール	78
タスク5: IoT Edgeデバイスにシミュレーションモジュールを追加	80

# このハンズオンについて

## 概要

このハンズオンでは、Microsoft Azure IoTサービスを使用して IoTアプリケーション（データの収集／加工／視覚化）をチュートリアル形式で構築します。 Azure IoTアーキテクチャをハンズオン形式で理解し、実現するシステムの構成をイメージできるようになることを目的とします。

## 必要な準備

Webブラウザから無線LANで接続可能なPCを必ず準備してください。

下記リンクから無償環境を取得いただけます。

<https://azure.microsoft.com/ja-jp/free/>

Power BIの環境を事前に取得の上、ご参加いただきますようお願いします。  
下記リンクから評価版の環境を取得いただけます。

<https://powerbi.microsoft.com/ja-jp/documentation/powerbi-service-self-service-signup-for-power-bi/>

下記リンクに本ハンズオンで使用するコードやコマンドを掲載していますのでご活用ください。

<https://msjplot.blob.core.windows.net/docs/Snippet.txt>

## ハンズオンで構築するIoTサービスのついて

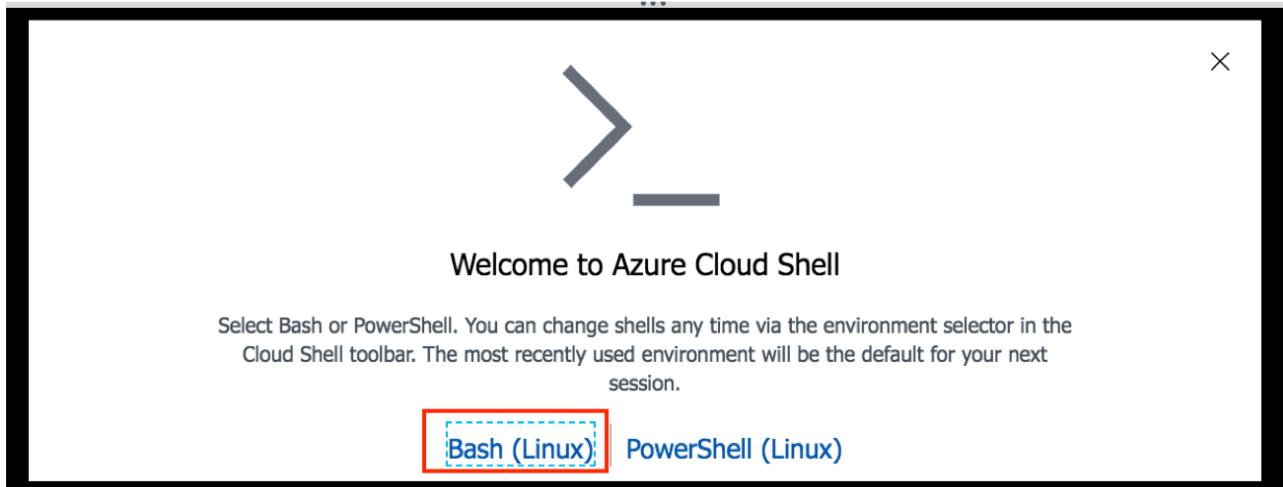
<図>

## 演習: Cloud Shell (Bash)の設定

Cloud Shell (Bash) 環境を設定します。

### タスク1: Cloud Shellの初期設定

1. [>] ボタンをクリックしてCloud Shellを開始します。
2. Azureポータル画面下部のダイアログで [Bash(Linux)] を選択します。



3. Cloud Shell用のストレージアカウントに利用するサブスクリプションを選択し[Create storage]ボタンをクリックします。



4. ストレージの作成が完了し、Bashが起動されることを確認します。

```
Bash | ⚡ ? 🚙 🔍 { }

Resource group: cloud-shell-storage-southeastasia
Storage account: cs-[REDACTED]b
File share: cs-[REDACTED]6

Initializing your account for Cloud Shell...\
Requesting a Cloud Shell.Succeeded.
Connecting terminal...

Welcome to Azure Cloud Shell

Type "az" to use Azure CLI 2.0
Type "help" to learn about Cloud Shell

kei@Azure:~$ █
```

5. VMにSSHでログインする場合は、接続文字列を入力してログインします。

```

Bash | ⌁ ? ⌂ ⌂ {} ⌂

Welcome to Azure Cloud Shell

Type "az" to use Azure CLI 2.0
Type "help" to learn about Cloud Shell

kei@Azure:~$ ssh myadmin@[IP addr] ← コピーした接続文字列
The authenticity of host '[IP addr]' ([IP addr])' can't be established.
ECDSA key fingerprint is SHA256:aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa/0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[IP addr]' (ECDSA) to the list of known hosts.
myadmin@[IP addr]'s password: ← パスワードを入力 (表示はされない)
Welcome to Ubuntu 18.04.1 LTS (GNU/Linux 4.15.0-1022-azure x86_64)

 * Documentation:  https://help.ubuntu.com

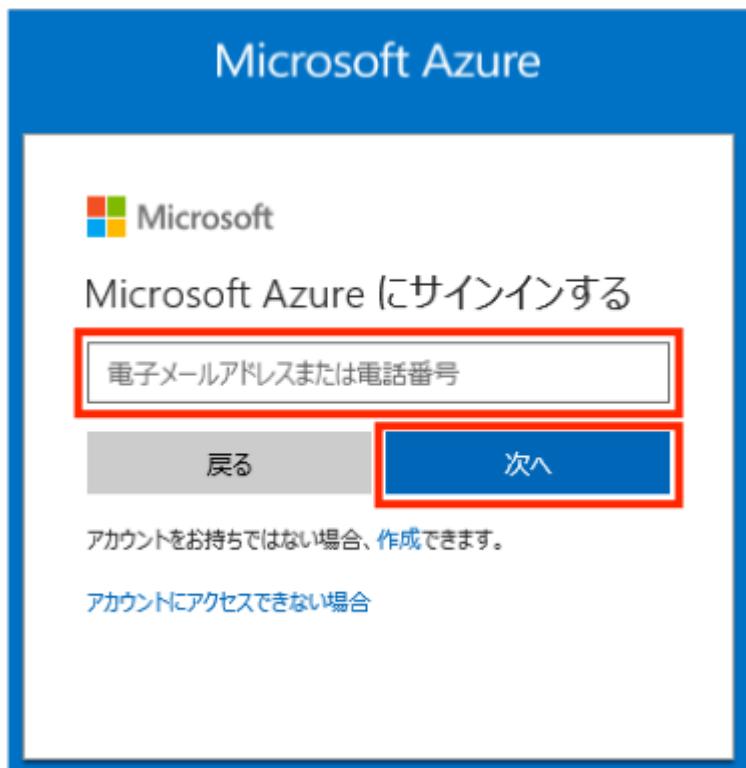
```

## 演習:Azure環境の準備

### Azureポータルへのアクセス

このタスクでは、Azureを操作するために、WebブラウザでAzureポータルにアクセスします。

1. ブラウザでAzure管理ポータルにアクセスします。 <https://portal.azure.com>
2. ログイン画面に取得いただいたマイクロソフトアカウントの[メールアドレス]と[パスワード]を入力して、[サインイン]ボタンをクリックします。



3. 管理ポータルにログインできたことを確認してください。

# リソースグループの作成

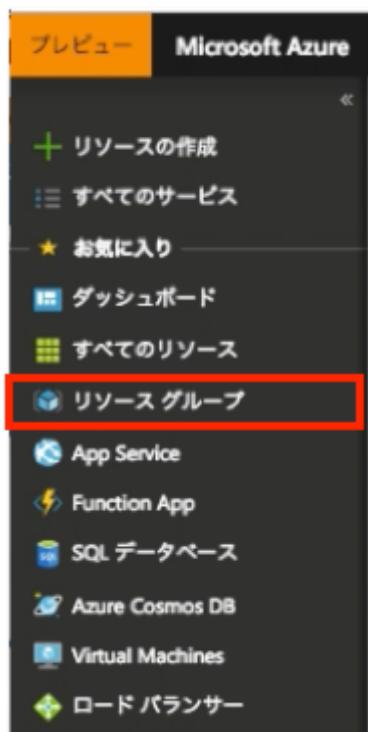
このタスクでは、リソースグループを作成します。リソースグループでは関連するサービスを同一のリソースグループ内にまとめて管理することができます。



Microsoft Azure ダッシュボード画面を表示。サイドメニューで「リソース グループ」が選択されている。右側のリストでは、複数のリソースグループが一覧表示されている。

リソース グループ	場所
intwest*****	東日本
intdemow*****	米国西部
intbandan*****	東日本
wia*****	東日本
*****	東日本
oneashift*****	米国東部
oneashiftorlogin	西日本
extension*****	米国東部

1. サイドメニューの[リソースグループ]をクリックします。



Microsoft Azure ダッシュボード画面を表示。サイドメニューで「リソース グループ」が赤枠で強調されている。

2. [+追加]をクリックします。リソースグループ作成ウィンドウが表示されます。



リソース グループ作成ウィンドウが表示。画面下部には「+ 追加」ボタンが赤枠で強調されている。

3. 下記の項目を設定します。各設定項目を入力後、ウィンドウ下部にある[作成]をクリックします。

Table 1. 設定項目と設定値

設定項目	設定値
リソースグループ名	iothandson 任意の値でも構いません ただし、サブスクリプション内で一意の名前にする必要があります 縁のチェックマークが表示されれば重複していません
サブスクリプション	ご用意いただいたAzureサブスクリプション（デフォルトのまま）
リソースグループの場所	東日本



4. リソースグループ画面の[更新]をクリックします。設定したリソースグループが追加されていることを確認してください。



## 演習: IoHubとデバイスの接続

この演習では、Azure上にIoT  
デバイスとしてRaspberry Pi オンライン シミュレーター(Webアプリケーション)を使用し、作成したIoT

Hubを作成し、デバイスからのメッセージを受信します。

Hubに温度データと湿度データを送信します。

**NOTE**

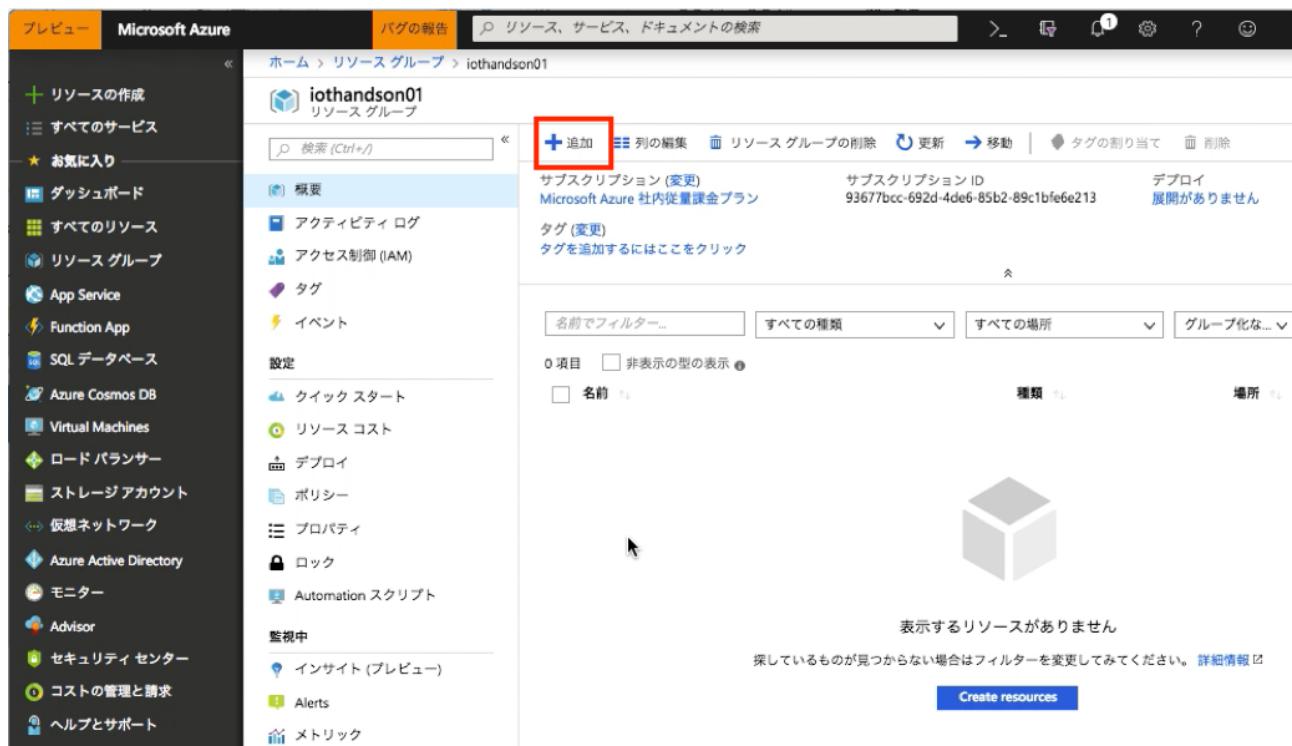
実際のPCやデバイスからデータを送信する場合、下記リンクに各開発言語/プラットフォーム毎(Node.js, .NET, Java, Python, iOS)の手順が掲載されています。

<https://docs.microsoft.com/ja-jp/azure/iot-hub/>

## IoT Hubの作成

このタスクでは、デバイスからのデータを受信するIoT Hubを作成します。

1. サイドメニューの[リソースグループ]をクリックし、リソースグループ一覧を表示します。
2. 「演習：Azure環境の準備」で作成したリソースグループの名前をクリックします。指定したリソースグループのリソース一覧画面が表示されます。（現在は空です）
3. [+追加]をクリックします。Everything ウィンドウが表示されます。



The screenshot shows the Microsoft Azure portal interface. The left sidebar contains a navigation menu with items like 'リソースの作成', 'すべてのサービス', 'お気に入り', 'ダッシュボード', etc. The main content area is titled 'iothandson01' and shows a list of resources under the 'Resource Groups' section. At the top of this list, there is a red box around the '+追加' (Add) button. Below the list, there are sections for 'サブスクリプション (変更)', 'タグ (変更)', and filtering options. A large 'Create resources' button is located at the bottom right of the main content area.

4. “IoT Hub”と入力し、Enter キーを入力します。検索結果が表示されます。[IoT
- Hub]をクリックします。



5. IoT Hub の詳細説明ウィンドウが表示されます。ウィンドウ下部の[作成]をクリックします。

**IoT Hub**  
Microsoft

Windows、Linux またはリアルタイムオペレーションシステムのいずれを実行している場合にも同時に膨大な接続デバイスをサポートします。次に、パフォーマンスを監視し、デジタル改革を加速するためにコマンドを送信します。

公開元 Microsoft

役に立つリンク 文書  
デバイス管理  
サービス概要  
価格とスケールの詳細  
Azure IoT Hubについての詳細

6. IoT Hubの作成ウィンドウが表示されます。各設定項目を入力後、ウィンドウ下部にある[Next: Size and scale]をクリックします。

Table 2. 設定項目と設定値

設定項目	設定値
サブスクリプション	ご用意いただいたAzureサブスクリプション（デフォルトのまま）
リソースグループ	既存のものを使用 演習1で作成したリソースグループ（デフォルトのまま）

設定項目	設定値
場所	東日本（デフォルトのまま）
名前	任意の名前 例) iothub-xx IoT Hubの名前はサービスURLとして使用されるため、Microsoft Azure全体で一意の名前を指定する必要があります

**IoT hub**  
Microsoft

[Basics](#) [Size and scale](#) [Review + create](#)

Create an IoT Hub to help you connect, monitor, and manage billions of your IoT assets. [Learn More](#)

**PROJECT DETAILS**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

\* Subscription [?](#) [\[Subscription dropdown\]](#)

\* Resource Group [?](#) [iothandson01](#) [\[New\]](#)

\* Region [?](#) [東日本](#)

\* IoT Hub Name [?](#) [iothub-201809](#) [\[Checkmark\]](#)

[Review + create](#) [Next: Size and scale »](#) [Automation options](#)

7. Size and scale ウィンドウで設定を確認して、ウィンドウ下部の[Review + create]をクリックします。

Table 3. 設定項目と設定値

設定項目	設定値
価格とスケールティア	S1 – Standard (デフォルトのまま, IoT Edge はStandardで利用可能)
IoT Hub ユニット	1 (デフォルトのまま)
Device-to-cloud /パーティション	4 (デフォルトのまま)

**IoT hub**  
Microsoft

Basics    Size and scale    Review + create

Each IoT Hub is provisioned with a certain number of units in a specific tier. The tier and number of units determine the maximum daily quota of messages that you can send. [Learn more](#)

SCALE TIER AND UNITS

\* Pricing and scale tier [S1: Standard tier](#) [Learn how to choose the right IoT Hub tier for your solution](#)

Number of S1 IoT Hub units [1](#) This determines your IoT Hub scale capability and can be changed as your need increases.

Pricing and scale tier <a href="#">S1</a>	Device-to-cloud-messages <a href="#">Enabled</a>
Messages per day <a href="#">400,000</a>	Message routing <a href="#">Enabled</a>
Cost per month <a href="#">2800.00 JPY</a>	Cloud-to-device commands <a href="#">Enabled</a>
	IoT Edge <a href="#">Enabled</a>
	Device management <a href="#">Enabled</a>

Advanced Settings

Device-to-cloud partitions [4](#)

[Review + create](#) [« Previous: Basics](#) [Automation options](#)

8. [Review + create] ウィンドウで[Create]ボタンをクリックしてIoT Hubを作成します。
9. デプロイが開始されると、通知ウィンドウに表示されます。

ホーム > Microsoft.IotHub-912155039 - 概要

## Microsoft.IotHub-912155039 - 概要

デプロイ

検索 (Ctrl+ /)

削除 キャンセル 再デプロイ 最新の情報に更新

... Your deployment is underway

Check the status of your deployment, manage resources, or troubleshoot deployment issues. Pin this page to your dashboard to easily find it next time.

Deployment name: Microsoft.IotHub-912155039  
Subscription: Microsoft Azure 社内従量課金プラン  
Resource group: iothandson01

DEPLOYMENT DETAILS (Download)  
Start time: 2018/9/12 15:50:43  
Duration: 1 秒  
Correlation ID: b84ce1b2-551f-4396-826d-a7788c5f7b52

リソース	種類	状態	操作の詳細
結果がありません。			

Additional Resources

- Windows Server 2016 VM Quickstart tutorial
- Cosmos DB Quickstart tutorial
- Web App Quickstart tutorial
- SQL Database Quickstart tutorial
- Storage Account Quickstart tutorial

Helpful Links

- Get started with Azure
- Azure architecture center

ホーム > Microsoft.IotHub-912155039 - 概要

## Microsoft.IotHub-912155039 - 概要

デプロイ

検索 (Ctrl+ /)

削除 キャンセル 再デプロイ 最新の情報に更新

✓ Your deployment is complete

Go to resource

Deployment name: Microsoft.IotHub-912155039  
Subscription: Microsoft Azure 社内従量課金プラン  
Resource group: iothandson01

DEPLOYMENT DETAILS (Download)  
Start time: 2018/9/12 15:54:46  
Duration: 4 分 4 秒  
Correlation ID: b84ce1b2-551f-4396-826d-a7788c5f7b52

リソース	種類	状態	操作の詳細
iothub-201809	Microsoft.Devices/Iot...	OK	操作の詳細

Additional Resources

- Windows Server 2016 VM Quickstart tutorial
- Cosmos DB Quickstart tutorial
- Web App Quickstart tutorial
- SQL Database Quickstart tutorial
- Storage Account Quickstart tutorial

Helpful Links

- Get started with Azure
- Azure architecture center

10. デプロイが完了すると、通知ウィンドウに完了メッセージが表示されます。[リソースに移動]をクリックし、設定したIoT Hubが作成されていることを確認します。

11. IoT Hubのメニューから[Endpoints]をクリックします

12. [エンドポイント]ウィンドウで組み込みエンドポイントの[Events]をクリックしてプロパティを開きます。

The screenshot shows the 'iothub-201809 - Endpoints' blade in the Azure portal. On the left, there's a navigation menu with links like Overview, Access logs, IAM, Tags, Events, Shared access policies, Pricing and scale, Operations monitoring, IP Filter, Certificates, Properties, Lock, and Automation scripts. Below that is an 'Explorers' section with a 'Query explorer' link.

The main area displays 'Built-in endpoints' and 'Custom endpoints'. Under 'Built-in endpoints', there's a table with one row for 'Events' (highlighted with a red box). Under 'Custom endpoints', there's a note about having up to 10 endpoints. Below these sections is a table header for 'NAME', 'ENDPOINT TYPE', and 'AZURE RESOURCE'.

On the right, the 'Properties' pane is open for 'iothub-201809'. It includes fields for 'Device-to-cloud settings' (Partitions set to 4), 'Event Hub-compatible name' (set to 'iothub-201809'), 'Event Hub-compatible endpoint' (set to 'Endpoint=sb://iothub-ns-iothub-201-762756-53733a38b8.s...'), 'Retention time' (set to 1 day), and 'Consumer groups' (\$Default).

13. [プロパティ] ウィンドウの[消費者グループ]に、既定の2つの消費者グループを追加します。(消費者グループの名前の例 `streamanalytics`)

`$Default` に加えて  
: `timeseriesinsights` ,  
Analytics, Time Series

消費者グループ(コンシューマーグループ)は、以降のステップでIoT Hubに送信されたデバイスデータを取得するサービス(Stream Insights)が使用するものです。

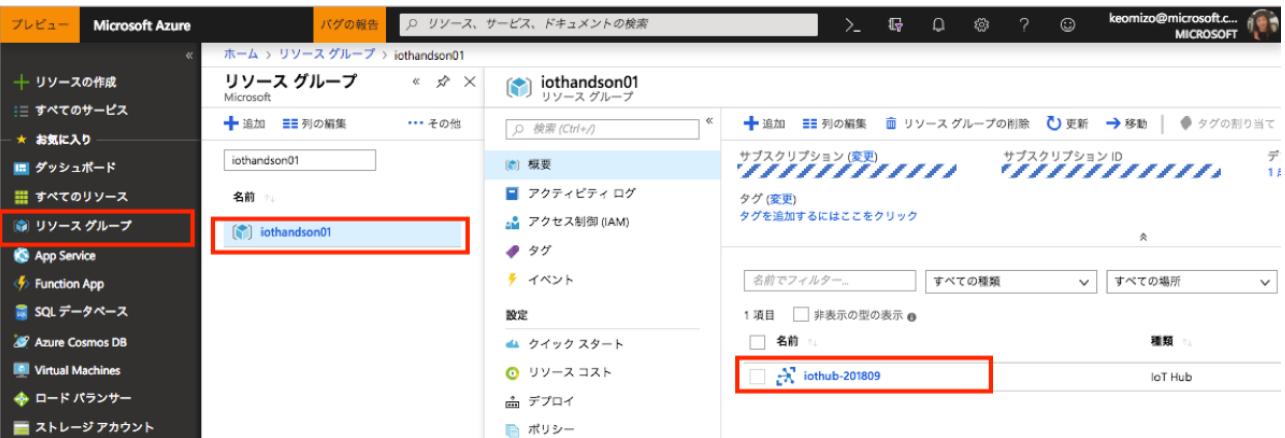
The screenshot shows the 'Properties' blade for 'iothub-201809'. The 'Save' button is highlighted with a red box. The 'Consumer groups' section shows '\$Default' and two additional entries: 'timeseriesinsights' and 'streamanalytics', both of which are highlighted with a red box.

# IoT Hubへのデバイス登録

このタスクでは、IoT Hubにデータを送信するためには、事前にIoT Hub上でデバイスを新規登録し、デバイスIDとデバイスキー（認証キー）を発行します。

Hubにデータを送信するデバイスを新規登録します。デバイスがIoT Hubにデバイスを登録しておく必要があります。IoT Hub上でデバイスを新規登録し、デバイスIDとデバイスキー（認証キー）を発行します。

1. タスク1で作成したIoT Hubの設定ウィンドウを表示します。



The screenshot shows the Microsoft Azure portal interface. The left sidebar has a red box around the 'リソースグループ' (Resource Group) item. The main content area shows the 'iotashboard01' resource group details. On the right, under the 'デバイス' (Devices) section, there is a list of devices. One device, 'iothub-201809', is highlighted with a red box.

2. IoT HubのメニューのExplorersのセクションから[IoT devices]をクリックします。

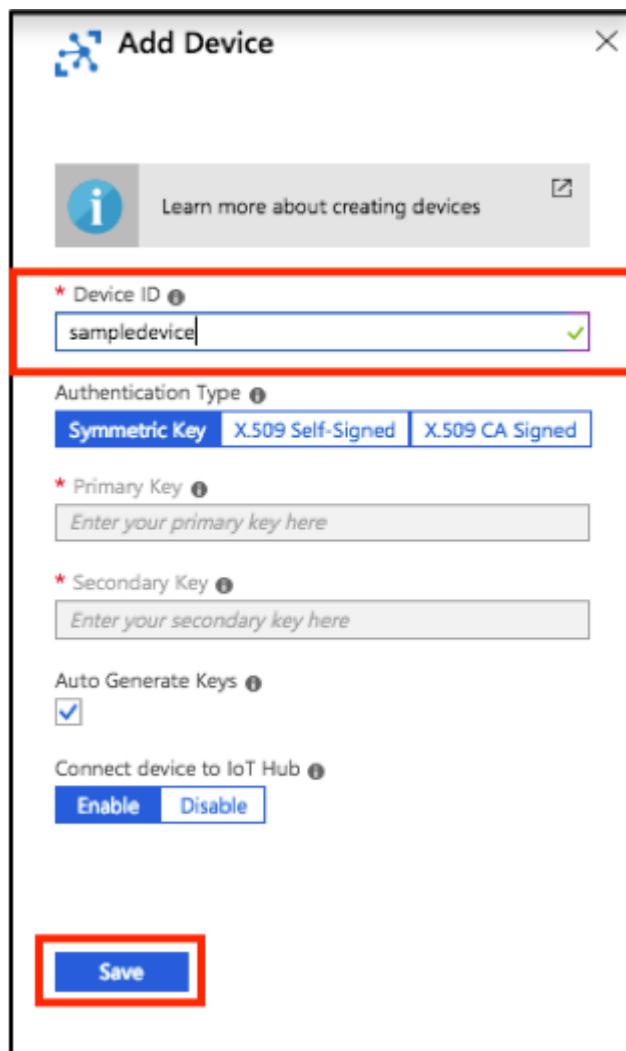
The screenshot shows the Azure IoT Hub Overview page for 'iothub-201809'. The left sidebar contains navigation links: Overview, Activity Log, IAM, Tags, Events, Settings (with sub-links for Shared access policies, Pricing and scale, Operations monitoring, IP Filter, Certificates, Properties, Lock, and Automation Scripts), Explorers (with Query explorer and IoT devices), and Automatic Device Management (with IoT Edge and IoT device configuration). The 'IoT devices' link in the Explorers section is highlighted with a red box.

3. ウィンドウ上部の[+Add]をクリックします。

The screenshot shows the 'IoT devices' management window for 'iothub-201809'. The top bar includes a search bar, a '+ Add' button (which is highlighted with a red box), a Refresh button, and a Delete button. The left sidebar is identical to the Overview page. The main area displays a message: 'You can use this tool to view, create, update, and delete devices on your IoT Hub.' Below this is a query editor with a 'Query' button, a SQL-like query 'SELECT \* FROM devices WHERE optional (e.g. tags.location='US')', and an 'Execute' button. A table at the bottom lists device information with columns: DEVICE ID, STATUS, LAST ACTIVITY, LAST STATUS UPDA..., AUTHENTICATION T..., CLOUD TO DEVICE ...'. The table shows the message '結果がありません' (No results found).

4. [デバイスの追加]ウィンドウにsampledeviceと入力し、[Save]をクリックし、デバイスを登録します

。ここで登録する文字列がデバイスIDとなります。



5. [デバイス] のデバイス一覧に、登録したデバイスが追加されていることを確認します。追加したデバイスをクリックすると、[デバイスの詳細] ウィンドウが表示されます。

The screenshot shows the 'IoT devices' list page for the 'iothub-201809' hub. At the top left is a blue gear icon with a white 'i'. To its right is the title 'iothub-201809 - IoT devices'. In the top right corner are three small icons: a gear, a magnifying glass, and a close button ('X'). Below the title is a search bar with placeholder text '(Ctrl+Shift+F)' and a 'Search' button. To the right of the search bar are three buttons: '+ Add' (highlighted with a blue dashed border), 'Refresh', and 'Delete'. On the left side is a sidebar with the following sections: 'Overview' (selected), 'アクティビティ ログ', 'アクセス制御 (IAM)', 'タグ', 'Events', '設定' (with sub-options: 'Shared access policies', 'Pricing and scale', 'Operations monitoring', 'IP Filter', 'Certificates', 'プロパティ', 'ロック'), and 'デバイス' (with sub-options: 'Create device', 'Delete device', 'Edit device', 'View device details'). The main content area contains an info bar with a blue 'i' icon and the text 'You can use this tool to view, create, update, and delete devices on your IoT Hub.' Below this is a 'Query' section with a text input field containing 'SELECT \* FROM devices WHERE optional (e.g. tags.location='US')' and a 'Execute' button. At the bottom is a table with the following columns: DEVICE ID, STATUS, LAST ACTIVITY, LAST STATUS UPDA..., AUTHENTICATION T..., CLOUD TO DEVICE ... . The table has one row with the data: 'sampledevice', 'Enabled', 'Sas', and '0'. This row is highlighted with a red box.

6. [デバイスの詳細] ウィンドウで、デバイスキーや接続文字列の確認や、デバイスツインやデバイスへのメッセージ送信、ダイレクトメソッドが利用できます。ここでは"Connection string (primary key)"の値をコピーしておきます。次のタスクで、コピーした接続文字列をデバイスのプログラムコー

ド内に設定します。

The screenshot shows the 'Device Details' page for a device named 'sampledevice'. At the top, there are buttons for Save, Regenerate keys, Device Twin, Message To Device, Direct Method, Add Module Identity, and Refresh. Below these are fields for Device Id (sampledevice), Primary key, Secondary key, and Connection string (primary key). The Connection string (primary key) field is highlighted with a red border. There is also a Connection string (secondary key) field. Underneath, there is a section for connecting the device to IoT Hub with 'Enable' and 'Disable' buttons, and a 'Module Identities' section with a table showing no listed identities.

## IoT Hubデバイスクライアントの開発

このタスクでは、Raspberry Piオンライン シミュレーターを使ってIoT Hubにデータを送信します。本エミュレータはRaspberry PiをシミュレートしたWebアプリケーションで、Node.jsでプログラムコードを記述することでIoT Hubにデータを送信することができます。

### 1. Raspberry Pi オンライン シミュレーターの設定

<https://docs.microsoft.com/ja-jp/azure/iot-hub/iot-hub-raspberry-pi-web-simulator-get-started>

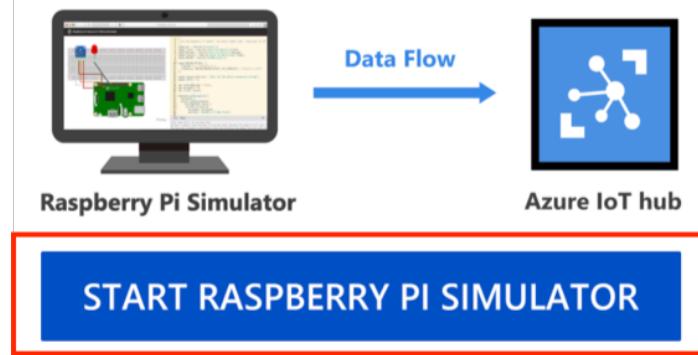
# Raspberry Pi オンライン シミュレーター の Azure IoT Hub への接続 (Node.js)

2018/04/11・共同作成者

Raspberry Pi Web シミュレーター

このチュートリアルでは、まず Raspberry Pi オンライン シミュレーターの操作の基礎について説明します。次に、[Azure IoT Hub](#) を使って、シミュレーターをクラウドにシームレスに接続する方法について説明します。

物理デバイスがある場合、「[Raspberry Pi の Azure IoT Hub への接続](#)」に移動して作業を開始してください。



- a. 上記URLをブラウザで開き、[START RAPBERRY PI SIMULATOR]をクリックします。
- b. [Step1 Raspberry Pi Web シミュレーターの概要]ウィンドウが開きます。[次のステップ]をクリックします。
- c. [Step2 IoT Hubの作成]ウィンドウが表示されます。[次のステップ]をクリックします。
- d. [Step3 Pi Web シミュレーターでのサンプルアプリケーションの実行]ウィンドウが表示されます。[私は理解して]をクリックします。
- e. [Raspberry Pi Azure IoT Online Simulator]ウィンドウが開きます。
- f. 15行目のconnectionStringの'[Your IoT hub device connection string]'を、作成したIoT Hubの接続文字列に変更します。記載方法は[]までを書き換えます。'コピーした接続文字列'となります。

変更前

```
const connectionString = '[Your IoT hub device connection string]';
```

変更後

```
const connectionString = 'HostName=iothub0910-001.azure-devices.net;DeviceId=sampledevice;SharedAccessKey=xxxxxxxxxxxxxxxxxxxxxx';
```

- g. 23行目～37行目にあるfunction  
getMessage(cb)を下記に書き換えます。シミュレーションデバイスが生成したデータ(機械: 温度/圧力, 大気: 温度/湿度)をJSONで返します。

```

function getMessage(cb) {
    messageId++;
    sensor.readSensorData()
        .then(function (data) {
            cb(JSON.stringify({
                messageId: messageId,
                deviceId: 'Raspberry Pi Web Client',
                machine : { temperature : data.temperature_C, pressure : 10 },
                ambient : { temperature : data.temperature_C, humidity : data.humidity }
            }), data.temperature_C > 30);
        })
        .catch(function (err) {
            console.error('Failed to read out sensor data: ' + err);
        });
}

```

- h. 画面下部の[Run]ボタンをクリックします。画面左のLEDが点滅していれば動作しています。  
またコンソールウィンドウに送信したテレメトリデータが表示されることを確認します。

エラーが発生せず、下記のように表示されていれば成功です。Webアプリケーションを閉じるか[Stop]ボタンをクリックするまでテレメトリデータは送信され続けます。

```

Sending message: {"messageId":1,"deviceId":"Raspberry Pi Web
Client","machine": {"temperature":23.61089154634202,"pressure":10}, "ambient": {"te
mperature":23.61089154634202, "humidity":73.80948916321065}}
>
Message sent to Azure IoT Hub
>

```

## IoT Hubでのデータ受信確認

このタスクでは、作成したIoT Hubの[概要]をクリックし、使用状況を表示します。メッセージがカウントアップされていればデバイスからのテレメトリデータが受信できています。

The screenshot shows the Azure IoT Hub Overview page for the resource group 'iothub-201809'. The 'Overview' tab is selected and highlighted with a red box. The main content area displays usage statistics for the IoT Hub, including the number of messages sent daily, the count of IoT devices, and the status of IoT Edge components. To the right, there is a chart titled 'Device twin operations' showing a single data point at 17:00 with a value of 1. A legend at the bottom indicates 'SUCCESSFUL TWIN READS FROM BACK END (カウント)'.

## 演習: Time Series Insightsによるデータの可視化

この演習では、IoT Hubで受信したデータをTime Series Insightsで表示します。

Time Series Insightsの詳細は下記URLに記載があります。

### NOTE

<https://docs.microsoft.com/ja-jp/azure/time-series-insights/time-series-insights-overview>

Time Series Insights は時系列データを念頭に設計されたデータベースです。ほぼリアルタイムでのデータ探索が可能です。また Time Series Insights REST API を使用して新しいカスタムソリューションを作成することもできます。既定のリテンション期間は31日間となっており、最大400日のデータリテンション期間を構成できます。

## Time Series Insightsの作成

このタスクでは、Azure上にTime Series Insightsを作成します。

- 「演習:Azure環境の準備」で作成したリソースグループを表示し、ウィンドウ上部の[+追加]をクリックします。

The screenshot shows the Microsoft Azure portal interface. On the left, a sidebar lists various service categories like 'リソースの作成', 'すべてのサービス', and 'リソース グループ'. The 'リソース グループ' item is highlighted with a red box. The main content area shows the 'iothandson01' resource group details. A red box highlights the '名前' field containing 'iothandson01'. The '概要' section includes links for 'アクティビティ ログ', 'アクセス制御 (IAM)', 'タグ', 'イベント', '設定', 'クイック スタート', 'リソース コスト', 'デプロイ', and 'ポリシー'. On the right, there's a 'サブスクリプション' section with a table showing 'サブスクリプション ID' (iohub-201809) and 'サブスクリプション名' (IoT Hub). A red box highlights the '追加' button at the top of the table.

2. Everythingウィンドウの検索ボックスで、”Time Series Insights”と入力しEnterキーを押します。
3. [Time Series Insights]をクリックします。

The screenshot shows the 'Everything' search results window. The search bar at the top contains 'Time Series Insights'. Below the search bar, there's a '結果' (Results) section. A red box highlights the first search result, which is a 'Time Series Insights' entry. This entry includes a small icon, the name 'Time Series Insights', the provider 'Microsoft', and the category 'Analytics'.

名前	公開元	カテゴリ
Time Series Insights	Microsoft	Analytics

4. 表示された詳細ウィンドウ下部の[作成]をクリックします。

## Time Series Insights

Microsoft



Azure Time Series Insights is a fully managed analytics, storage, and visualization service that makes it incredibly simple to explore and analyze billions of IoT events simultaneously. Time Series Insights gives you a global view of your data, letting you quickly validate your IoT solution and avoid costly downtime to mission-critical devices by helping you discover hidden trends, spot anomalies, and conduct root-cause analyses in near real-time.

- Find actionable insights in seconds
- Start in seconds, scale in minutes
- Create a global view of your IoT-scale data
- Leverage Time Series Insights in your Apps and Solutions

♡ 後で使用するために保存



公開元

Microsoft

役に立つリンク

[Documentation](#)  
[Service overview](#)  
[Pricing details](#)

作成

5. Time Series Insights environmentウィンドウの各設定値を入力後、ウィンドウ下部にある[作成]をクリックします。

Table 4. 設定項目と設定値

設定項目	設定値
環境名	任意の名前 例) timeseriesinsights001
サブスクリプション	ご用意いただいたAzureサブスクリプション（デフォルトのまま）

設定項目	設定値
リソースグループ	「演習：Azure環境の準備」で作成したリソースグループを選択
場所	米国東部（デフォルトのまま）
SKU	S1（デフォルトのまま）
Capacity	1（デフォルトのまま）



6. デプロイが開始されると、通知ウィンドウに表示されます。
7. デプロイが完了すると、通知ウィンドウに完了メッセージが表示されます。[リソースに移動]をクリックし、設定したTime Series Insightsが作成されていることを確認します。



# Time Series Insightsの設定

このタスクでは、作成したTime Hubをイベントソースに設定します。

Series

Insightsに、演習 2 で作成したIoT

1. タスク 1 で作成したTime Series Insightsの設定ウィンドウを表示します。
2. Time Series Insightsの設定メニューにある[イベントソース]をクリックします。ウィンドウ上部の [+追加]をクリックします。



3. [イベントソース]ウィンドウの各設定値を入力後、ウィンドウ下部にある[作成]をクリックします。

Table 5. 設定項目と設定値

設定項目	設定値
イベントソース名 Event source name	任意の名前 例) eventsource001
ソース Source	IoT Hub
インポートオプション Import option	使用可能なサブスクリプションから... (デフォルトのまま) Use IoT Hub from available subscription
サブスクリプションID Subscription Id	ご用意いただいたサブスクリプション (デフォルトのまま )
IoT Hub名 IoT hub name	「演習:IoT Hubのデバイス接続」で作成したIoT Hub名 ex) iothub-xx

設定項目	設定値
IoT Hubポリシー名 IoT hub policy name	自動入力のまま (iothubowner)
IoT Hubポリシーキー Iot hub policy key	自動入力のまま (**)
IoT Hub コンシューマグループ Iot hub consumer group	timeseriesinsights (IoT Hubで追加したコンシューマグループ)
Event serialization format	自動入力のまま (JSON)
Timestamp property name	空白のまま

新しいイベントソース

\* イベントソース名  
eventsource001

\* ソース  
IoT Hub

\* インポートオプション  
使用可能なサブスクリプションから IoT ...

\* サブスクリプション  
[リスト]

\* IoT Hub名  
iothub-201809

\* IoT Hubポリシー名  
iothubowner

\* IoT Hubポリシーキー  
\*\*\*\*\*

**⚠ 特定のコンシューマー グループが同時に持つことのできるアクティブなリーダーは1つだけであるため、このコンシューマー グループはこのイベントソース専用に使用する必要があります。**

\* IoT Hub コンシューマーグループ  
\$Default

イベントシリアル化形式  
JSON

タイムスタンプのプロパティ名

**作成**

4. [イベントソース] ウィンドウに追加したイベントソースが表示されていることを確認します。確認後、右上の[X]ボタンをクリックして[イベントソース] ウィンドウを閉じます。Time Series Insights概要ウィンドウの”イベントソース”が ”1” となります。

The screenshot shows the Azure Time Series Insights environment overview page for 'timeseriesinsights001'. The left sidebar has a 'Event Sources' section with a red box around the number '1'. The main area shows basic information like resource group 'iothandson01', status 'Succeeded', and SKU 'S1'. On the right, there's a 'Data Access FQDN' section with a URL: <https://insights.timeseries.azure.com/?environment=timeseriesinsights001>.

5. 以下のようなデータアクセスポリシーの警告が表示される場合、データアクセスポリシーを追加します。

The screenshot shows the Azure Time Series Insights environment overview page. A yellow warning bar at the top says: '⚠ データ アクセス ポリシーを追加してください。データ アクセス ポリシーが定義されていない環境のデータにはアクセスできません。' (Please add a data access policy. Data access policies are required for environments where no data access policies are defined.) Below the bar, the 'Event Sources' section shows a count of '1' with a red box around it. The rest of the page displays basic environment details.

Tim Series Insightsのメニューから[データアクセスポリシー]をクリックします。

timeseriesinsights001 - ディレクトリ  
Time Series Insights 環境

検索 (Ctrl+ /) <<

- 概要
- アクティビティ ログ
- アクセス制御 (IAM)
- タグ

設定

- ロック
- Automation スクリプト
- 構成

環境トポロジ

- データ アクセス ポリシー
- イベント ソース
- 参照データ セット

- a. データアクセスポリシーウィンドウで、ユーザーが一件も登録されていないことを確認して[追加]をクリックします。

timeseriesinsights001 - データ アクセス ポリシー  
Time Series Insights 環境

検索 (Ctrl+ /) <<

+ 追加

概要

名前

- b. [ユーザー役割の選択]ウィンドウで[ユーザーの選択]をクリックして、Azure管理ポータルにログインしたご自身のアカウント(Azure管理ポータルの右上に表示)を検索して追加します。

ユーザー役割の選択 X ユーザーの選択

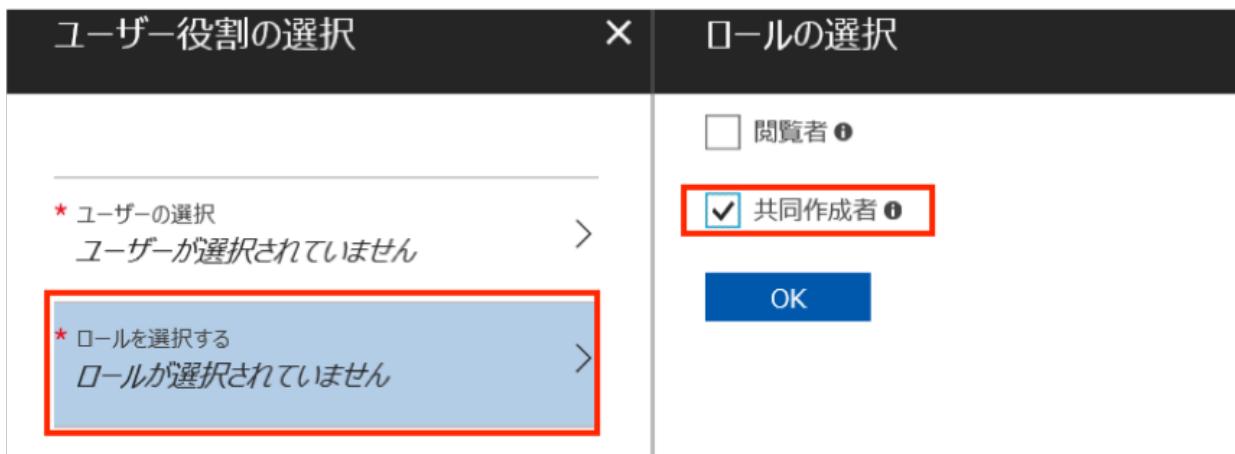
\* ユーザーの選択  
ユーザーが選択されていません >

\* ロールを選択する  
ロールが選択されていません >

+ 招待する

選択 ⓘ  
名前または電子メール アドレスで検索

- c. [ユーザー役割の選択]ウィンドウで[ロールを選択する]をクリックして、[共同作成者]を選択して[OK]をクリックします。



- d. ユーザーとロールの選択後、[OK]をクリックします。
- データ保持期間やキャパシティを設定する場合には、[構成]メニューをクリックします。

Data retention time (in days) やCapacityを必要に応じて設定します。

## Time Series Insightsでのデータ表示とカスタマイズ

このタスクでは、作成したTime Series Insightsエクスプローラーへアクセスし、IoT Hubで受信した温度データと湿度データを表示します。

- 作成したTime Series Insightsを表示します。
- [概要]ウィンドウ上部の[環境を開く]をクリックします。[Time Series Insights エクスプローラーのURL]がURLになります

- Time Series Insightsのポータル画面が表示されます。

4. 温度データを表示するために、サイドメニューのMEASUREプルダウンを”Events”から”ambient.temperature”に変更します。

The screenshot shows the AWS Time Series Insights interface. At the top, it says "Time Series Insights" and "timeseriesinsights001 Last 60 Mins, UTC". Below that, it shows "Interval size: 1 Minute". There are "Add" and "Refresh" buttons. A modal window is open, titled "ambient.temperature". It has a "WHERE" section with a placeholder "Add a search predicate...". Below that is a "MEASURE" section with four options: COUNT, AVG, SUM, and ambient.temperature (which is highlighted with a red box). There is also a gear icon. The "SPLIT BY" section shows "(None)".

5. 湿度データを追加するために、サイドメニューの[Add]をクリックし、MEASUREプルダウンを”Events”から”ambient.humidity”に変更します。

Time Series Insights

timeseriesinsights001

Last 60 Mins, UTC

Interval size: 1 Minute

Add Refresh

ambient.te... Only Hide X

WHERE Add a search predicate...

MEASURE COUNT AVG SUM

ambient.temperature

SPLIT BY (None)

ambient.h... Only Hide X

WHERE Add a search predicate...

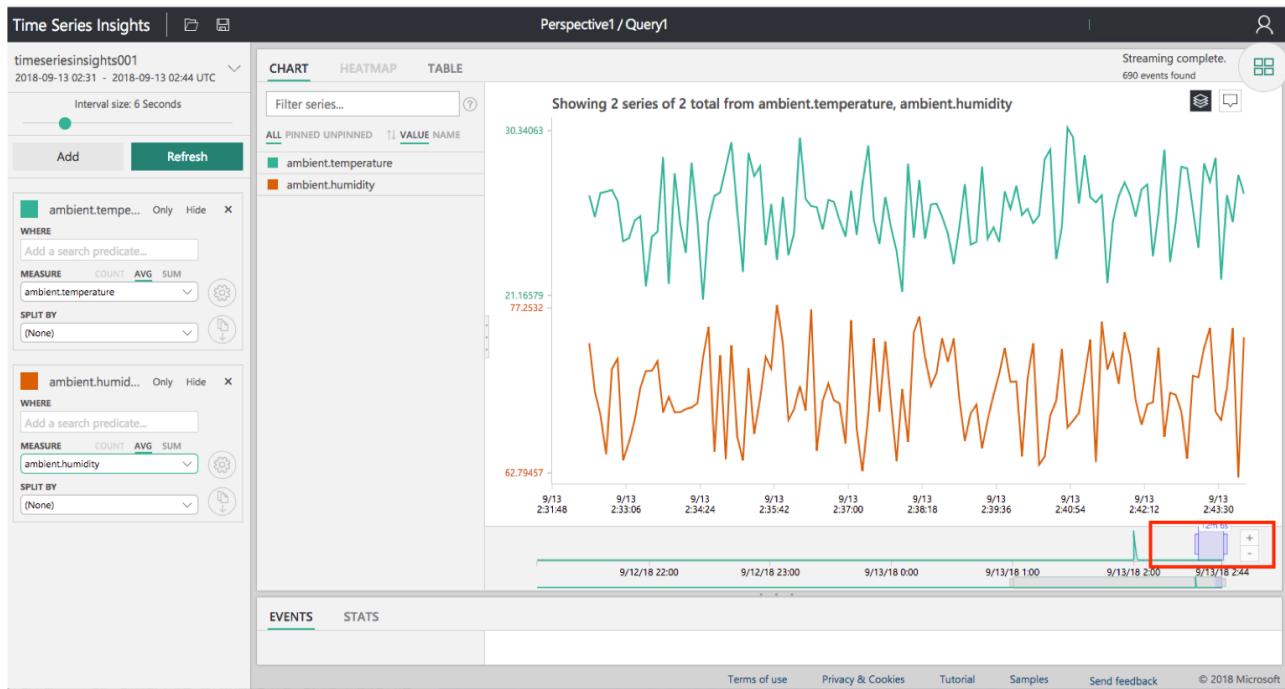
MEASURE COUNT AVG SUM

ambient.humidity

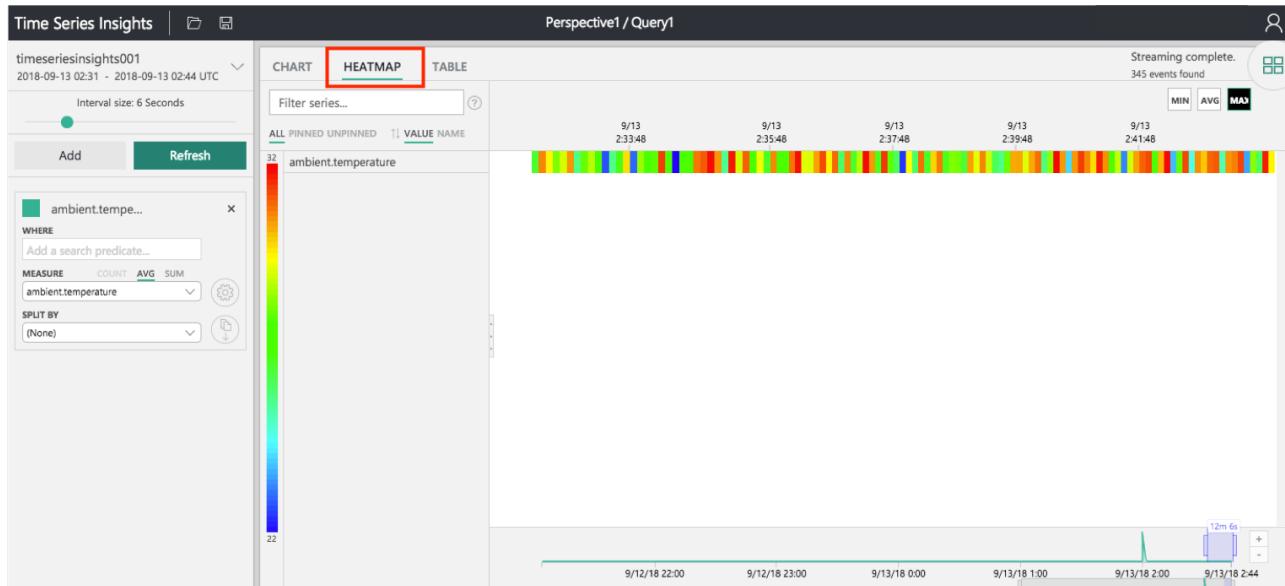
SPLIT BY (None)

The screenshot shows the AWS Time Series Insights interface. At the top, it displays 'Time Series Insights' and the dataset 'timeseriesinsights001' for the last 60 minutes in UTC. The interval size is set to 1 minute. Below this, there are two main configuration sections. The first section, 'ambient.te...', has its MEASURE dropdown set to 'AVG' and contains the metric 'ambient.temperature'. The second section, 'ambient.h...', has its MEASURE dropdown set to 'SUM' and contains the metric 'ambient.humidity'. Both sections have their SPLIT BY dropdown set to '(None)'. A red box highlights the 'Add' button at the top left, and another red box highlights the 'ambient.h...' configuration.

- より詳細なデータを確認するために、表示期間を調整します。紫色の両端をドラッグアンドドロップし、範囲を調整します。ポータル左上のメニューからも範囲設定ができます。



7. 左サイドメニューの[Interval size]を設定すると、グラフの描画粒度を調整することができます。Interval sizeごと平均の値がグラフに描画されています。
8. 表示するプロパティが1つの場合、ヒートマップを表示することができます。”ambient.humidity”のグラフ表示設定を削除し、[HEATMAP]タブをクリックします。ヒートマップは多くのデバイスを表示する際のデータ傾向把握に有効です。



## 演習: Stream Analyticsによるリアルタイムデータ加工

この演習では、Stream

Storageへのデータ格納を行います。様々なリアルタイムデータ処理を実現することができます。ここでは、個別データと1分単位のサマリデータを出力するように設定します。

Analyticsにより、リアルタイム分析とBlob

## NOTE

Stream Analyticsの詳細は、下記URLに記載があります。

<https://docs.microsoft.com/ja-jp/azure/stream-analytics/>

# Stream Analyticsの作成

このタスクでは、Azure上にStream Analyticsを作成します。

- 「演習：Azure環境の準備」で作成したリソースグループを表示し、ウィンドウ上部の[+追加]をクリックします。



- Everythingウィンドウの検索ボックスで、"Stream Analytics"と入力しEnterキーを押します。

A screenshot of the Azure 'Everything' search results. The search bar at the top has 'stream analytics' typed in. Below the search bar, there is a '結果' (Results) section. It shows two items in a table format:

名前	公開元
Stream Analytics job	Microsoft
Azure Stream Analytics on IoT Edge	Microsoft

The first item, 'Stream Analytics job', is highlighted with a red box.

- [Stream Analytics job]をクリックし、表示された詳細ウィンドウ下部の[作成]をクリックします。

**Stream Analytics job**

Azure Stream Analytics

Azure Stream Analytics is a fully managed, cost effective real-time event processing engine that helps to unlock deep insights from data. Stream Analytics makes it easy to set up real-time analytic computations on data streaming from devices, sensors, web sites, social media, applications, infrastructure systems, and more.

With a few clicks in the Azure portal, you can author a Stream Analytics job specifying the input source of the streaming data, the output sink for the results of your job, and a data transformation expressed in a SQL-like language. You can monitor and adjust the scale/speed of your job in the Azure portal to scale from a few kilobytes to a gigabyte or more of events processed per second.

Stream Analytics leverages years of Microsoft Research work in developing highly tuned streaming engines for time-sensitive processing, as well as language integrations for intuitive specifications of such.

[後で使用するための保存](#)

```

graph LR
    IoTHub[IoT Hub] --> SA[Stream Analytics]
    SA --> Dashboard[Dashboard]
    SA --> Alerts[Alerts]
    SA --> Storage[Storage]
  
```

公開元 Microsoft

[What is Azure Stream Analytics?](#)

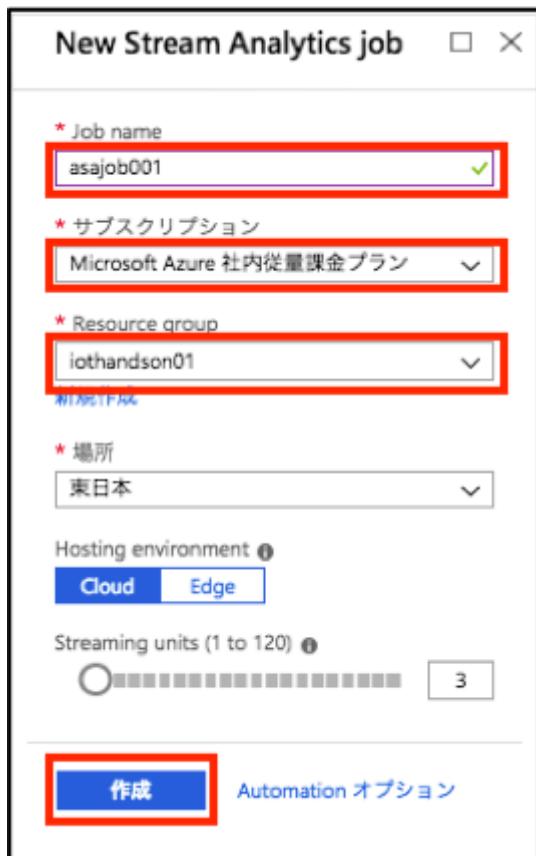
[作成](#)

4. [新しい Stream Analytics ジョブ] ウィンドウの各設定値を入力後、ウィンドウ下部にある[作成]をクリックします。

Table 6. 設定項目と設定値

設定項目	設定値
ジョブ名 Job name	任意の名前 例) asajob001
サブスクリプション	ご用意いただいたAzureサブスクリプション（デフォルトのまま）
リソースグループ Resource group	既存のものを使用 「演習:Azure環境の準備」で作成したリソースグループ
場所	東日本（デフォルトのまま）

設定項目	設定値
ホスティング環境 Hosting environment	Cloud (デフォルトのまま)
ストリーミングユニット Streaming units	3 (デフォルトのまま)



5. デプロイが完了すると、通知ウィンドウに完了メッセージが表示されます。[リソースに移動]をクリックし、設定したStream Analyticsが作成されていることを確認します。

## Stream Analyticsの入力設定

このタスクでは、作成したStream Analyticsに[入力](Stream Analyticsへの入力データ)を設定します。

1. 「タスク：Stream Analyticsの作成」で作成したStream Analyticsの設定ウィンドウを表示します。

[サイドメニュー] > [リソースグループ] > [作成したリソースグループ] > “作成したStream Analyticsの名前”

iothandson01  
リソース グループ

検索 (Ctrl+ /)

**概要**

- アクティビティ ログ
- アクセス制御 (IAM)
- タグ
- イベント

設定

- クイック スタート
- リソース コスト
- デプロイ
- ポリシー
- プロパティ

サブスクリプション (変更)

タグ (変更)  
タグを追加するにはここをクリック

サブスクリプション ID

名前でフィルター... すべての種類

4 項目  非表示の型の表示

名前
<input checked="" type="checkbox"/> asajob001
<input type="checkbox"/> iothub-201809
<input type="checkbox"/> timeseriesinsights001
<input type="checkbox"/> eventsource001 (timeseriesinsights001/eventsource001)

2. Stream Analyticsの設定メニューにある[Inputs]をクリックし、ウィンドウ上部の[+Add stream input]-[IoT Hub]をクリックします。

asajob001 - Inputs  
Stream Analytics job

検索 (Ctrl+ /)

**Overview**

- アクティビティ ログ
- アクセス制御 (IAM)
- タグ
- 問題の診断と解決

設定

- ロック

Job topology

**Inputs**

Add stream input Add reference input

Event Hub

**IoT Hub**

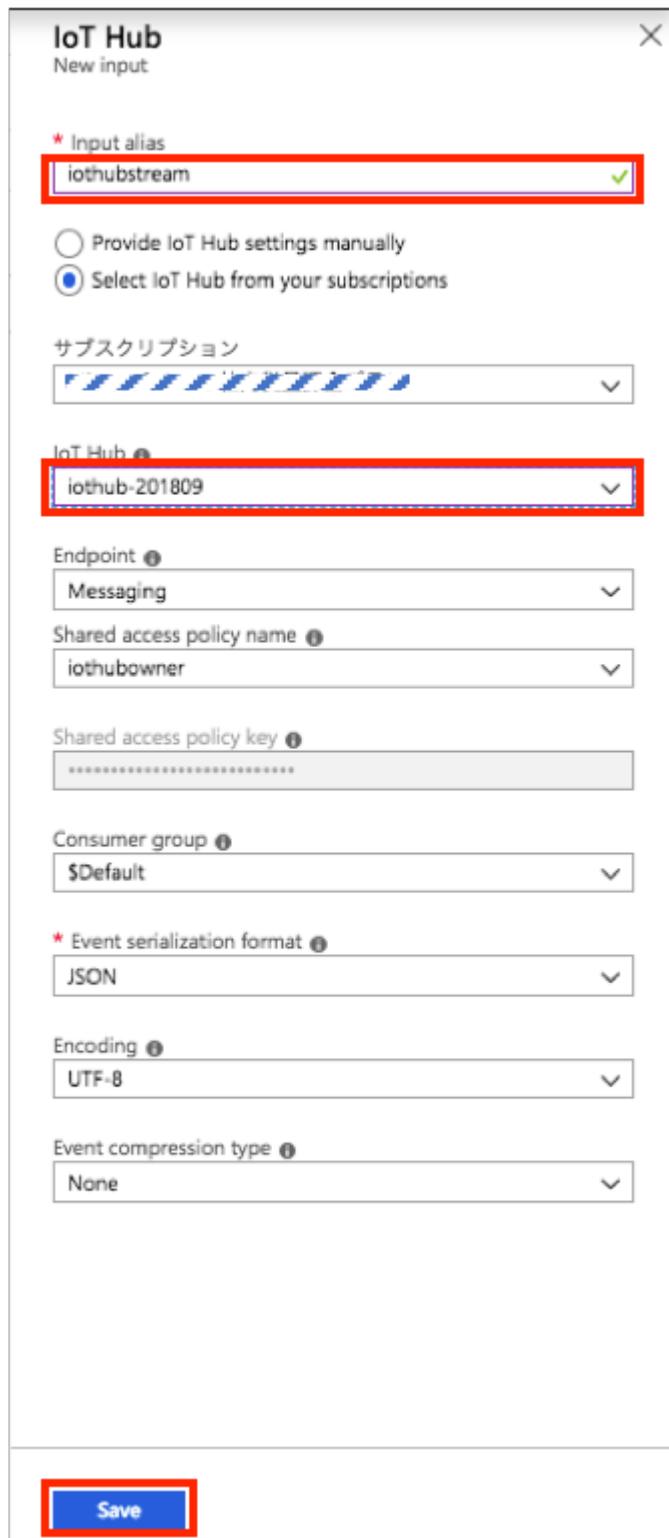
Blob storage

3. [新規入力]ウィンドウの各設定値を入力後、ウィンドウ下部にある[作成]をクリックします。

Table 7. 設定項目と設定値

設定項目	設定値
入力のエイリアス Input alias	任意の名前 指定した名前は後述のクエリの構文内で参照 例)IoTHubStream

設定項目	設定値
インポートオプション ○Provide IoT Hub settings manually ●Select IoT Hub from your subscriptions	Select IoT Hub from your subscriptions (デフォルトのまま)
サブスクリプション	ご用意いただいたAzureサブスクリプション（デフォルトのまま）
IoT Hub	「演習：IoT Hubとデバイスの接続」で作成したIoT Hubを選択
エンドポイント Endpoints	Messaging
共有アクセスポリシー名 Shared access policy name	iothubowner (デフォルトのまま)
共有アクセスポリシーキー Shared access policy key	デフォルトのまま
コンシューマーグループ Consumer group	「演習：IoT Hubとデバイス接続」で追加したStream Analytics用のコンシューマーグループ streamanalytics
イベントシリアル化形式 Event serialization format	JSON (デフォルトのまま)
エンコード+ Encoding	UTF-8 (デフォルトのまま)
イベントの圧縮タイプ Event compression type	None



4. 通知ウィンドウに「入力のテスト中」と表示されます。「接続テストが成功しました」と表示されれば、入力の追加完了です。

## Stream Analyticsの出力設定（ストレージへの出力）

このタスクでは、作成したStream Analyticsに出力を設定します。デバイスから送信された全データを Azure Blobストレージに保存します。

1. 「演習：Azure環境の準備」作成したリソースグループを表示し、ウィンドウ上部の[+追加]をクリックします。



2. Everythingウィンドウの検索ボックスで、”ストレージ”と入力しEnterキーを押します。[ストレージアカウント]をクリックします。

名前	公開元	カテゴリ
ストレージ アカウント - Blob, File, Table, Queue	Microsoft	Storage
SQL エラスティック データベース プール	Microsoft	Databases

3. 表示された詳細ウィンドウ下部の[作成]をクリックします。

## ストレージ アカウント - Blob、File、Table、Queue

Microsoft



Microsoft Azure では拡張性と耐障害性に優れたクラウドストレージ、バックアップ、および回復のソリューションを提供しており、データの規模に関わらず、あらゆるデータに対応します。既存インフラストラクチャと連携させることができるために、従来から利用してきたアプリケーションとビジネス継続性の戦略を低成本で強化できます。さらにクラウドアプリケーションに必要なストレージを提供し、構造化されていないテキストデータやバイナリデータ(ビデオ、オーディオ、画像など)にも対応します。

後で使用するため保存

公開元

Microsoft

後に立つリンク

[ドキュメント](#)  
[サービスの概要](#)  
[価格](#)

作成

- 下記のとおり、各設定値を入力後、ウィンドウ下部にある[作成]をクリックします。

Table 8. 設定項目と設定値

設定項目	設定値
名前	任意の名前。 Microsoft Azure全体で一意の名前 例) iothandsonstorage
デプロイモデル	Resource Manager (デフォルトのまま)
アカウントの種類	BLOBストレージ
場所	東日本
レプリケーション	ローカル冗長ストレージ (デフォルトのまま)
パフォーマンス	Standard (デフォルトのまま)
アクセス層	ホット (デフォルトのまま)

設定項目	設定値
安全な転送が必須	無効
サブスクリプション	ご準備いただいたサブスクリプション
リソースグループ	既存のものを使用 「演習:Azure環境の準備」で作成したリソースグループ
仮想ネットワーク	無効
Data Lake Storage Gen2 (プレビュー)	何もしない

ストレージ アカウントの... □ ×

新規作成エクスペリエンスのプレビュー →  
一を行います

ストレージ アカウントのコストは、使用量と、下で選ぶオプションに応じて決まります。

[詳細情報](#)

\* 名前 [●](#)  
iothandsonstorage201809

.core.windows.net

デプロイ モデル [●](#)  
[Resource Manager](#) [クラシック](#)

アカウントの種類 [●](#)  
**BLOB ストレージ**

\* 場所  
東日本

レプリケーション [●](#)  
ローカル冗長ストレージ (LRS)

パフォーマンス [●](#)  
[Standard](#) [Premium](#)

アクセス層 (既定) [●](#)  
[クール](#) [ホット](#)

\* 安全な転送が必須 [●](#)  
[無効](#) [有効](#)

\* サブスクリプション

\* リソース グループ  
iothandson01

仮想ネットワーク  
仮想ネットワークを構成します [●](#)  
[無効](#) [有効](#)

Data Lake Storage Gen2 (プレビュー)  
階層構造の名前空間 [●](#)

**作成** [Automation オプション](#)

5. リソースグループ内に作成したストレージアカウントが追加されていることを確認します。

The screenshot shows the Azure portal's Resource Groups interface. On the left, there's a sidebar with navigation links like Overview, Logs, IAM, Tags, Events, Settings, Quick Start, Cost Management, Deployment, Policies, Properties, and Locks. The main area is titled 'iothandson01' and shows a list of resources under 'Subscriptions (Change)'. A specific resource, 'iothandsonstorage201809', is highlighted with a red box. The details pane on the right shows it's a Stream Analytics job.

6. Stream Analyticsの設定メニューにある[出力]をクリックし、ウィンドウ上部の[+Add]-[Blob storage]をクリックします。

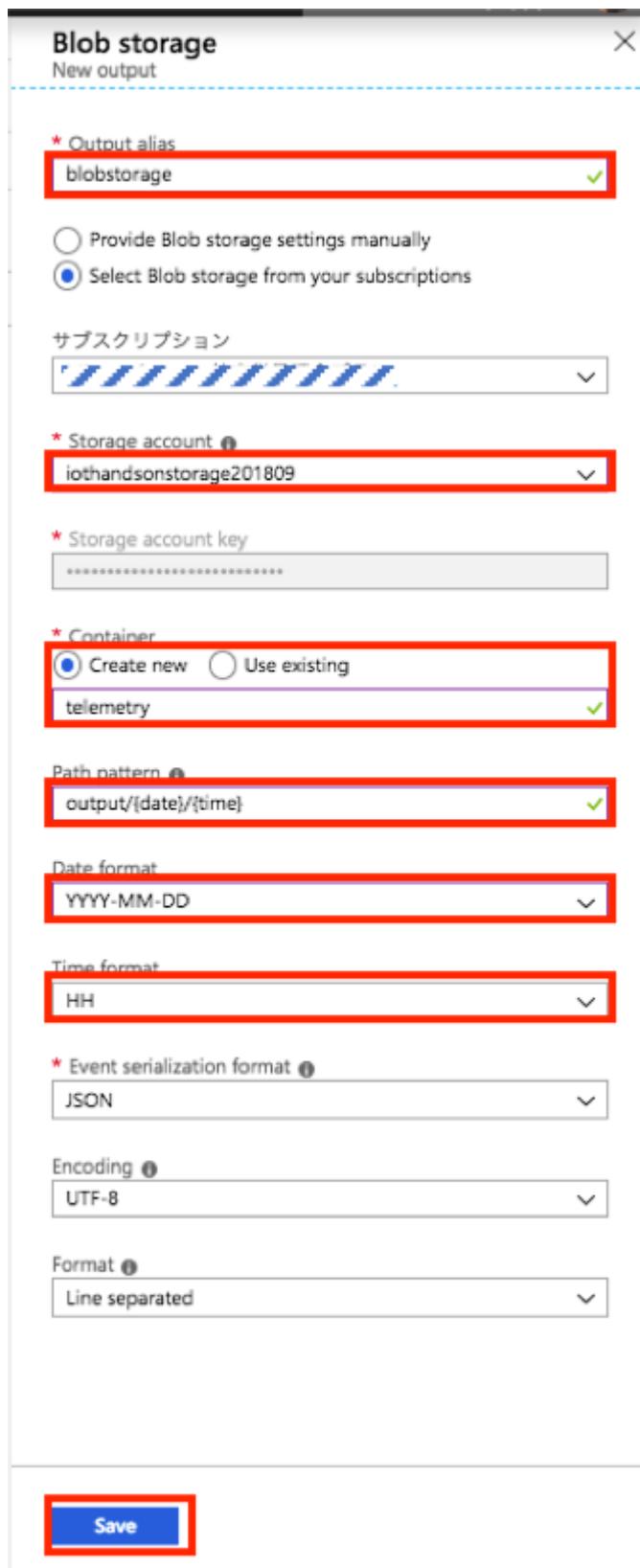
The screenshot shows the Stream Analytics job 'asajob001 - Outputs' settings. The 'Outputs' tab is selected. A modal window is open, showing various output options: Event Hub, SQL Database, Blob storage, Table storage, Service Bus topic, Service Bus queue, Cosmos DB, Power BI, and Data Lake Storage Gen1. The 'Blob storage' option is highlighted with a red box, and the 'Add' button at the top of the modal is also highlighted with a red box.

7. [新規出力]ウィンドウの各設定値を入力後、ウィンドウ下部にある[作成]をクリックします。

Table 9. 設定項目と設定値

設定項目	設定値
出力エイリアス Output alias	任意の名前。 指定した名前は後述のクエリの構文内で参照 例) blobstorage

設定項目	設定値
インポートオプション ○Provide IoT Hub settings manually ●Select IoT Hub from your subscriptions	Select IoT Hub from your subscriptions (デフォルトのまま)
インポートオプション	サブスクリプションからBlob Storageを選択する
サブスクリプション	ご準備いただいたサブスクリプション
ストレージアカウント Storage account	当タスクで作成したストレージアカウントを選択する
ストレージアカウントキー Storage accountkey	自動入力
コンテナ指定方式 ●Create new ○Use existing	Create new (デフォルトのまま)
コンテナ名 Container	任意の名前 例) telemetry
パスパターン	出力パスを設定する 例) output/{date}/{time}
日付の形式 Date format	形式を選択する。ここでは YYYY-MM-DD を選択
時刻の形式 Time format	形式を選択する
イベントシリアル化形式 Event serialization format	JSON
エンコード Encoding	UTF-8asa-querystorage-query.png
フォーマット Format	Line separated



## Stream Analyticsのクエリ設定（ストレージへの出力）

このタスクでは、Stream Analyticsにストレージ出力するクエリを設定します。

1. Stream Analyticsの設定メニューにある[クエリ]をクリックします。

The screenshot shows the Azure Stream Analytics job configuration interface for 'asajob001'. The left sidebar has a red box around the 'Query' item under 'Job topology'. The main area shows inputs from 'IoTHubStream' and outputs to 'blobstorage'. A code snippet on the right provides a template for writing queries:

```
1 SELECT
2 *
3 INTO [YourOutputAlias]
4 FROM [YourInputAlias]
```

2. 右側のウィンドウ内のクエリを修正し、[Save]をクリックします。

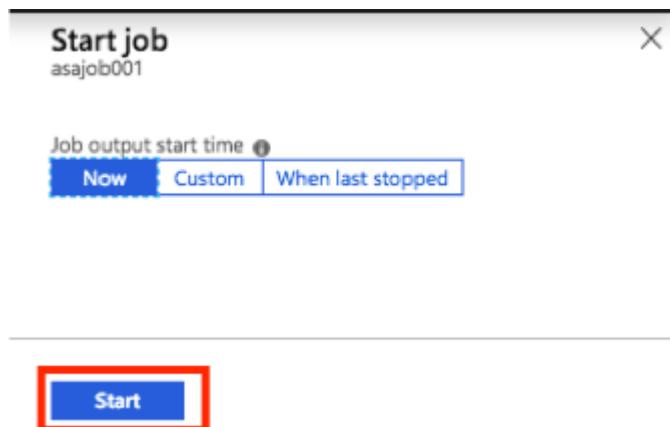
```
SELECT * INTO [blobstorage] FROM [IoTHubStream]
```

The screenshot shows the same configuration interface after modifying the query. The 'Save' button is highlighted with a red box. The query now reads: 'SELECT \* INTO [blobstorage] FROM [IoTHubStream]'. The rest of the interface remains the same.

3. Stream Analyticsの設定メニューにある[Overview]をクリックし、ウィンドウ上部にある[Start]をクリックします。

The screenshot shows the 'Overview' page for the job. The 'Overview' menu item is highlighted with a red box. The top navigation bar has a red box around the 'Start' button. Below the navigation, it says 'Created' and lists the resource group as 'iothandson01'.

4. [ジョブの開始]ウィンドウが表示されたら、ウィンドウ下部の[Start]をクリックします。



## 5. 通知ウィンドウで正常にStream

Analyticsジョブが開始されたかを確認します。（1～2分かかります）

The screenshot shows the Stream Analytics job 'asajob001' in the 'Running' state. A notification in the activity log states: 'Streaming job started successfully.' and 'Started Streaming job 'asajob001' successfully.' by 'me' a few seconds ago.

## 6. 指定したストレージにテレメトリデータが保存されていることを確認します。

「演習：Azure環境の準備」で作成したリソースグループを表示し、作成したストレージアカウントを表示します。[ストレージアカウント]ウィンドウ内のサービス欄の[BLOB]をクリックします。

The screenshot shows the storage account 'iothandsonstorage201809' settings page. In the 'サービス' (Services) section, the 'BLOB' service is highlighted with a red box. It is described as a '非構造化データの REST ベースのオブジェクトストレージ' (Unstructured data REST-based object storage).

7 作成したコンテナ内に指定したパスが作成され、データが保存されていることを確認します。

iothandsonstorage201809 - BLOB

ストレージ アカウント

検索 (Ctrl+ /)

コンテナー 更新 削除

概要

アクティビティ ログ

アクセス制御 (IAM)

タグ

問題の診断と解決

ストレージ アカウント: **iothandsonstorage201809**

プレフィックスによるコンテナーの検索

名前

telemetry

8. ファイル名を選択後、ウィンドウ上部の[BLOBの編集]をクリックし、データを確認します。

9. データを確認後、Stream Analyticsジョブを停止しておきます。

asajob001	Stream Analytics job
<input type="text"/> 検索 (Ctrl+ /)	«
	 Start  Stop  Delete

# Stream Analyticsの出力設定（SQLデータベースへの出力）

このタスクでは、作成したStream 1分間の平均値を保存します。

- 「演習:Azure環境の準備」で作成したリソースグループを表示し、ウィンドウ上部の[+追加]をクリックします。



- Everythingウィンドウの検索ボックスで、"SQL"と入力しEnterキーを押します。[SQL Database]をクリックし、表示された詳細ウィンドウ下部の[作成]をクリックします。

The screenshot shows the 'Everything' search results for 'SQL'. The search bar contains 'SQL'. The results table has columns: '名前' (Name), '公開元' (Owner), and 'カテゴリ' (Category). The rows are:

名前	公開元	カテゴリ
Web App + SQL	Microsoft	Web
SQL Server 2016 SP1 Enterprise on Windows Server 2016	Microsoft	Compute
SQL Database	Microsoft	Databases

# SQL Database

□ ×

\* データベース名

sqldb1208001



\* サブスクリプション



\* リソース グループ

iothandson01



[新規作成](#)

\* ソースの選択

空のデータベース



\* サーバー



sqlsv2018001 (東日本)

SQL エラスティック プールを使用しますか?



はい  後で

\* 価格レベル



Standard S0: 10 DTU、250 GB

\* 照合順序

SQL\_Latin1\_General\_CI\_AS

**作成**

Automation オプション

3. 下記のとおり、各設定値を入力後、ウィンドウ下部にある[作成]をクリックします。

Table 10. 設定項目と設定値

設定項目	設定値
データベース名	任意の名前 例) sqldb2018001
サブスクリプション	ご準備いただいたサブスクリプション（デフォルトのまま）
リソースグループ	既存のものを使用
ソースの選択	空のデータベース（デフォルトのまま）
サーバー	[必要な設定の構成]をクリックし、次の「新しいサーバーの作成」に記載の内容を入力。
新しいサーバーの作成	
サーバー名	任意の名前。Azure全体で一意の名前。 例) sqlsv2018001
サーバー管理者ログイン	任意の管理者IDを設定 例) {sqldb-admin-usernam}
パスワード	任意の管理者パスワードを設定 例) @dmin123
パスワードの確認	パスワードと同じ文字列を設定
場所	東日本（デフォルトのまま）
Azureサービスにサーバーへのアクセスを許可する	チェック（デフォルトのまま）
Advanced Thread Protection	後で（デフォルトのまま）
～新しいサーバーの作成の項目はここまで～	
SQLエラスティックプールを使用しますか？	後で（デフォルトのまま）
価格レベル	[必要な設定の構成]をクリックし、[Basic]をクリック後、下部の[適用]をクリック
照合順序	SQL_Latin1_General_CI_AS（デフォルトのまま）

<p><b>SQL Database</b></p> <p>* データベース名 sqldb1208001</p> <p>* サブスクリプション iothandson01</p> <p>* リソース グループ iothandson01</p> <p>* ソースの選択 ① 空のデータベース</p> <p>サーバー 必要な設定の構成</p> <p>SQL エラスティック プールを使用しますか? ① はい ② 後で</p> <p>価格レベル ① 必要な設定の構成</p> <p>* 照合順序 ① SQL_Latin1_General_CI_AS</p> <p><b>作成</b> Automation オプション</p>	<p><b>サーバー</b></p> <p>+ 新しいサーバーの作成</p>	<p>* サーバー名 sqlsv2018001 .database.windows.net</p> <p>* サーバー管理者ログイン dbadmin</p> <p>* パスワード .....</p> <p>* パスワードの確認 .....</p> <p>* 場所 東日本</p> <p><input checked="" type="checkbox"/> Azure サービスにサーバーへのアクセスを許可する ①</p> <p>Advanced Threat Protection ① <b>無料試用版の開始</b> 後で</p> <p>60 日間の無料試用期間があり、その後はサーバーごとに毎月 1680 JPY となります。</p> <p><b>詳細情報</b></p> <p><b>選択</b></p>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

# SQL Database

□ ×

\* データベース名

sqldb1208001



\* サブスクリプション



\* リソース グループ

iothandson01



[新規作成](#)

\* ソースの選択

空のデータベース



\* サーバー



sqlsv2018001 (東日本)

SQL エラスティック プールを使用しますか?



はい  後で

\* 価格レベル



Standard S0: 10 DTU、250 GB

\* 照合順序

SQL\_Latin1\_General\_CI\_AS

**作成**

Automation オプション

4. リソースグループ内に作成したSQLデータベースが表示されていることを確認します。

名前	種類	場所
sqlsv2018001	SQL Server	米国東部
sqldb1208001 (sqlsv2018001/sqldb1208001)	SQL データベース	米国東部

5. SQLデータベースのメニューから[クエリエディタ]を選択肢、  
[OK]をクリックします。

Table 11. 設定項目と設定値

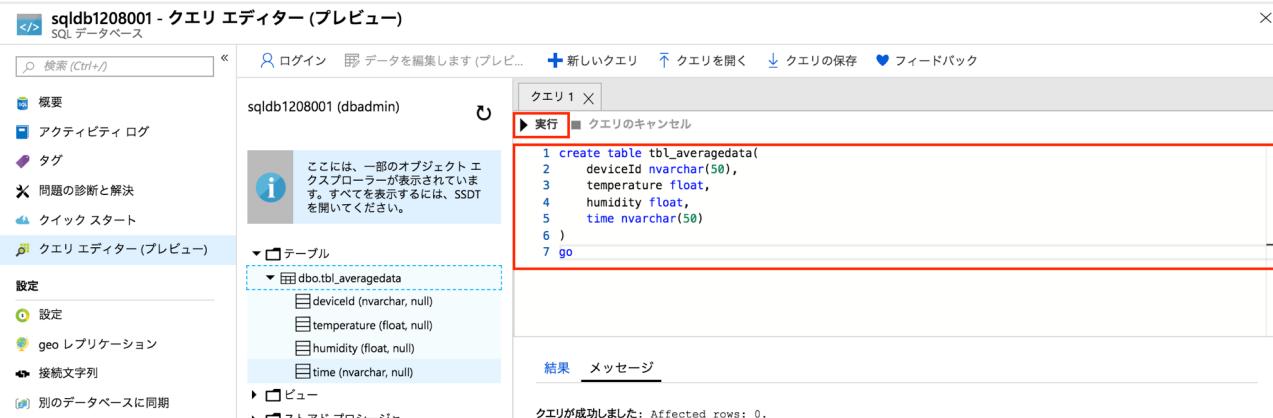
設定項目	設定値
承認の種類	SQL Server認証（デフォルトのまま）
ログイン	データベース作成時に指定したユーザー名 例) dbadmin （デフォルトのまま）
パスワード	データベース作成時に指定したパスワード 例) @dmin123



## 6. 1分平均値を格納するテーブル”tbl\_averagedata”を作成します。

[クエリ 1]に下記のSQLを入力し、[実行]をクリックします。テーブル一覧を更新し、テーブルが正常に作成されていることを確認します。

```
create table tbl_averagedata(
    deviceId nvarchar(50),
    temperature float,
    humidity float,
    time nvarchar(50)
)
go
```



## 7. Stream Analyticsの設定メニューにある[Output]をクリックし、ウィンドウ上部の[+Add]-[SQL Database]をクリックします。

The screenshot shows the Azure Stream Analytics job interface for 'asajob001'. The left sidebar has a search bar and navigation links: 'Job topology', 'Inputs', 'Functions', 'Query', and 'Outputs' (which is highlighted with a red box). The main area shows output options: 'Event Hub', 'SQL Database' (highlighted with a red box), 'Blob storage', 'Table storage', and 'Service Bus topic'.

8. [新規出力]ウィンドウの各設定値を入力後、ウィンドウ下部にある[作成]をクリックします。

Table 12. 設定項目と設定値

設定項目	設定値
出力エイリアス	任意の名前 例) sqldatabase
シンク	SQLデータベース
インポートオプション	サブスクリプションからSQLデータベースを選択する
データベース	作成したデータベースを選択 例) sqldb2018001
サーバー名	自動入力
ユーザー名	SQL Database作成時に指定した管理ユーザー 例) dbadmin
パスワード	SQL Database作成時に指定したパスワード 例) @dmin123
テーブル	tbl_averagedata

# SQL Database



New output

\* Output alias

sqldatabase



Provide SQL Database settings manually

Select SQL Database from your subscriptions

サブスクリプション



Database i

sqldb1208001



Server name

sqlsv2018001.database.windows.net

\* Username

dbadmin



\* Password

.....



\* Table

tbl\_averagedata

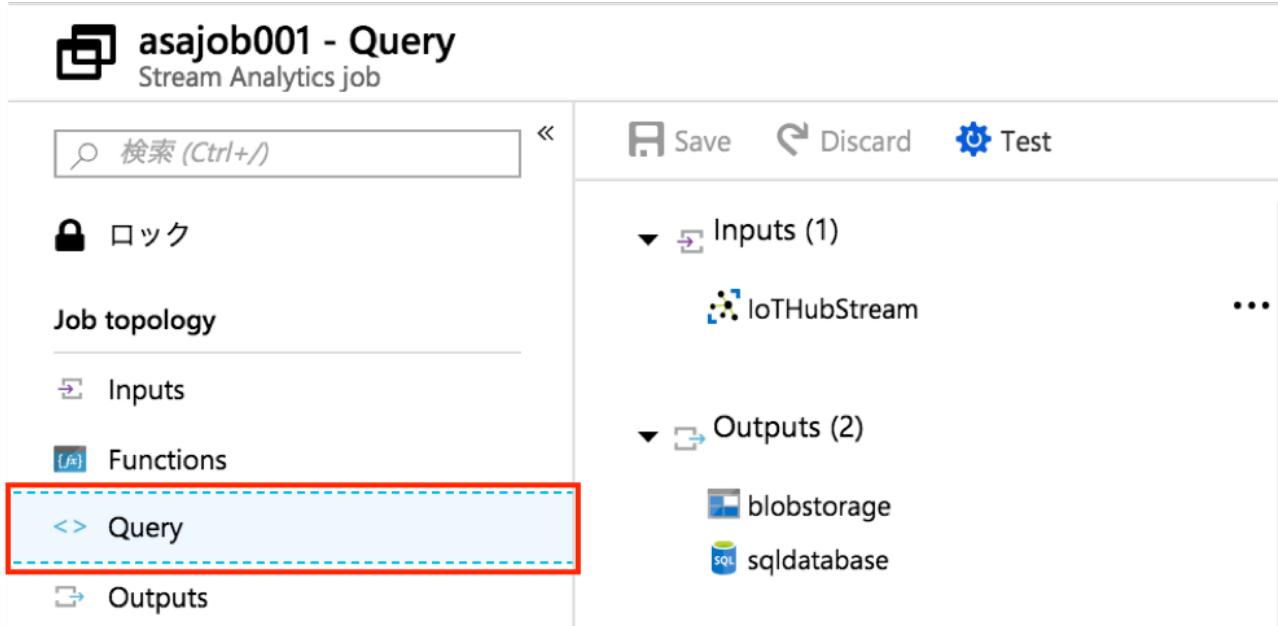


Save

# Stream Analyticsのクエリ設定（SQLデータベースへの出力）

このタスクでは、Stream AnalyticsにSQLデータベースに出力するクエリを追加します。

1. Stream Analyticsの設定メニューにある[Query]をクリックします。



2. 右側のウィンドウ内のクエリを追加し、[保存]をクリックします。

WITH句の中でIoT

Hubにデータを送信したデバイス名を取得し(deviceId)、過去

1分間のデバイス毎の温度/湿度の平均値を取得してSQLデータベースに出力します。

```

WITH [averagedata] AS (
    SELECT
        IoTHubStream.IoTHub.ConnectionDeviceId as deviceId,
        ambient.temperature as temperature,
        ambient.humidity as humidity
    FROM
        [IoTHubStream]
)

SELECT
    deviceId,
    AVG(temperature) as temperature,
    AVG(humidity) as humidity,
    System.Timestamp as time
INTO
    [sqldatabase]
FROM
    [averagedata]
GROUP BY
    deviceId, TumblingWindow(minute, 1)

SELECT * INTO [blobstorage] FROM [IoTHubStream]

```

The screenshot shows the Azure Stream Analytics job editor for a job named "asajob001". The left sidebar contains navigation links for Job topology, Inputs, Functions, Query (which is selected), Outputs, Configure (Scale, Locale, Event ordering, Error policy, Compatibility level), General, and Tools. The top right has Save, Discard, and Test buttons. The main area displays the Stream Analytics query code. A red box highlights the first part of the query:

```

1 WITH [averagedata] AS (
2     SELECT
3         IoTHubStream.IoTHub.ConnectionDeviceId as deviceId,
4         ambient.temperature as temperature,
5         ambient.humidity as humidity
6     FROM
7         [IoTHubStream]
8 )

```

Below this, another red box highlights the rest of the query:

```

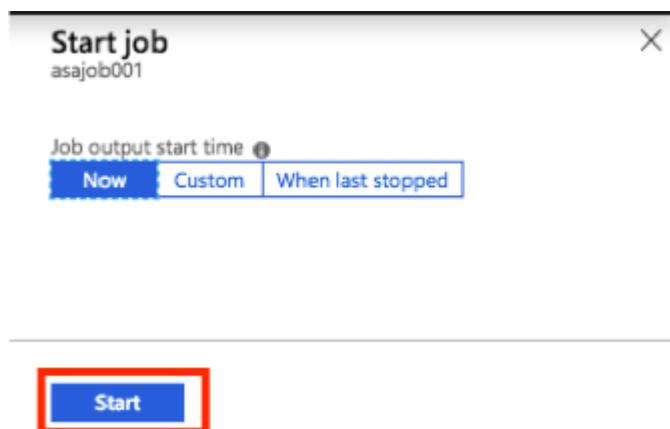
9
10 SELECT
11     deviceId,
12     AVG(temperature) as temperature,
13     AVG(humidity) as humidity,
14     System.Timestamp as time
15 INTO
16     [sqldatabase]
17 FROM
18     [averagedata]
19 GROUP BY
20     deviceId, TumblingWindow(minute, 1)
21
22 SELECT * INTO [blobstorage] FROM [IoTHubStream]

```

3. Stream Analyticsの設定メニューにある[Overview]をクリックし、ウィンドウ上部にある[Start]をクリックします。

The screenshot shows the Azure Stream Analytics Overview page for the "asajob001" job. The top navigation bar includes a search bar, a Start button (highlighted with a red box), a Stop button, and a Delete button. Below the navigation is a status summary: "Created" (blue icon). The main content area is titled "Overview" (highlighted with a red box) and lists "アクティビティ ログ" and "アクセス制御 (IAM)". On the right, there's a "リソース グループ (変更)" section showing "iothandson01" and a "状態" section showing "Running".

4. [ジョブの開始]ウィンドウが表示されたら、ウィンドウ下部の[Start]をクリックします。



5. 通知ウィンドウで正常にStream Analyticsジョブが開始されることを確認します(数分かかります)。

6. SQLデータベースに格納されたデータを確認します。SQLデータベースの[クエリエディター]で、テーブルを選択し、[データを編集します]をクリックして1分間隔に集計されたデータを確認します。

DEVICEID	TEMPERATURE	HUMIDITY	TIME
sampledevice	26.0601329208575	69.4182639917058	2018-09-14T05:58:00.000000Z
sampledevice	25.4237362717751	69.0482657581142	2018-09-14T05:57:00.000000Z
sampledevice	26.3457125470158	68.9988272743568	2018-09-14T06:00:00.000000Z

[新しいクエリ]でSQLを直接入力して[実行]ボタンをクリックしてデータ一覧を確認することもできます。

```
select * from dbo.tbl_averagedata order by time
```

7. データを確認後、Stream Analyticsジョブを停止しておきます。

# 演習：Power BIでのデータ表示

このタスクでは、Stream AnalyticsからのストリーミングデータをPower BIでリアルタイム表示します。Stream AnalyticsにPower BI用の出力を追加し、Power BI側でストリーミングデータセットとして扱います。

## Stream Analyticsの出力設定（PowerBIへの出力）

1. Stream Analyticsを停止します。
2. Stream Analyticsの新規出力設定でpowerbiを追加します。

The screenshot shows the 'Outputs' section of the Stream Analytics job 'asajob001'. On the left, there's a sidebar with links like Overview, Activity Log, IAM, Tags, Troubleshoot, Settings (with Lock), Job topology, Inputs, Functions, Query, and Outputs. The 'Outputs' link is highlighted with a red box. On the right, there's a list of output types: Event Hub, SQL Database, Blob storage, Table storage, Service Bus topic, Service Bus queue, Cosmos DB, Power BI, and Data Lake Storage Gen1. The 'Power BI' option is also highlighted with a red box.

3. 設定項目を入力後、Power BIの[Authorize]をクリックし、Power BIユーザーでサインインします。別のブラウザウィンドウが起動します。

Table 13. 設定項目と設定値

設定項目	設定値
出力エイリアス Output alias	任意の名前 例) powerbi
グループワークスペース Group workspace	My workspace
データセット名 Dataset name	任意の名前 例) iothandsdataset001
テーブル名 Table name	任意の名前 例) iothandsontable001

**Power BI**  
New output

\* Output alias  
powerbi

Group workspace  
Authorize connection to load workspaces

\* Dataset name  
iothandsdataset001

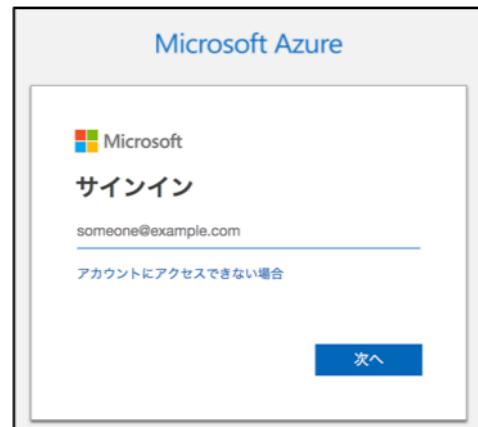
\* Table name  
iothandsontable001

**Authorize connection**  
You'll need to authorize with Power BI to configure your output settings.

**Authorize**

Don't have a Microsoft Power BI account yet?  
[Sign up](#)

**i** Note: You are granting this output permanent access to your Power BI dashboard. Should you need to revoke this access in the future you can do one of the following:  
1. Change the user account password.  
2. Delete this output.  
3. Delete this job.



4. Power BIの認証が完了したら、[作成]をクリックします。

## Power BI



New output

\* Output alias

Group workspace

\* Dataset name

\* Table name

Currently authorized as [Iname\(someone@example.com\)](#)

### Authorize connection

You'll need to authorize with Power BI to configure your output settings.



Note: You are granting this output permanent access to your Power BI dashboard. Should you need to revoke this access in the future you can do one of the following:

1. Change the user account password.
2. Delete this output.
3. Delete this job.

**TIP**

下記URLに詳細説明の記載があります。  
<https://docs.microsoft.com/ja-jp/azure/stream-analytics/stream-analytics-power-bi-dashboard>

5. Stream Analyticsのクエリの末尾にPower BI出力用のクエリを追加します。過去15秒間のデバイス毎の温度/湿度の平均値を取得してPower BIにプッシュで出力します。

```
WITH [averagedata] AS (
    SELECT
        IoTHubStream.IoTHub.ConnectionDeviceId as deviceId,
        ambient.temperature as temperature,
        ambient.humidity as humidity
    FROM
        [IoTHubStream]
)

SELECT
    deviceId,
    AVG(temperature) as temperature,
    AVG(humidity) as humidity,
    System.Timestamp as time
INTO
    [sqldatabase]
FROM
    [averagedata]
GROUP BY
    deviceId, TumblingWindow(minute, 1)

SELECT * INTO [blobstorage] FROM [IoTHubStream]

SELECT
    deviceId,
    AVG(temperature) as temperature,
    AVG(humidity) as humidity,
    System.Timestamp as time
INTO
    [powerbi]
FROM
    [averagedata]
GROUP BY
    deviceId, TumblingWindow(second, 15)
```

6. クエリを設定後、ジョブを開始します。

Stream AnalyticsのOverviewで[Start]をボタンをクリックします。

The screenshot shows the Azure Stream Analytics job overview page for 'asajob001'. At the top right, there are three buttons: 'Start' (highlighted with a red box), 'Stop', and 'Delete'. Below these buttons, the status is shown as 'Created'. On the left, there's a sidebar with links: 'Overview' (highlighted with a red box), 'アクティビティ ログ', and 'アクセス制御 (IAM)'. To the right, it shows the resource group 'iothandson01' and the status 'Running'.

ダイアログで[Start]ボタンをクリックしてJobをスタートします。

The screenshot shows the 'Start job' dialog box for the 'asajob001' job. It has a 'Job output start time' section with three options: 'Now' (highlighted with a red box), 'Custom', and 'When last stopped'. At the bottom right, there is a large blue 'Start' button, which is also highlighted with a red box.

## 7. Power BIにログインします。

<https://powerbi.microsoft.com/> にアクセスし、上部の[サインイン]をクリックしてサインインします。

The screenshot shows the Microsoft Power BI sign-in page. It features the Microsoft logo and the 'Power BI' logo. On the right side, there is a red-outlined 'サインイン' (Sign In) button.

## 8. サイドメニューからマイワークスペースを開き、ウィンドウ右上の[+作成]をクリックします。

The screenshot shows the Microsoft Power BI dashboard. On the left, there is a sidebar with various options: 'ホーム (プレビュー)', 'お気に入り', '最近', 'アプリ', '自分と共有', 'ワークスペース', and 'マイワークスペ...'. The 'マイワークスペ...' item is highlighted with a red box. At the top right, there is a red-outlined '+ 作成' (Create) button. The main area shows a search bar, a navigation bar with 'ダッシュボード', 'レポート', 'ブック', and 'データセット', and a list of items with columns for '名前' (Name), 'アクション' (Actions), and '所' (Location).

9. [ダッシュボード]を選択し、任意の名前を設定します。（例：iothandsondashboard001）



### ダッシュボードの作成

ダッシュボード名

iothandsondashboard001

作成 キャンセル

This is a modal dialog titled 'ダッシュボードの作成'. It has a text input field for 'ダッシュボード名' containing 'iothandsondashboard001'. At the bottom are two buttons: a yellow '作成' button and a grey 'キャンセル' button. The '作成' button is highlighted with a red border.

10. ウィンドウ上部の[+タイルの追加]をクリックし、リアルタイムデータ欄の[カスタムストリーミングデータ]を選択し、[次へ]をクリックします。

### タイルの追加

ソースの選択

ビデオ

### リアルタイム データ

((o))

カスタム ストリーミング データ

次へ キャンセル

This is a modal dialog titled 'タイルの追加'. It has a section 'ソースの選択' with a 'ビデオ' option. Below it is a section 'リアルタイム データ' containing a placeholder '(o)' and a red-bordered item 'カスタム ストリーミング データ'. At the bottom are two buttons: a yellow '次へ' button and a grey 'キャンセル' button. The '次へ' button is highlighted with a red border.

11. Stream Analyticsのデータセット名に設定したデータセット（例：iothandsondataset001）が、データセット欄に表示されていることを確認してください。表示されていればこれを選択し、[次へ]をクリックします。

## カスタム ストリーミング データ タイ ルの追加

ストリーミング データセットの選択

＋ ストリーミング データセットの追加

### データセット

iothandsondataset001

データセットの管理

戻る

次へ

キャンセル

#### CAUTION

Stream

Analyticsジョブが正常に動作していないと、データセットとして表示されません。

12. 視覚化タイプで[カード]を選択し、フィールドタブの[+値の追加]をクリックします。

# カスタムストリーミングデータ タイ ルの追加

ストリーミングデータセットの選択 > 視覚化デザイン



13. プルダウンから”temperature”を選択します。

# カスタム ストリーミング データ タイ ルの追加

ストリーミング データセットの選択 > 視覚化デザイン

視覚化タイプ

カード



フィールド

temperature



データセットの管理

戻る

次へ

キャンセル

14. データラベルのタブを選択し、[小数点以下桁数の値]を設定します。(例: “2”)

# カスタム ストリーミング データ タイ ルの追加

ストリーミング データセットの選択 > 視覚化デザイン



15. ウィンドウ下部の[次へ]をクリックします。



16. [タイトル]、[サブタイトル]に任意の文字列をして、[適用]をクリックします。

# タイルの詳細

\* 必須

## 詳細

タイトルとサブタイトルの表示

### タイトル

温度

### サブタイトル

既定値に戻す

技術的な詳細

戻る

適用

キャンセル

ダイアログ表示されますが、少し待っていると表示されなくなります。  
すぐに非表示にしたい場合は、右上の×ボタンをクリックしてください。

CAUTION



### ダッシュボードに追加

×

ダッシュボードに視覚エフェクトを追加しました。ダッシュボードの Phone ビューを最適化するため、ビューに含めるタイルのサイズ変更、並べ替え、サブセットの選択を行うことができます。

Phone ビューの作成

1. 目的に応じて、ダッシュボードを完成させてください。

**TIP**

Power

BI

サービスのダッシュボードの詳細は、下記をご参照ください。

<https://powerbi.microsoft.com/ja-jp/documentation/powerbi-service-dashboards/>

2. タイルの視覚化タイプに”折れ線グラフ”を選択した場合

# カスタム ストリーミング データ タイ ルの追加

ストリーミング データセットの選択 > 視覚化デザイン

視覚化タイプ

折れ線グラフ

軸

time

+ 値の追加

凡例

deviceid

+ 値の追加

値

temperature

+ 値の追加

表示する時間枠

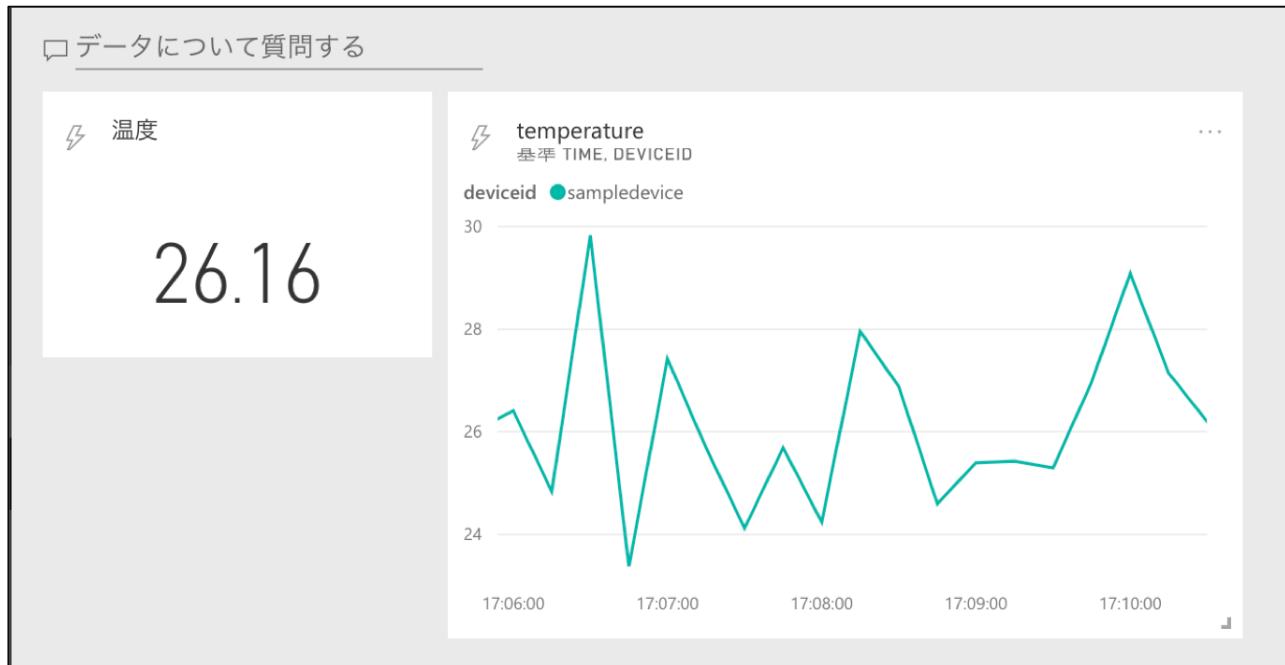
最後 5 分

データセットの管理

戻る 次へ キャンセル

Table 14. 設定項目と設定値

設定項目	設定値
軸	time
凡例	deviceid
値	temperature
表示する時間枠	最後：5分



## 演習: Azure IoT Edge デバイスの作成

この演習では、Azure上にデプロイしたVM(Ubuntu LTS 18.04)をAzure IoT Edgeデバイスとして作成し、シミュレーションデータ（温度/湿度）をIoT Hubに送信します。

### タスク1：Linux VMのデプロイ

このタスクでは、IoT Edgeデバイスとして動作させる仮想マシン(Ubuntu LTS 18.04)をデプロイします。

1. Azure管理ポータル画面のサイドメニューから[リソースグループ]をクリックして、先に作成したリソースグループ(ex. iothandson)を選択します。
2. リソースグループの[追加]をクリックします。
3. "Ubuntu"で検索した結果から"Ubuntu LTS 18.04"を選択し、[作成]をクリックします。

▼ フィルター

 ×

結果

名前	公開元	カテゴリ
 Ubuntu Server 16.04 LTS	Canonical	Compute
 Ubuntu Server 18.04 LTS 	Canonical	Compute 

1. [基本設定の構成]で各種項目を設定します。

# 設定

□ ×

## 高可用性

### 可用性ゾーン ⓘ

なし



選択した場所で利用できる Availability Zones はありません。Availability Zones をサポートしている場所を表示するには、  
[aka.ms/zonedregions](http://aka.ms/zonedregions) に移動します

### \* 可用性セット ⓘ



なし

## Storage

### マネージド ディスクを使用 ⓘ

いいえ

はい

### OS ディスク サイズ ⓘ

既定のサイズ (30 GiB)



## ネットワーク

### \* Virtual Network ⓘ



iothandson-vnet

### \* サブネット ⓘ



default (172.16.1.0/24)

### \* パブリック IP アドレス ⓘ



(新規) edgeDevice-ip

### ネットワーク セキュリティ グループ ⓘ

Basic

詳細

### \* パブリック受信ポートを選択 ⓘ

0 項目が選択されました



トトコニルルルル

OK

Table 15. 設定項目と設定値

設定項目	設定値
名前	任意のマシン名
VMディスクの種類	HDDまたはSSD
ユーザー名	任意のユーザー名（例：iotuser）
認証の種類	[パスワード]を選択
パスワード	次の条件を満たす、任意のパスワード (例：!IoTPass1234) * 12-72文字 * 1つの小文字、1つの大文字、1つの数字および "¥"または"-\"以外の1つの特殊文字
Azure Active Directoryでログインする（プレビュー）	[無効]を選択
サブスクリプション	利用するサブスクリプションを選択
リソースグループ	[既存のものを使用]を選択し、先に作成した iothandson を選択
場所	任意の場所を選択

2. [仮想マシンのサイズの選択]でサイズを選択します。

（本ハンズオンでは 8GB 以上のメモリが割り当てられるスペックを選択します）

3. [オプション機能の構成]でパブリック受信ポートを選択で"SSH(22)"を選択します。

\* パブリック受信ポートを選択 

SSH (22) 



4. [概要]で[作成]ボタンをクリックします。

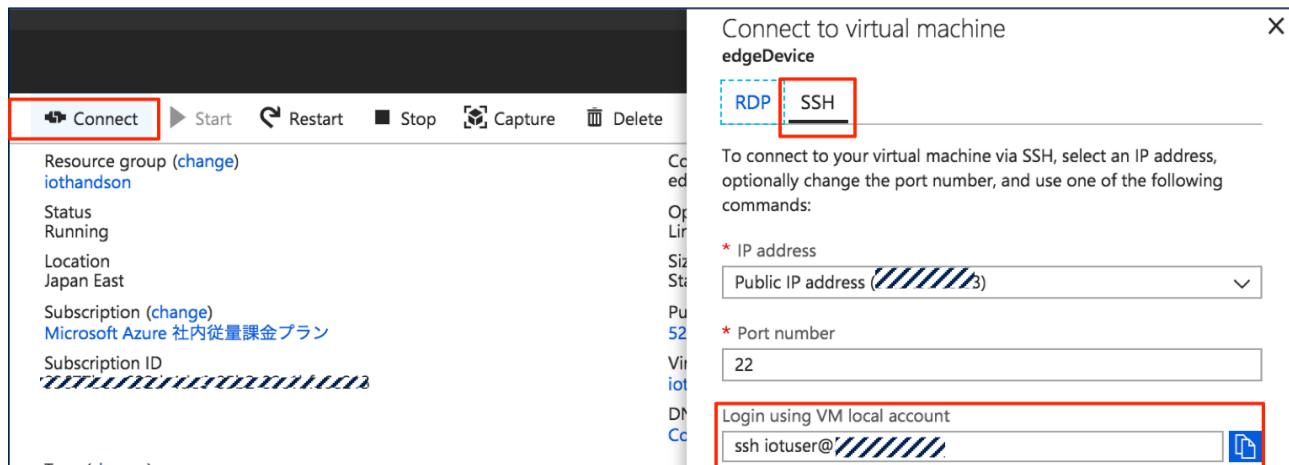
## タスク2：コンテナランタイムのインストール

このタスクでは、IoT Edgeのランタイムをインストールします。

Edgeデバイス用にデプロイしたVMにIoT

1. 接続文字列を確認します。

VMのOverviewを開き[CONNECT]をクリックし、右に表示されるダイアログで[SSH]タブを選択します。 Login using VM local accountに記載されている文字列をコピーします。



## 2. Cloud Shell またはSSHクライアントでVMに接続します。

前の手順でコピーした接続文字列をCloud Shellにペーストしてログインします。

```
ssh iotuser@[IPアドレス]
```

## 3. root ユーザにスイッチします

```
sudo -i
```

## 4. パッケージを取得するリポジトリを設定します。

```
curl https://packages.microsoft.com/config/ubuntu/16.04/prod.list > ./microsoft-prod.list
cp ./microsoft-prod.list /etc/apt/sources.list.d/
curl https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor > microsoft.gpg
cp ./microsoft.gpg /etc/apt/trusted.gpg.d/
```

## 5. コンテナの実行エンジンのMobyをインストールします。

Moby(<http://mobyproject.org>) は、オープンソース版のDockerの実行環境です。

```
apt-get -y update
apt-get -y install moby-engine
apt-get -y install moby-cli
```

## 6. Moby が正常にインストールされたことを確認します。

```
docker run hello-world
```

以下のメッセージが表示されれば、mobyが正しくインストールされています。

Hello from Docker!  
This message shows that your installation appears to be working correctly.

## タスク3：IoT Edgeデバイスの追加

このタスクでは、IoT HubにIoT Edgeデバイスを新規登録します。

1. Azure管理ポータルで、先に作成したIoT Hub iothub-xx を開きます。
2. IoT Edgeをクリックして、[+ Add IoT Edge Device]をクリックします。

The screenshot shows the Azure IoT Edge Devices management interface. On the left, there's a sidebar with 'Search (Ctrl+ /)' and sections for 'Locks', 'Automation script', 'EXPLORERS' (Query explorer, IoT devices), and 'AUTOMATIC DEVICE MANAGEMENT' (IoT Edge, IoT device configuration). The main area has a header with '+ Add IoT Edge Device' (highlighted with a red box), 'Add IoT Edge Deployment', 'Refresh', and 'Delete'. Below the header, there's a brief description of what IoT Edge does. Under 'IoT Edge Devices', there's a section titled 'IoT Edge Devices' with a cloud icon containing a microchip. A note says: 'IoT Edge devices have the IoT Edge runtime installed and are flagged as "IoT Edge device" in the device details. Each IoT Hub supports up to 1000 IoT Edge devices.' At the bottom, there's a 'Query' section with a placeholder 'SELECT \* FROM devices WHERE'.

3. Device IDを入力して[Save]ボタンをクリックします

The screenshot shows the 'Add Device' dialog box. It includes fields for 'Device ID' (with 'msiotedge001' entered), 'Authentication Type' (set to 'Symmetric Key'), 'Primary Key' (placeholder 'Enter your primary key here'), 'Secondary Key' (placeholder 'Enter your secondary key here'), 'Auto Generate Keys' (checkbox checked), and 'Connect device to IoT Hub' (checkbox checked, with 'Enable' selected). At the bottom is a 'Save' button.

4. IoT Edge Devideを選択して接続文字列をコピーしておきます

[IoT Edge Devices](#) [IoT Edge Deployments](#)



## IoT Edge Devices

**i** IoT Edge devices have the IoT Edge runtime installed and are flagged as "IoT Edge device" in the device details. Each IoT Hub supports up to 1000 IoT Edge devices.

Query [?](#)

```
SELECT * FROM devices WHERE
optional (e.g. tags.location='US')
```

[Execute](#)

DEVICE ID	RUNTIME RESPONSE	MODULE COUNT	CONNECTED CLIENT ...	DEPLOYMENT COUNT
msiotedge001	N/A	0	0	0

**Device Details** msiotedge001

Save Regenerate keys Device Twin Message To Device Direct Method Set Modules Refresh

Device Id [?](#) msiotedge001 [Copy](#)

Primary key [?](#) [Copy](#)

Secondary key [?](#) [Copy](#)

Connection string—primary key [?](#) [Copy](#)

Connection string—secondary key [?](#) [Copy](#)

Connect device to IoT Hub [?](#)

[Enable](#) [Disable](#)

## タスク4：IoT Edgeランタイムのインストール

このタスクでは、先にデプロイした Ubuntu VM に IoT Edgeランタイムをインストールします。

Ubuntu VMにSSHでログインして、以降のコマンドを順次実行します。

### 1. SSHでログイン

```
ssh iotuser@[IPアドレス]
```

2. rootユーザにスイッチします

```
sudo -i
```

3. IoT Edgeパッケージを取得します

```
apt-get -y install iotedge
```

4. IoT Edgeランタイムを構成します。

`/etc/iotedge/config.yaml` を編集して、デバイスの接続文字列を設定します。設定文字列の定義は26行目あたりにあります。

<ADD DEVICE CONNECTION STRING HERE>  
を、前のタスクでコピーしておいた接続文字列で置き換えます。

編集前

```
provisioning:  
  source: "manual"  
  device_connection_string: "<ADD DEVICE CONNECTION STRING HERE>"
```

編集後

```
provisioning:  
  source: "manual"  
  device_connection_string: "HostName=xxxxx;DeviceId=xxx;SharedAccessKey=xxxxxxxx"
```

5. IoT Edgeランタイムを開始します

```
systemctl start iotedge
```

6. IoT Edgeランタイムのコンテナが動作していることを確認します

```
systemctl status iotedge
```

7. IoT Edgeランタイムのコンテナログを確認します

```
docker ps  
docker log edgeAgent
```

`watch -n 10 docker ps` とすると、10秒間隔で`docker ps`コマンドを実行します。

8. Azure管理ポータルから、対象のIoT Edgeデバイスを開き[最新の情報に更新]をクリックします

Device Details  
msiotedge001

Save Regenerate keys Device Twin Message To Device Direct Method Set Modules Refresh

Device Id msiotedge001

9. デプロイされたモジュール一覧に \$edgeAgent が追加されていることを確認します

Device Details  
msiotedge001

Save Regenerate keys Device Twin Message To Device Direct Method Set Modules Refresh

Device Id msiotedge001

Primary key

Secondary key

Connection string—primary key

Connection string—secondary key

Connect device to IoT Hub

Enable Disable

Edge Runtime Response

N/A

Connected

Modules IoT Edge Hub Connections Deployments

This section specifies desired and reported modules. To adjust desired modules, click "Set Modules" or create a Deployment targeting the device. A delay may occur between the time a module is added or removed and the change is reported by the device. The "Reported by Device" column indicates whether a desired module has been reported by the device. The "Specified in Deployment" column indicates whether the module is specified in the applied deployment. A device may host up to ten modules.

NAME	SPECIFIED IN DEPLOYMENT	REPORTED BY DEVICE	RUNTIME STATUS	EXIT CODE	LAST START TIME (UTC)	TYPE
\$edgeAgent	No	Yes	running	-	Tue Jul 17 2018 14:28:02 ...	System module

## タスク5：IoT Edgeデバイスにシミュレーションモジュールを追加

このタスクでは、IoT

Edgeにマイクロソフトが今回するセンサーシミュレーションモジュールを追加します。センサーシミュレーションは、擬似的にテレメトリデータを生成した送信するモジュールです。

1. Azure管理ポータルから、対象のIoT Edgeデバイスを開き[Set Modules]をクリックします

Device Details  
msiotedge001

Save Regenerate keys Device Twin Message To Device Direct Method Set Modules Refresh

Device Id msiotedge001

2. Add Modulesの[+追加]-[IoT Edge Module]をクリックします

1 Add Modules  
(optional)

2 Specify Routes  
(optional)

3 Review Deployment



You can specify credentials to container registries hosting module images. Listed credentials are used to retrieve modules with a matching URL. The Edge Agent will report error 500 if it can't find a container registry setting for a module.

#### Container Registry Settings

NAME	ADDRESS	USER NAME	PASSWORD
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>



An IoT Edge module is a Docker container you can deploy to IoT Edge devices. It communicates with other modules and sends data to the IoT Edge runtime. Using this UI you can import Azure Service IoT Edge modules or specify the settings for an IoT Edge module. Setting modules on each device will be counted towards the quota and throttled based on the IoT Hub tier and units. For example, for S1 tier, modules can be set 10 times per second if no other updates are happening in the IoT Hub.



#### Deployment Modules

Add  Delete	DESIRED STATUS
IoT Edge Module	
Azure Stream Analytics Module	running
Azure Machine Learning Module	

IoT Edge Modulesで項目を入力して[Save(保存)]ボタンをクリックします

設定項目	設定値
名前(name)	tempsensor
Image url	mcr.microsoft.com/azureiotedge-simulated-temperature-sensor:1.0

3. [Set Module]の画面に戻るので[Next]をクリックします

**Set Modules**

1 Add Modules (optional)    2 Specify Routes (optional)    3 Review Deployment

**Container Registry Settings**

NAME	ADDRESS	USER NAME	PASSWORD
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

**Deployment Modules**

**Add** **Delete**

<input type="checkbox"/> NAME	DESIRED STATUS
<input type="checkbox"/> tempsensor	running

[Configure advanced Edge Runtime settings](#)

[Previous](#) **Next** [Submit](#)

An IoT Edge module is a Docker container you can deploy to IoT Edge devices. It communicates with other modules and sends data to the IoT Edge runtime. Using this UI you can import Azure Service IoT Edge modules or specify the settings for an IoT Edge module. Setting modules on each device will be counted towards the quota and throttled based on the IoT Hub tier and units. For example, for S1 tier, modules can be set 10 times per second if no other updates are happening in the IoT Hub.

この画面では、メッセージのルーティングを定義します。 デフォルトの `` "route": "FROM /\* INTO \$upstream" `` では、全てのモジュールの出力メッセージが \$upstream (IoT Hubを示す予約語) に送信されます。

The screenshot shows the 'Set Modules' interface. At the top, there's a header with a gear icon and the text 'Set Modules'. Below it is a message: 'You can set routes between modules, which gives you the flexibility to send messages where they need to go without additional services to process messages or to write additional code.' A code editor window displays the following JSON code:

```

1  {
2    "routes": [
3      "route": "FROM /* INTO $upstream"
4    ]
5  }

```

At the bottom of the interface are two buttons: 'Previous' and 'Next', with 'Next' being highlighted.

4. 次の画面で[Submit]をクリックします
5. [Refresh]をクリックして **\$edgeHub** (IoT Edgeランタイムモジュールの一つ) と `tempsensor` モジュールが追加されていることを確認します

The screenshot shows the 'Device Details' page for device 'msiotedge001'. At the top, there's a toolbar with 'Save', 'Regenerate keys', 'Device Twin', 'Message To Device', 'Direct Method', 'Set Modules', and a 'Refresh' button (which is highlighted with a red box). Below the toolbar, the 'Device Id' is listed as 'msiotedge001'.

The main area has tabs for 'Connected', 'Modules', 'IoT Edge Hub Connections', and 'Deployments'. The 'Modules' tab is selected. A message box provides instructions for managing modules.

NAME	SPECIFIED IN DEPL...	REPORTED BY DEVI...	RUNTIME STATUS	EXIT CODE	LAST START TIME (...)	TYPE
\$edgeAgent	✓ Yes	✓ Yes	running	-	Tue Jul 17 2018 14:28:...	System module
<b>\$edgeHub</b>	✓ Yes	✓ Yes	running	-	Tue Jul 17 2018 15:06:...	System module
<b>tempsensor</b>	✓ Yes	✓ Yes	running	-	Tue Jul 17 2018 15:06:...	Custom module

tempsensorはシミュレーションデバイス、edgeHub(=Edge Hub(IoT Edgeランタイム))を示しています

6. IoT EdgeがインストールされたUbuntu VMでDockerコンテナー一覧を確認します。

```
docker ps
```

## 7. IoT Edgeランタイム(Edge Hub)コンテナのログを確認します

```
docker logs edgeHub
```

## 8. シミュレーションデバイスコンテナのログから、送信メッセージを確認します。

```
docker logs tempsensor
```

シミュレーションデバイスで生成され、IoT Hubに送信されるデータの例

```
[  
  {  
    "machine": {  
      "temperature": 105.82414784646771,  
      "pressure": 10.663510514154549  
    },  
    "ambient": {  
      "temperature": 20.848158501250744,  
      "humidity": 25  
    },  
    "timeCreated": "2017-11-28T09:29:34.4441434Z"  
  }  
]
```

IoT Hubに送信されたメッセージは、Time Series Insightsなどで確認することができます。

NOTE

IoT EdgeにStream Analytics、Machine Learning、  
Funcsionsのモジュールを使用する場合は、以下のリンクをご参照ください（チュートリアルメニューの下に各モジュールの設定手順が掲載されています）  
<https://docs.microsoft.com/ja-jp/azure/iot-edge/>

以上で、ハンズオンは終了です。お疲れ様でした！！