Innovation Project:

Automatically Visualizing 3D Manufacturing Line with Azure Digital Twins

Contents

Introduction

Azure Digital Twins (ADT) is a platform that allows creating and interacting with digital representations of environments, things, people and their relationships. While 3D visualization of them allows user to navigate, understand and monitor data in Azure digital twins in an extremely simple and intuitive way.

Our toolkit is based on Visualizing Azure Digital Twins in 3D - Microsoft Tech Community, which greatly combines Azure Digital Twins with web application to display device 3D models and their real-time telemetry data on a web page. And we use Azure CLI and ARM templates to automate the process of building and connecting all the required Azure resources such as Digital Twins, Azure function and Event Grid Subscription.

In this user guide, we’ ll show you how to use our automation toolkit at https://github.com/NingHsia/Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins/ to automatically visualize 3D Manufacturing Line with Azure Digital-Twins.

Prerequisites

Before getting started, please make sure you meet the following criteria:

- Own a Windows environment
- Own a device which is already connected to Azure IoT Hub
- Own an Azure account and a Azure subscription
- Go https://github.com/NingHsia/Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins/ and download the source code

Getting started

1. Download software required
2. Run the automation code

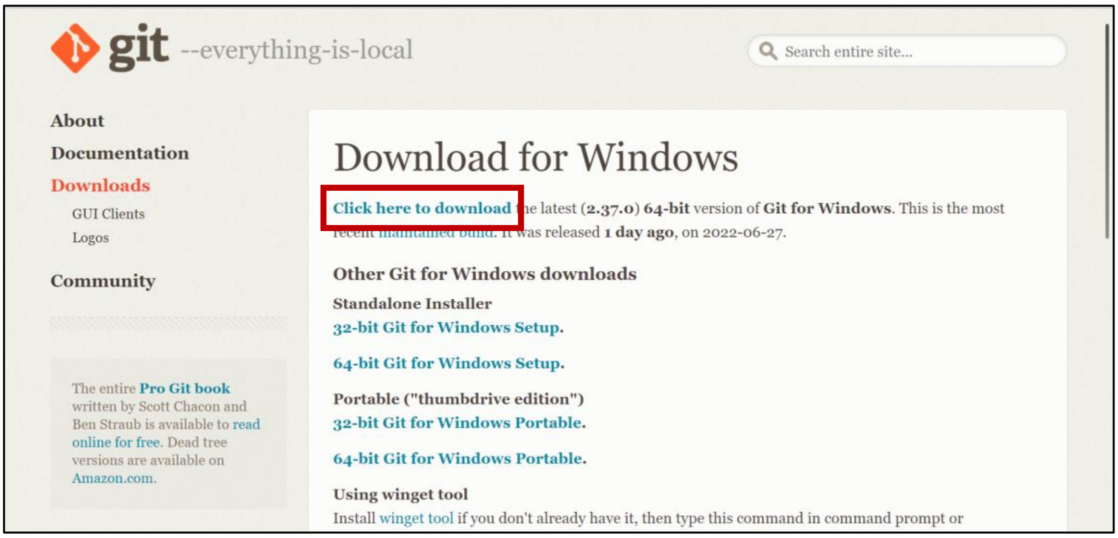
1. Download software required

The following is all the software you need to install before you first run the automation code.

software name	version
1. Git for Windows 64-bit	2.37.0
2. Azure CLI	2.33.1
3. Python	3.10
4. zip and bzip2 for Git Bash on Windows	zip: 3.0 bzip2: 1.0.5
5. jq for Windows 64-bit	1.6

a. Git for Windows 64-bit 2.37.0

- Go https://git-scm.com/download/win and download the Git for Windows (2.37.0) 64-bit.



The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to read online for free. Dead tree versions are available on Amazon.com.

Download for Windows

Click here to download the latest (**2.37.0**) **64-bit** version of **Git for Windows**. This is the most recent maintained build. It was released **1 day ago**, on 2022-06-27.

Other Git for Windows downloads

Standalone Installer

32-bit Git for Windows Setup.

64-bit Git for Windows Setup.

Portable ("thumbdrive edition")

32-bit Git for Windows Portable.

64-bit Git for Windows Portable.

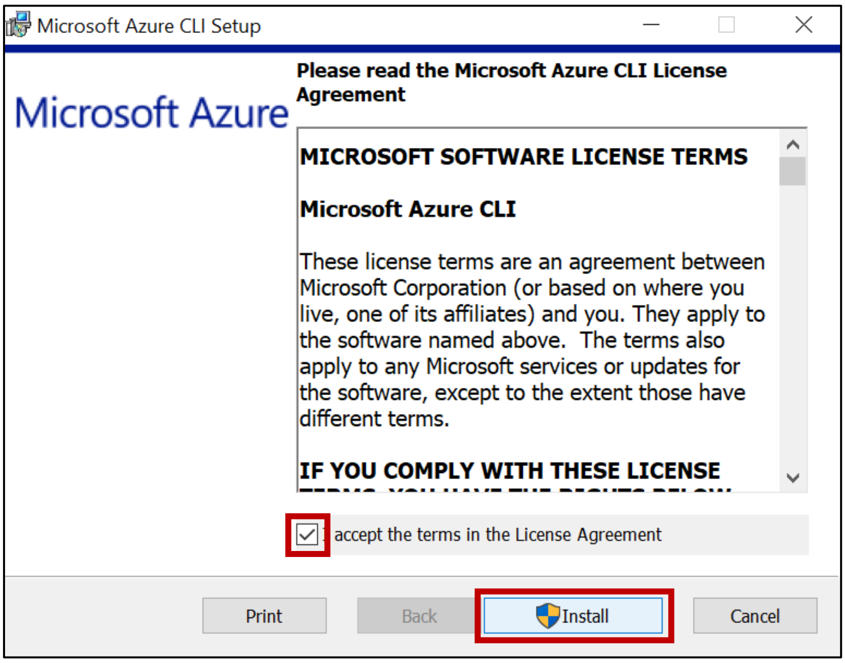
Using winget tool

Install winget tool if you don't already have it, then type this command in command prompt or

- Install it **with default path.**

b. Azure CLI 2.33.1

- Go https://azcliprod.blob.core.windows.net/msi/azure-cli-2.33.1.msi and download & install Azure CLI (2.33.1)



Please read the Microsoft Azure CLI License Agreement

MICROSOFT SOFTWARE LICENSE TERMS

Microsoft Azure CLI

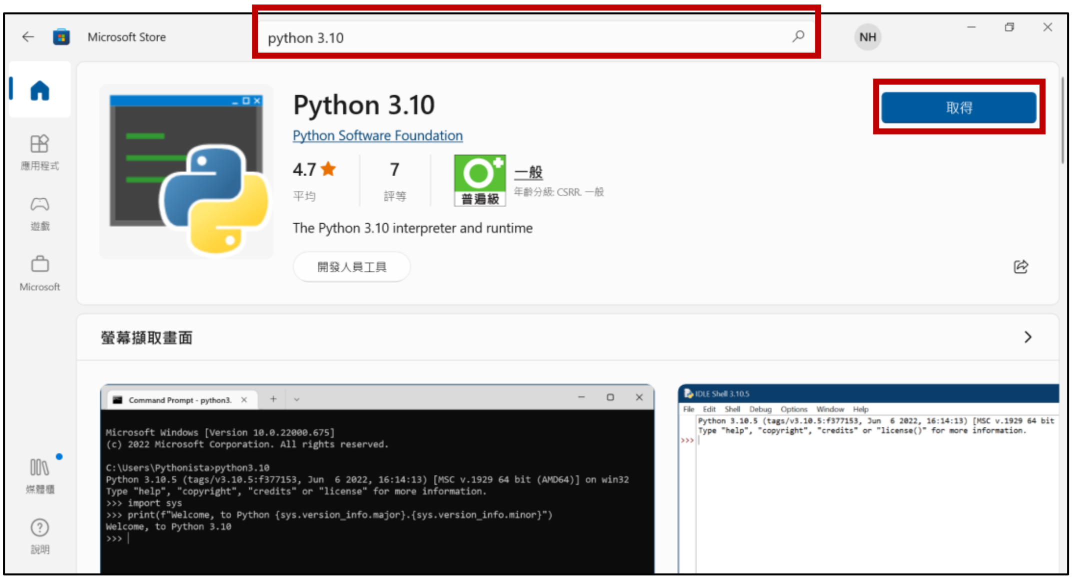
These license terms are an agreement between Microsoft Corporation (or based on where you live, one of its affiliates) and you. They apply to the software named above. The terms also apply to any Microsoft services or updates for the software, except to the extent those have different terms.

IF YOU COMPLY WITH THESE LICENSE

☑ I accept the terms in the License Agreement

c. Python 3.10

- Go Microsoft Store and install "Python 3.10"



python 3.10

Python 3.10

Python Software Foundation

4.7 ★ 平均 | 7 評等 | 普遍級 一般 年齡分級: CSRR. 一般

The Python 3.10 interpreter and runtime

```
Microsoft Windows [Version 10.0.22000.675]
(c) 2022 Microsoft Corporation. All rights reserved.

C:\Users\Pythonista>python3.10
Python 3.10.5 (tags/v3.10.5:f377153, Jun  6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import sys
>>> print(f"Welcome, to Python {sys.version_info.major}.{sys.version_info.minor}")
Welcome, to Python 3.10
>>>
```

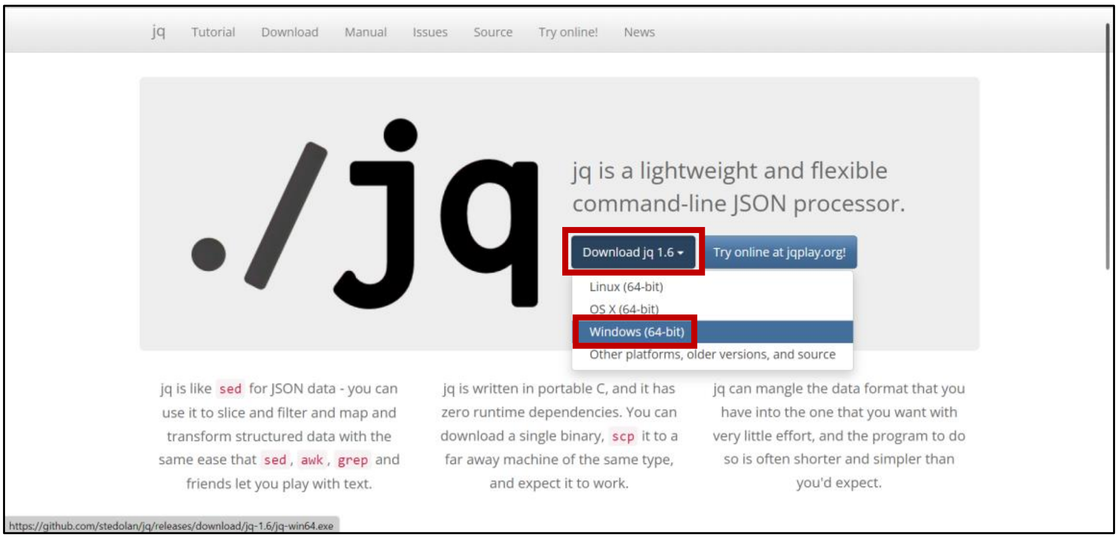
```
Python 3.10.5 (tags/v3.10.5:f377153, Jun  6 2022, 16:14:13) [MSC v.1929 64 bit
Type "help", "copyright", "credits" or "license()" for more information.
>>>
```

d. zip 3.0 and bzip2 1.0.5 for Git Bash on Windows

- Go https://sourceforge.net/projects/gnuwin32/files/ and download
  - zip/3.0/zip-3.0-bin.zip
  - bzip2/1.0.5/bzip2-1.0.5-bin.zip
- Unzip the two zip files and find "zip.exe" under zip-3.0-bin\bin and "bzip2.dll" under bzip2-1.0.5-bin\bin
- Put "zip.exe" and "bzip2.dll" to C:\Program Files\Git\mingw64\bin

e. jq for Windows 64-bit

- Go https://stedolan.github.io/jq/ and download jq 1.6 for Windows 64-bit (jq-win64.exe)



jq is a lightweight and flexible command-line JSON processor.

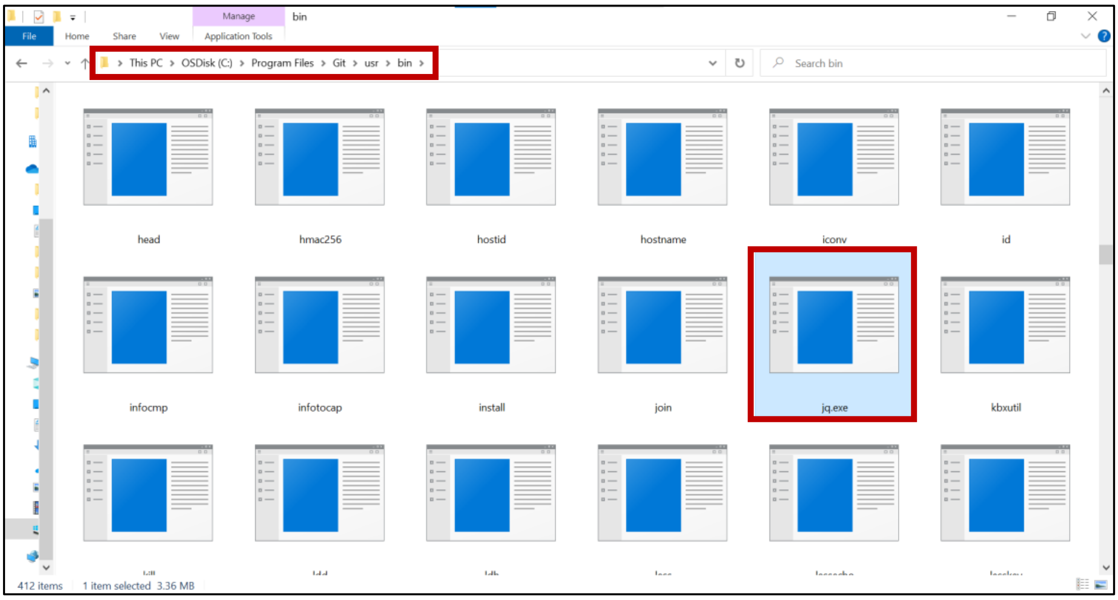
jq is like `sed` for JSON data - you can use it to slice and filter and map and transform structured data with the same ease that `sed`, `awk`, `grep` and friends let you play with text.

jq is written in portable C, and it has zero runtime dependencies. You can download a single binary, `scp` it to a far away machine of the same type, and expect it to work.

jq can mangle the data format that you have into the one that you want with very little effort, and the program to do so is often shorter and simpler than you'd expect.

https://github.com/stedolan/jq/releases/download/jq-1.6/jq-win64.exe

- rename it as "jq.exe" and put it to C:\Program Files\Git\usr\bin



2. Run the automation code

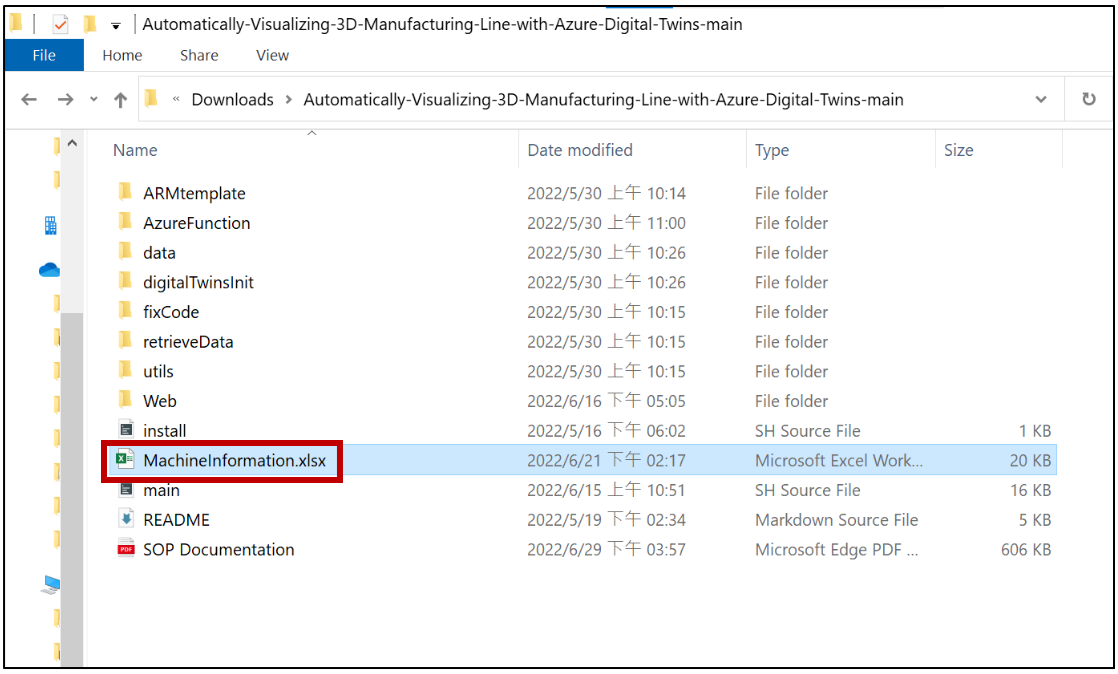
After installing all the software required, we can run the automation code.

The following are all steps to run it.

step
1. Fill device information in “MachineInformation.xlsx”
2. Put device 3D model file under specific folder
3. Open the source code folder in Git bash
4. Run command “bash install.sh” in Git Bash for the first time
5. Run command “bash main.sh” in Git Bash

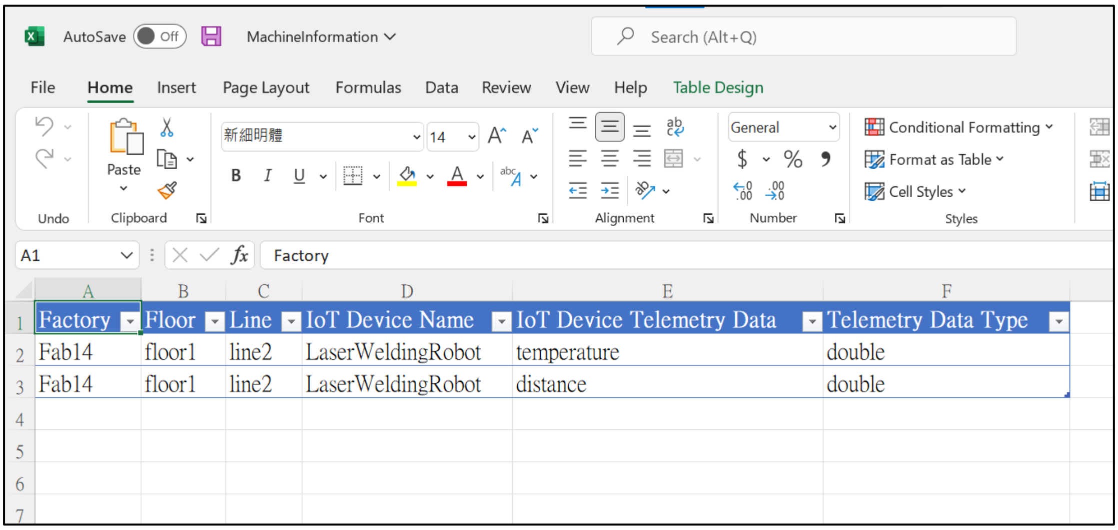
1. Fill device information in “MachineInformation.xlsx”

- Go to source code folder \Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins-main and find “MachineInfomation.xlsx”


Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins-main
File
Home
Share
View
Downloads › Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins-main

Name	Date modified	Type	Size
ARMtemplate	2022/5/30 上午 10:14	File folder	
AzureFunction	2022/5/30 上午 11:00	File folder	
data	2022/5/30 上午 10:26	File folder	
digitalTwinsInit	2022/5/30 上午 10:26	File folder	
fixCode	2022/5/30 上午 10:15	File folder	
retrieveData	2022/5/30 上午 10:15	File folder	
utils	2022/5/30 上午 10:15	File folder	
Web	2022/6/16 下午 05:05	File folder	
install	2022/5/16 下午 06:02	SH Source File	1 KB
MachineInformation.xlsx	2022/6/21 下午 02:17	Microsoft Excel Work...	20 KB
main	2022/6/15 上午 10:51	SH Source File	16 KB
README	2022/5/19 下午 02:34	Markdown Source File	5 KB
SOP Documentation	2022/6/29 下午 03:57	Microsoft Edge PDF ...	606 KB

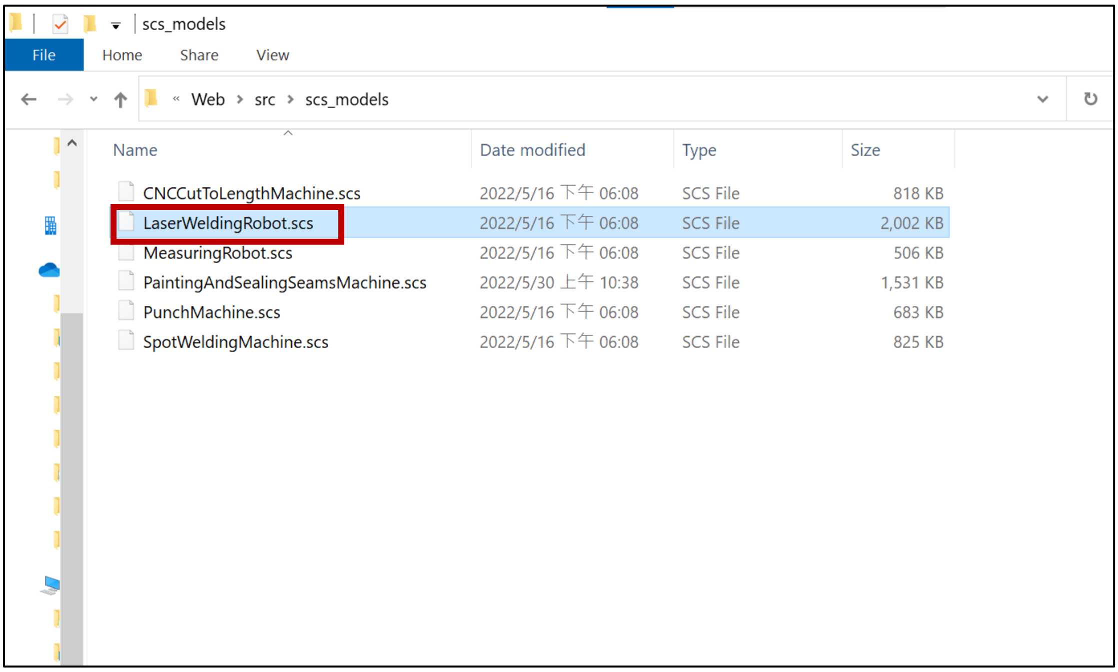
- Open it and fill the device information (such as name, telemetry data, etc.) in it.



Factory	Floor	Line	IoT Device Name	IoT Device Telemetry Data	Telemetry Data Type
Fab14	floor1	line2	LaserWeldingRobot	temperature	double
Fab14	floor1	line2	LaserWeldingRobot	distance	double

2. Put device 3D model file under specific folder

- Put device 3D model "LaserWeldingRobot.scs" under \Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins-main\Web\src\scs_models



Name	Date modified	Type	Size
CNCCutToLengthMachine.scs	2022/5/16 下午 06:08	SCS File	818 KB
LaserWeldingRobot.scs	2022/5/16 下午 06:08	SCS File	2,002 KB
MeasuringRobot.scs	2022/5/16 下午 06:08	SCS File	506 KB
PaintingAndSealingSeamsMachine.scs	2022/5/30 上午 10:38	SCS File	1,531 KB
PunchMachine.scs	2022/5/16 下午 06:08	SCS File	683 KB
SpotWeldingMachine.scs	2022/5/16 下午 06:08	SCS File	825 KB

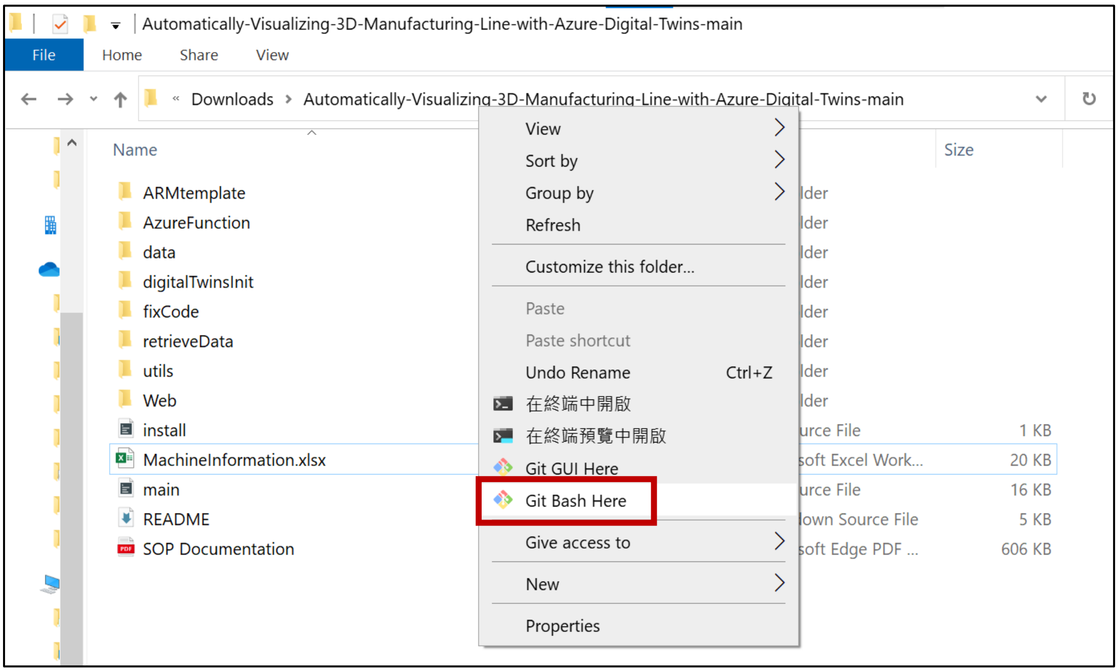
＊ File name should be the same as name filled in "MachineInformation.xlsx"

＊ Only .scs file allowed

＊ If you have a 3D model file with other file extension, please check Appendix in this user guide for file extension conversion information.

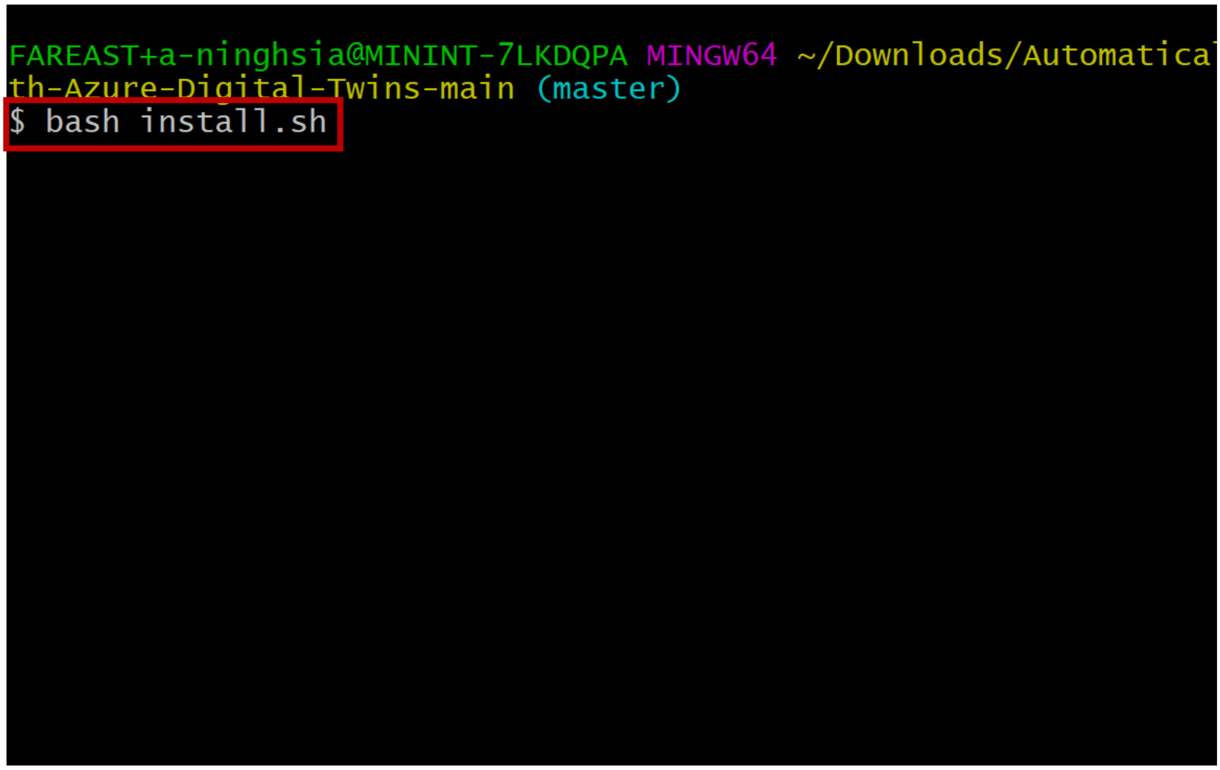
3. Open the source code folder in Git bash

- Go to source code folder \Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins-main, right click on the blank area and click "Git Bash Here"



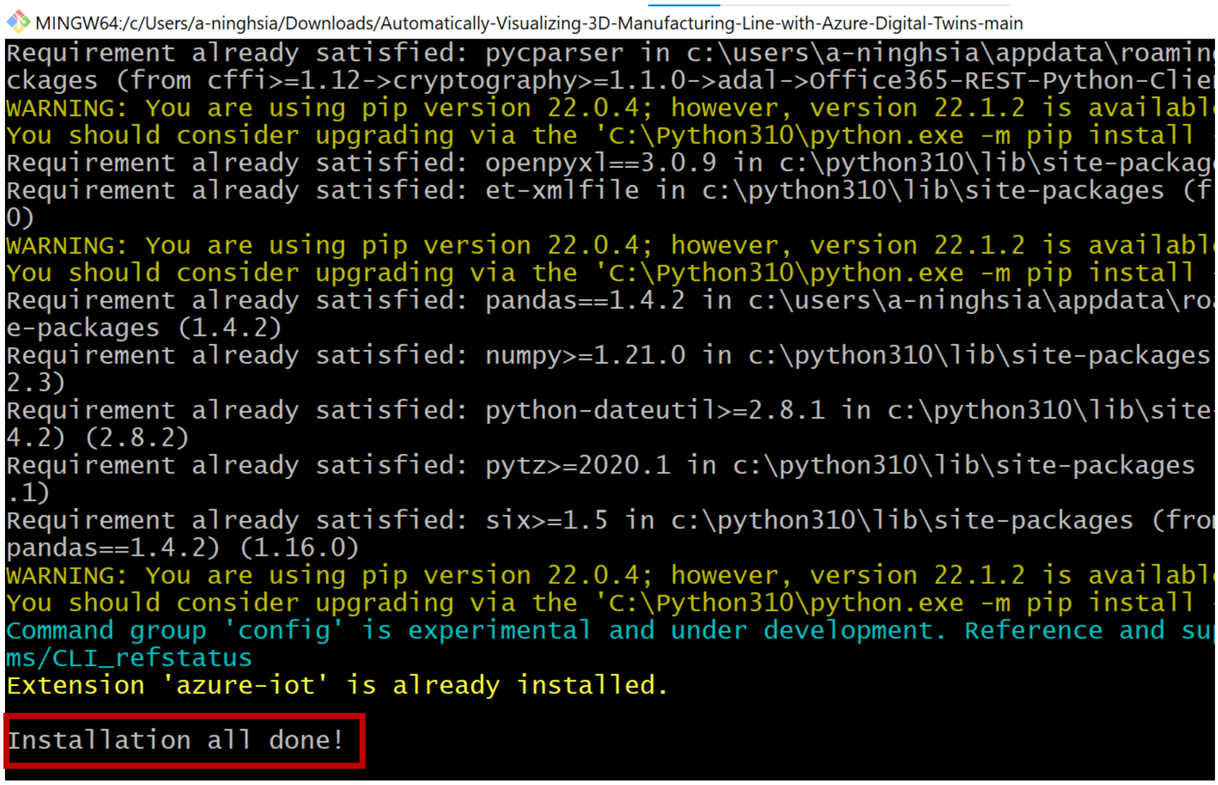
4. Run command “bash install.sh” in Git Bash for the first time

- For the first time, type in “bash install.sh” in Git Bash and then press enter to run the command.



```
FAREAST+a-ninghsia@MININT-7LKDQPA MINGW64 ~/Downloads/Automatica
th-Azure-Digital-Twins-main (master)
$ bash install.sh
```

- You’ll see “Installation all done!” when finished.

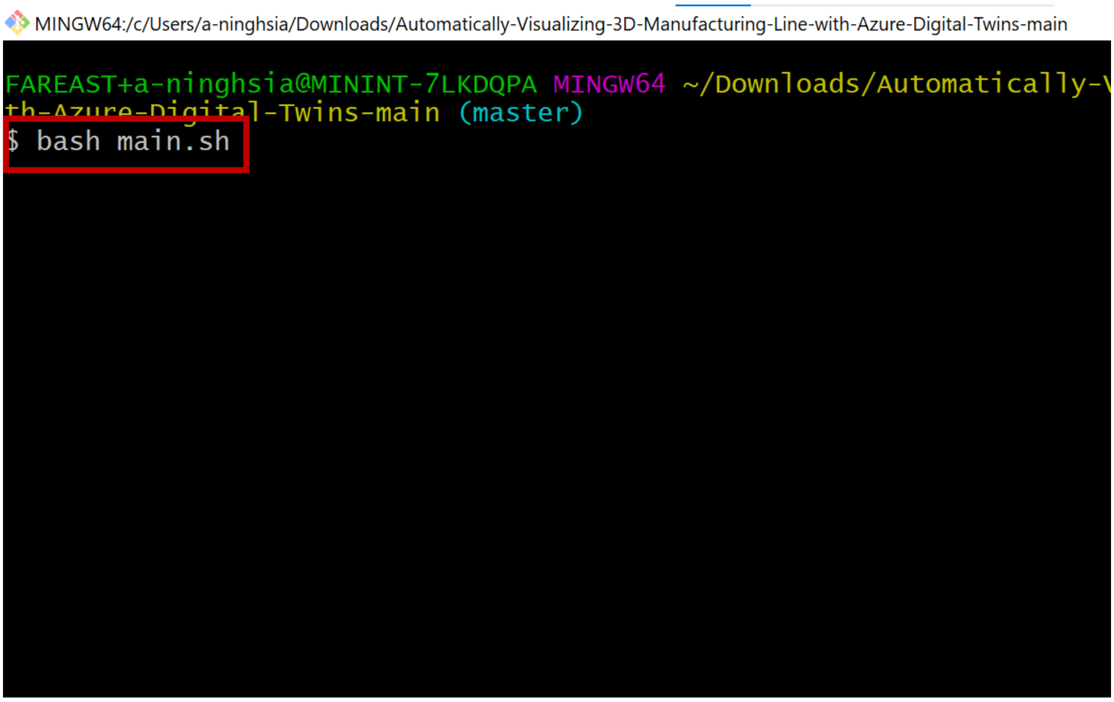


MINGW64:/c/Users/a-ninghsia/Downloads/Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins-main

```
Requirement already satisfied: pycparser in c:\users\a-ninghsia\appdata\roamin
ckages (from cffi>=1.12->cryptography>=1.1.0->adal->Office365-REST-Python-Clie
WARNING: You are using pip version 22.0.4; however, version 22.1.2 is availabl
You should consider upgrading via the 'C:\Python310\python.exe -m pip install 
Requirement already satisfied: openpyxl==3.0.9 in c:\python310\lib\site-packag
Requirement already satisfied: et-xmlfile in c:\python310\lib\site-packages (f
0)
WARNING: You are using pip version 22.0.4; however, version 22.1.2 is availabl
You should consider upgrading via the 'C:\Python310\python.exe -m pip install 
Requirement already satisfied: pandas==1.4.2 in c:\users\a-ninghsia\appdata\ro
e-packages (1.4.2)
Requirement already satisfied: numpy>=1.21.0 in c:\python310\lib\site-packages
2.3)
Requirement already satisfied: python-dateutil>=2.8.1 in c:\python310\lib\site
4.2) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\python310\lib\site-packages 
.1)
Requirement already satisfied: six>=1.5 in c:\python310\lib\site-packages (fro
pandas==1.4.2) (1.16.0)
WARNING: You are using pip version 22.0.4; however, version 22.1.2 is availabl
You should consider upgrading via the 'C:\Python310\python.exe -m pip install 
Command group 'config' is experimental and under development. Reference and su
ms/CLI_refstatus
Extension 'azure-iot' is already installed.

Installation all done!
```

5. Run command “bash main.sh” in Git Bash

- To run the automation code, type in “bash main.sh” in Git Bash and then press enter to run the command.

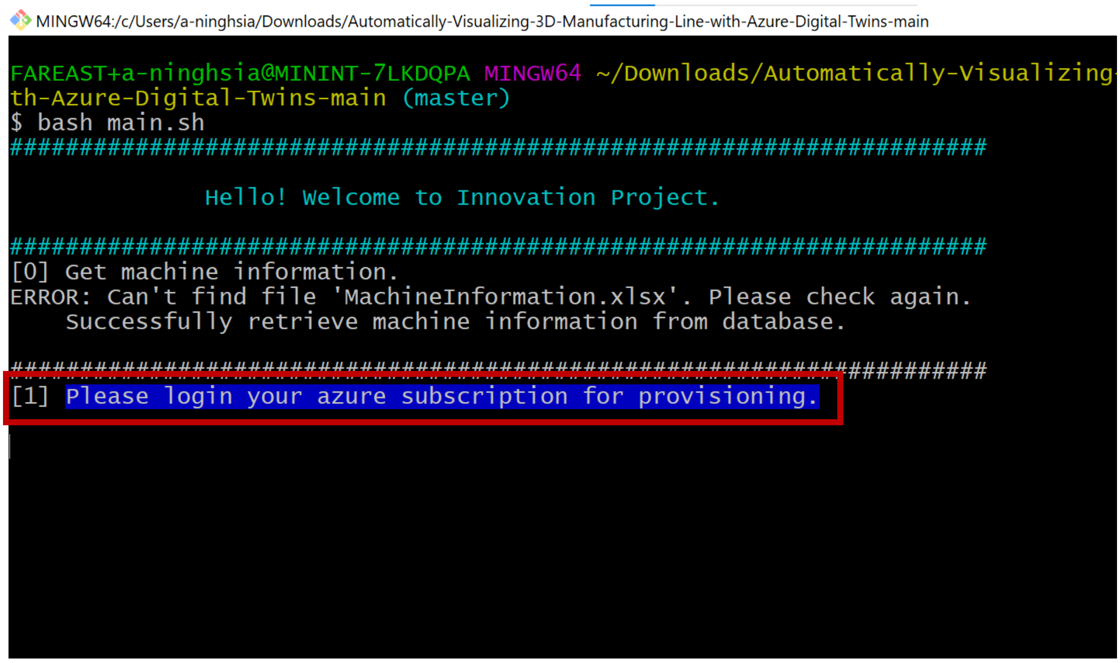


MINGW64:/c/Users/a-ninghsia/Downloads/Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins-main

```
FAREAST+a-ninghsia@MININT-7LKDQPA MINGW64 ~/Downloads/Automatically-V
th-Azure-Digital-Twins-main (master)
$ bash main.sh
```

- There would be several actions and choices you have to make during provision:

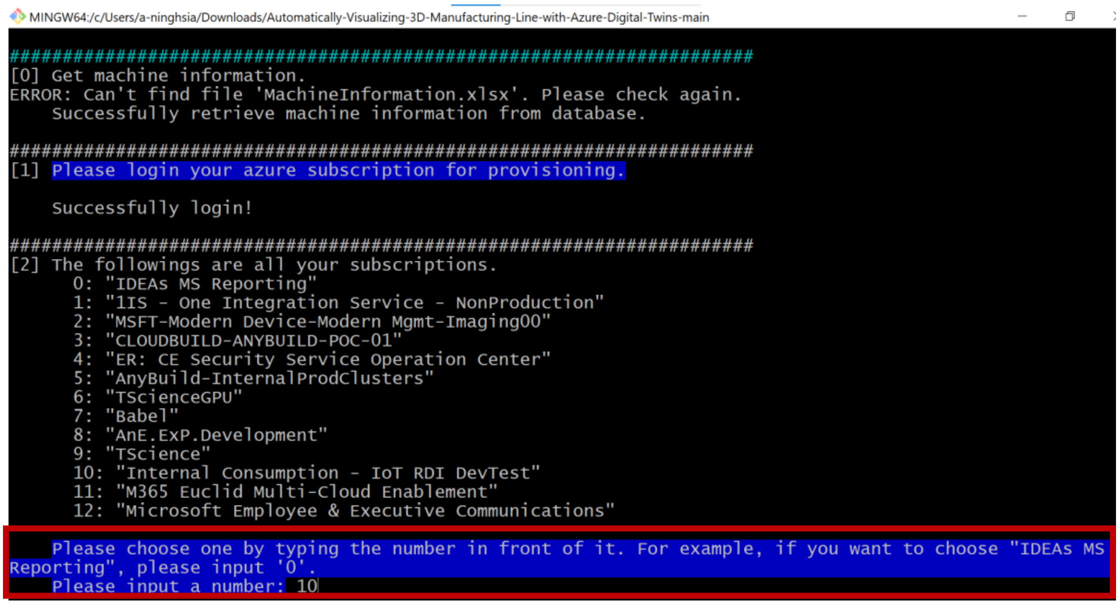
  - In section [1], please login your Azure account in the pop out webpage.



MINGW64:/c/Users/a-ninghsia/Downloads/Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins-main

```
FAREAST+a-ninghsia@MININT-7LKDQPA MINGW64 ~/Downloads/Automatically-Visualizing
th-Azure-Digital-Twins-main (master)
$ bash main.sh
######################################################################

              Hello! Welcome to Innovation Project.

######################################################################
[0] Get machine information.
ERROR: Can't find file 'MachineInformation.xlsx'. Please check again.
    Successfully retrieve machine information from database.

######################################################################
[1] Please login your azure subscription for provisioning.
```

  - In section [2], please choose an Azure subscription by inputting the number in front of it.



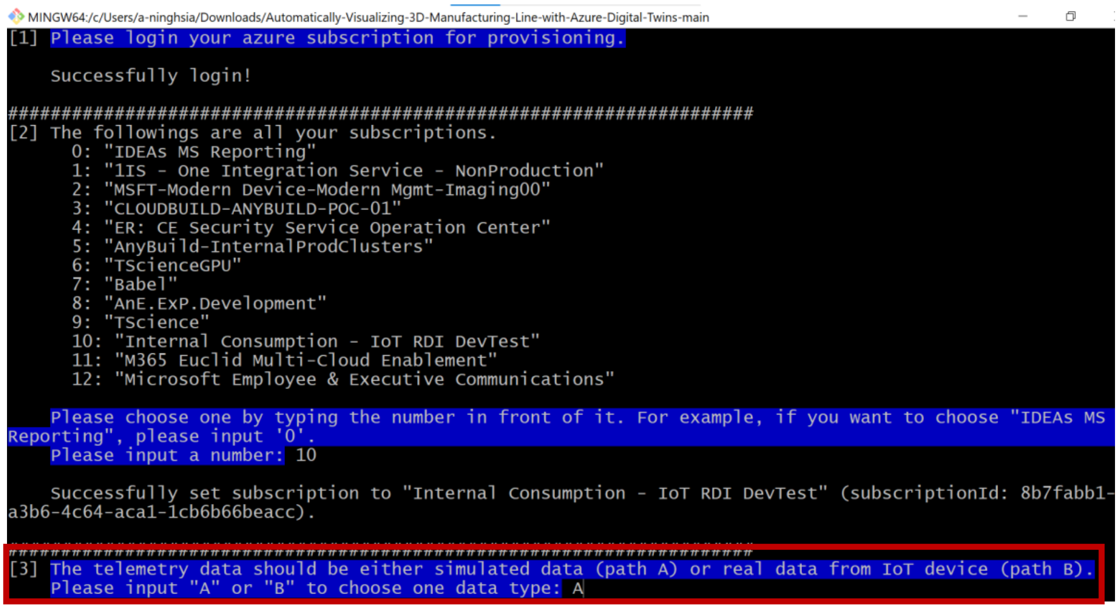
MINGW64:/c/Users/a-ninghsia/Downloads/Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins-main

```
######################################################################
[0] Get machine information.
ERROR: Can't find file 'MachineInformation.xlsx'. Please check again.
    Successfully retrieve machine information from database.

######################################################################
[1] Please login your azure subscription for provisioning.

    Successfully login!

######################################################################
[2] The followings are all your subscriptions.
      0: "IDEAs MS Reporting"
      1: "1IS - One Integration Service - NonProduction"
      2: "MSFT-Modern Device-Modern Mgmt-Imaging00"
      3: "CLOUDBUILD-ANYBUILD-POC-01"
      4: "ER: CE Security Service Operation Center"
      5: "AnyBuild-InternalProdClusters"
      6: "TScienceGPU"
      7: "Babel"
      8: "AnE.ExP.Development"
      9: "TScience"
      10: "Internal Consumption - IoT RDI DevTest"
      11: "M365 Euclid Multi-Cloud Enablement"
      12: "Microsoft Employee & Executive Communications"

    Please choose one by typing the number in front of it. For example, if you want to choose "IDEAs MS
Reporting", please input '0'.
    Please input a number: 10
```

- In section [3], please choose a data type by inputting "A" or "B":
    - "A": If you do NOT have a device which is already connected to Azure IoT Hub, you can choose "A" and our automation code will simulate some data for you according to your telemetry data type.
    - "B": If you already have a device which is connected to Azure IoT Hub, you can choose "B" and our automation code will automatically connect your device telemetry data in IoT Hub to Digital Twins.

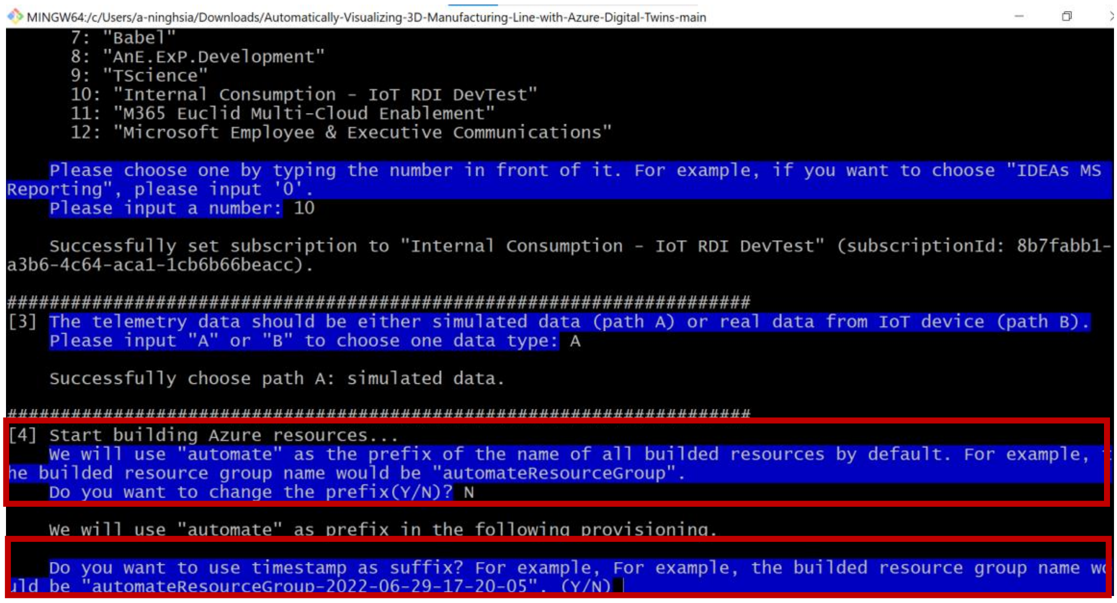
＊ If you choose "B", you should ensure that only ONE device is connected to the IoT Hub, or you may have to manually set a filter in Event Grid Subscriptions. Please check Appendix in this user guide for more information.

＊ Noted that in path "B", since the telemetry data format from IoT Hub may differ among different devices, you may need to manually fix the data format transformation code for function app and then manually publish it to function app again, if your data format is different from our default case: Siemens S7-1500 + MOXA MC1121. Please check Appendix in this user guide for more information.


MINGW64:/c/Users/a-ninghsia/Downloads/Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins-main

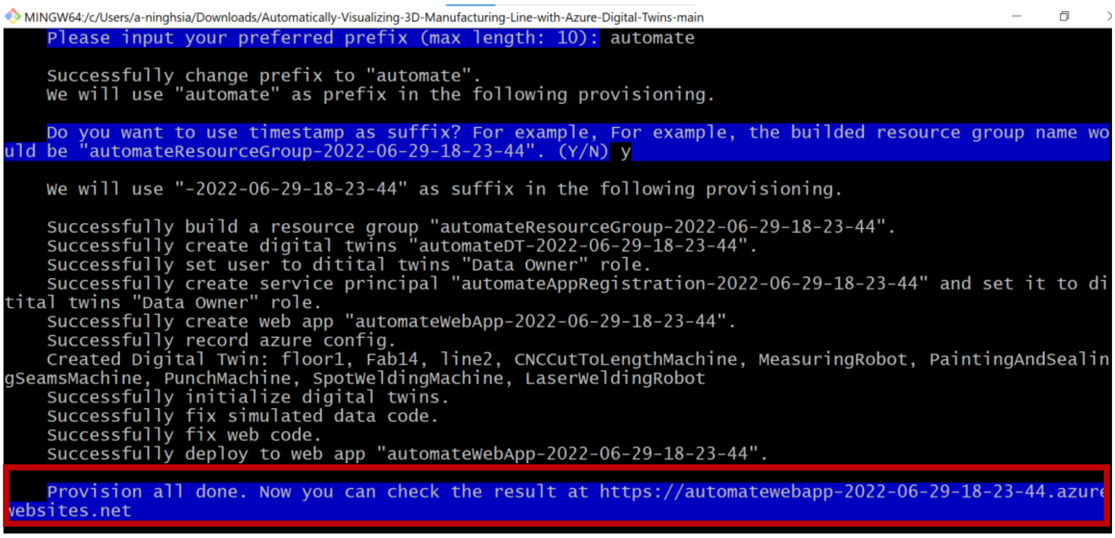
```
[1] Please login your azure subscription for provisioning.

    Successfully login!

######################################################################
[2] The followings are all your subscriptions.
      0: "IDEAs MS Reporting"
      1: "1IS - One Integration Service - NonProduction"
      2: "MSFT-Modern Device-Modern Mgmt-Imaging00"
      3: "CLOUDBUILD-ANYBUILD-POC-01"
      4: "ER: CE Security Service Operation Center"
      5: "AnyBuild-InternalProdClusters"
      6: "TScienceGPU"
      7: "Babel"
      8: "AnE.ExP.Development"
      9: "TScience"
      10: "Internal Consumption - IoT RDI DevTest"
      11: "M365 Euclid Multi-Cloud Enablement"
      12: "Microsoft Employee & Executive Communications"

    Please choose one by typing the number in front of it. For example, if you want to choose "IDEAs MS
Reporting", please input '0'.
    Please input a number: 10

    Successfully set subscription to "Internal Consumption - IoT RDI DevTest" (subscriptionId: 8b7fabb1-
a3b6-4c64-aca1-1cb6b66beacc).

######################################################################
[3] The telemetry data should be either simulated data (path A) or real data from IoT device (path B).
    Please input "A" or "B" to choose one data type: A
```

- In section [4], please input “Y” or “N” to decide whether you would like to change the default resource prefix “automate” and use timestamp as suffix.


MINGW64:/c/Users/a-ninghsia/Downloads/Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins-main

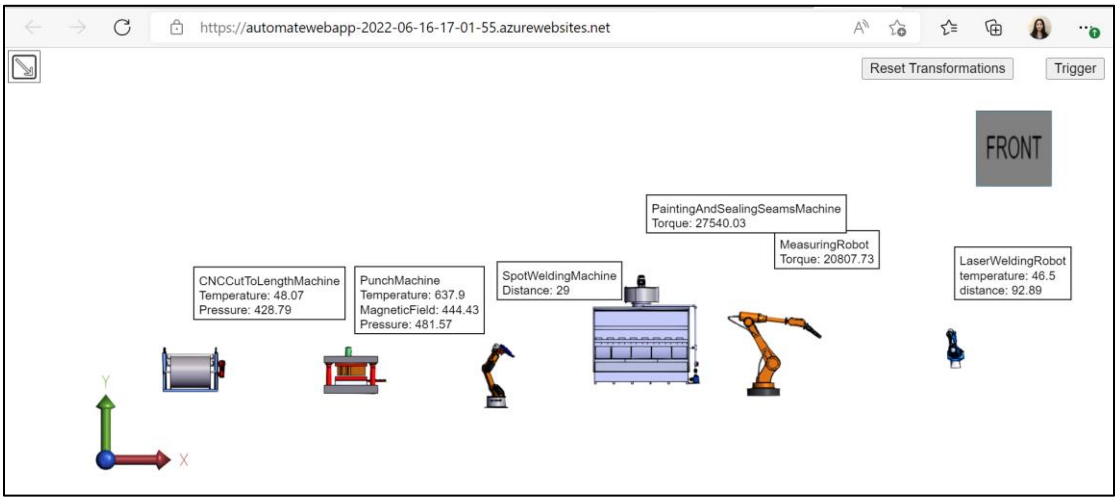
```
      7: "Babel"
      8: "AnE.ExP.Development"
      9: "TScience"
      10: "Internal Consumption - IoT RDI DevTest"
      11: "M365 Euclid Multi-Cloud Enablement"
      12: "Microsoft Employee & Executive Communications"

    Please choose one by typing the number in front of it. For example, if you want to choose "IDEAs MS
Reporting", please input '0'.
    Please input a number: 10

    Successfully set subscription to "Internal Consumption - IoT RDI DevTest" (subscriptionId: 8b7fabb1-
a3b6-4c64-aca1-1cb6b66beacc).

######################################################################
[3] The telemetry data should be either simulated data (path A) or real data from IoT device (path B).
    Please input "A" or "B" to choose one data type: A

    Successfully choose path A: simulated data.

######################################################################
[4] Start building Azure resources...
    We will use "automate" as the prefix of the name of all builded resources by default. For example, t
he builded resource group name would be "automateResourceGroup".
    Do you want to change the prefix(Y/N)? N

    We will use "automate" as prefix in the following provisioning.

    Do you want to use timestamp as suffix? For example, For example, the builded resource group name wo
uld be "automateResourceGroup-2022-06-29-17-20-05". (Y/N)
```

- The automation code would start the provision, building resources such as resource group, digital twins, app registration and web application. After that, it’ ll return an URL to you, where you can see your 3D factory with 3D device models and real-time telemetry data.



MINGW64:/c/Users/a-ninghsia/Downloads/Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins-main

```
    Please input your preferred prefix (max length: 10): automate

    Successfully change prefix to "automate".
    We will use "automate" as prefix in the following provisioning.

    Do you want to use timestamp as suffix? For example, For example, the builded resource group name wo
uld be "automateResourceGroup-2022-06-29-18-23-44". (Y/N) y

    We will use "-2022-06-29-18-23-44" as suffix in the following provisioning.

    Successfully build a resource group "automateResourceGroup-2022-06-29-18-23-44".
    Successfully create digital twins "automateDT-2022-06-29-18-23-44".
    Successfully set user to ditital twins "Data Owner" role.
    Successfully create service principal "automateAppRegistration-2022-06-29-18-23-44" and set it to di
tital twins "Data Owner" role.
    Successfully create web app "automateWebApp-2022-06-29-18-23-44".
    Successfully record azure config.
    Created Digital Twin: floor1, Fab14, line2, CNCCutToLengthMachine, MeasuringRobot, PaintingAndSealin
gSeamsMachine, PunchMachine, SpotWeldingMachine, LaserWeldingRobot
    Successfully initialize digital twins.
    Successfully fix simulated data code.
    Successfully fix web code.
    Successfully deploy to web app "automateWebApp-2022-06-29-18-23-44".

    Provision all done. Now you can check the result at https://automatewebapp-2022-06-29-18-23-44.azure
websites.net
```



Appendix

1. Connect to IoT Hub

If you haven' t connected your device to IoT Hub, here' re some reference links for you.

- Ingest IoT Hub telemetry into Azure Digital Twins:
    - ◆ https://docs.microsoft.com/en-us/azure/digital-twins/how-to-

ingest-iot-hub-data

- Ingesting OPC UA data with Azure Digital Twins:
    - ◆ https://docs.microsoft.com/en-us/azure/digital-twins/how-to-ingest-opcua-data

2. 3D Model file extension conversion

1. Go https://manage.techsoft3d.com/ and sign up an account.
2. Download HOOPS Web Platform
3. Converter.exe located in

   "HOOPS_Web_Platform\authoring\converter\bin\win64"
4. Command line example:
    - `converter.exe --input ./raw/test.stp  --license "license" --output_scs ./raw/test.scs`
    - documentation:

      https://docs.techsoft3d.com/communicator/latest/build/prog_guide/data_import/cad_conversion/converter_app/converter-application-overview.html

3. Filter in Event Grid Subscription

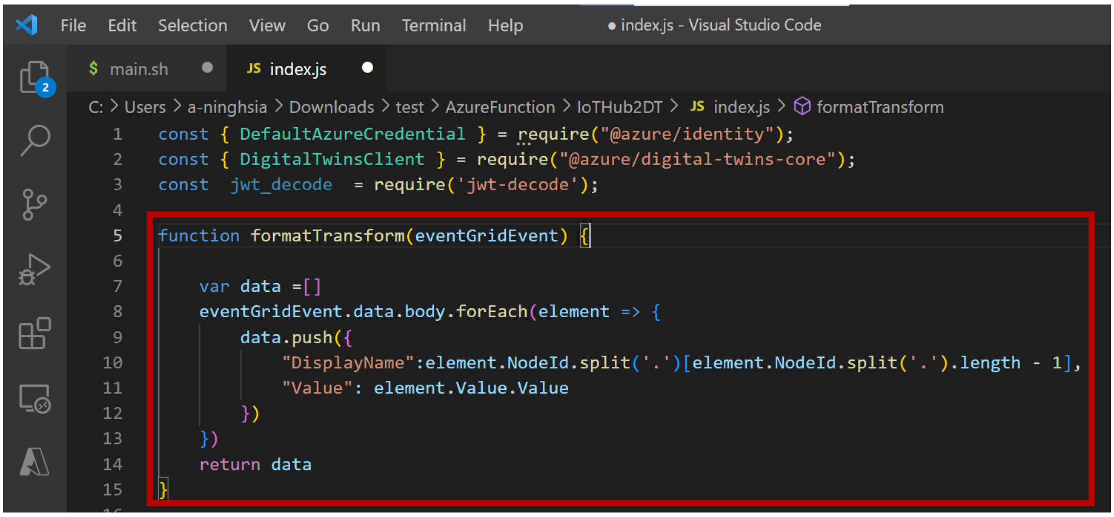
If there’ re more than one device connected to the same IoT Hub, you need to manually set a filter in Event Grid Subscription to filter out unwanted data. The following link is a documentation about filtering events for Event Grid provided by Microsoft.

https://docs.microsoft.com/en-us/azure/event-grid/how-to-filter-events

4. If your device is different from our default case

If your device is different from our default case: Siemens S7-1500 + MOXA MC1121, you may need to you may need to manually fix the data format transformation code for function app at Automatically-Visualizing-3D-Manufacturing-Line-with-Azure-Digital-Twins-main\AzureFuction\IoTHub2DT\index.js function formatTransform() and then manually publish it to function app again

- Function formatTransform() with default case:



File
Edit
Selection
View
Go
Run
Terminal
Help
● index.js - Visual Studio Code
main.sh
index.js
C: > Users > a-ninghsia > Downloads > test > AzureFunction > IoTHub2DT > index.js > formatTransform

```javascript
const { DefaultAzureCredential } = require("@azure/identity");
const { DigitalTwinsClient } = require("@azure/digital-twins-core");
const  jwt_decode  = require('jwt-decode');

function formatTransform(eventGridEvent) {

    var data =[]
    eventGridEvent.data.body.forEach(element => {
        data.push({
            "DisplayName":element.NodeId.split('.')[element.NodeId.split('.').length - 1],
            "Value": element.Value.Value
        })
    })
    return data
}
```

- Reference Links for deploying functions to Azure function:

  - ◆ Using Visual Studio Code:

    - https://docs.microsoft.com/en-us/azure/azure-functions/functions-develop-vs-code?tabs=csharp

  - ◆ Zip deployment for Azure Functions

- https://docs.microsoft.com/en-us/azure/azure-functions/functions-develop-vs-code?tabs=csharp

5. Other reference links

- Step-by-step tutorial for creating an Azure Digital Twins instance in Azure Portal:
  - ◆ https://docs.microsoft.com/en-us/azure/digital-twins/quickstart-azure-digital-twins-explorer
- DTDL model brief introduction:
  - ◆ https://docs.microsoft.com/en-us/azure/digital-twins/concepts-models
- Coding with the Azure Digital Twins SDK:
  - ◆ https://docs.microsoft.com/en-us/azure/digital-twins/tutorial-code