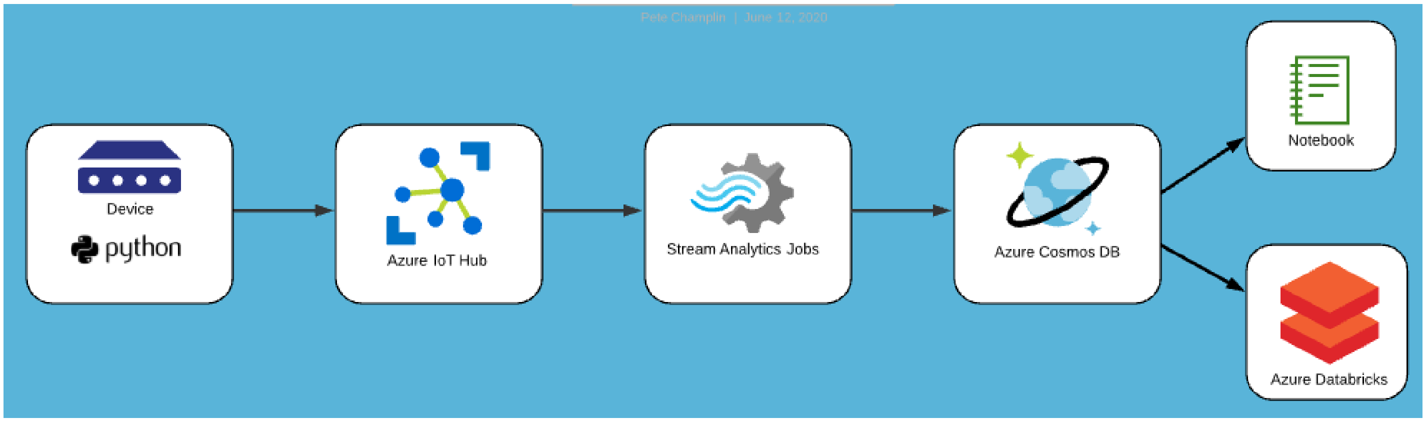
Pete Champlin

Big Data 230A

Week 8 Assignment (Alternate)

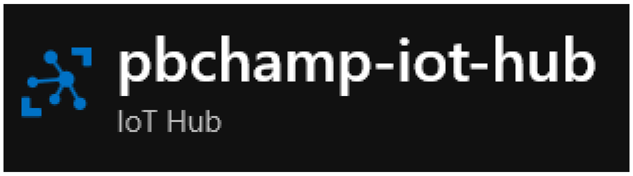
Using my own Azure subscription I followed this documentation - https://docs.microsoft.com/en-us/azure/iot-hub/quickstart-send-telemetry-python - to create a simulated IoT device (using Python), consume the output through Azure IoT Hub and a Stream Analytics job, and write the data to Cosmos DB. I was able to query the data in Data Explorer, an Azure Notebook, and a Databricks (Community Edition) Notebook.

Here is the basic data flow:



Below are the steps followed:

- Create an Azure IoT Hub:

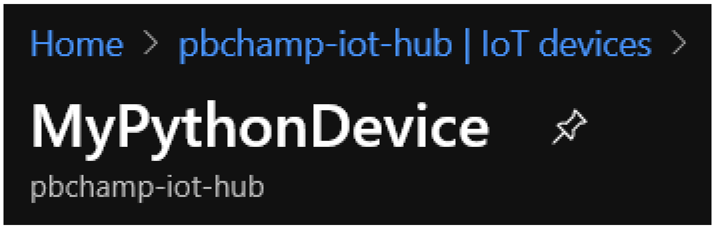


pbchamp-iot-hub

IoT Hub

- Associate simulated (Python) IoT device to IoT Hub using Azure CLI

az iot hub device-identity create --hub-name pbchamp-iot-hub --device-id MyPythonDevice

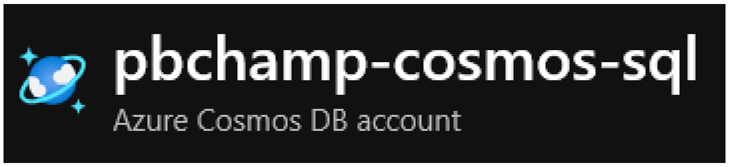


Home > pbchamp-iot-hub | IoT devices >

MyPythonDevice

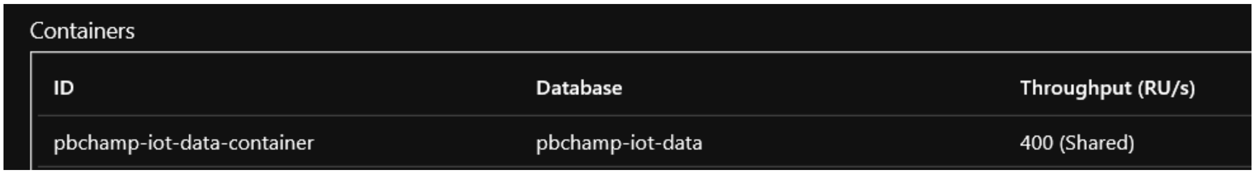
pbchamp-iot-hub

- Create a Cosmos DB (SQL) account, container, and database



pbchamp-cosmos-sql

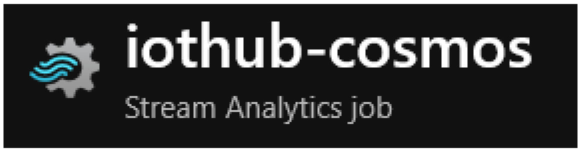
Azure Cosmos DB account



Containers

ID	Database	Throughput (RU/s)
pbchamp-iot-data-container	pbchamp-iot-data	400 (Shared)

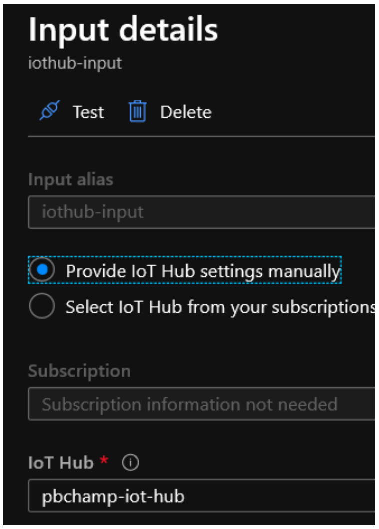
- Create a Stream Analytics job



iothub-cosmos

Stream Analytics job

- Create a Stream Analytics Input (Iot Hub)



Input details

iothub-input

Input alias: iothub-input

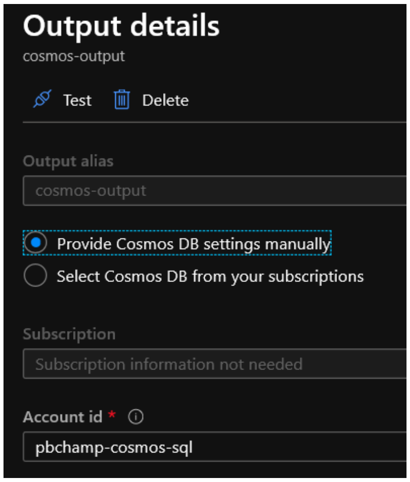
● Provide IoT Hub settings manually

○ Select IoT Hub from your subscriptions

Subscription: Subscription information not needed

IoT Hub *: pbchamp-iot-hub

- Create a Stream Analytics Output (Cosmos DB)



Output details

cosmos-output

Output alias: cosmos-output

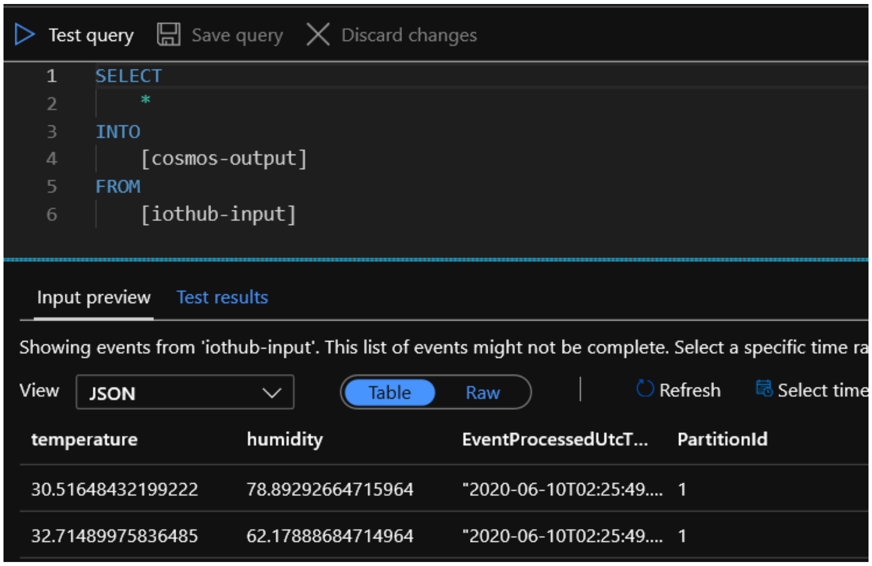
● Provide Cosmos DB settings manually

○ Select Cosmos DB from your subscriptions

Subscription: Subscription information not needed

Account id *: pbchamp-cosmos-sql

- Create a Stream Analytics Query (Python device is running)
    - Note: Square brackets needed around input and output since they include a dash



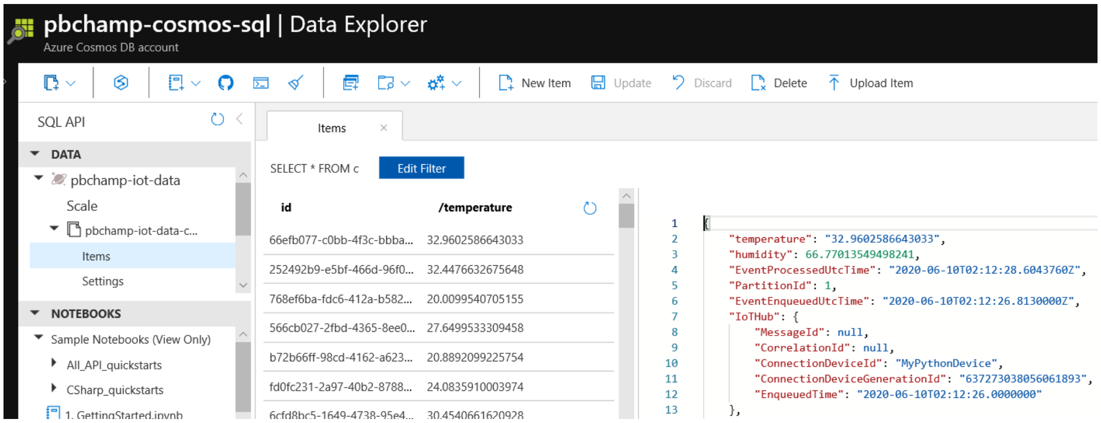
```
SELECT
    *
INTO
    [cosmos-output]
FROM
    [iothub-input]
```

Showing events from 'iothub-input'. This list of events might not be complete. Select a specific time ra

View JSON

temperature	humidity	EventProcessedUtcT...	PartitionId
30.51648432199222	78.89292664715964	"2020-06-10T02:25:49....	1
32.71489975836485	62.17888684714964	"2020-06-10T02:25:49....	1

- Query Cosmos DB using Data Explorer

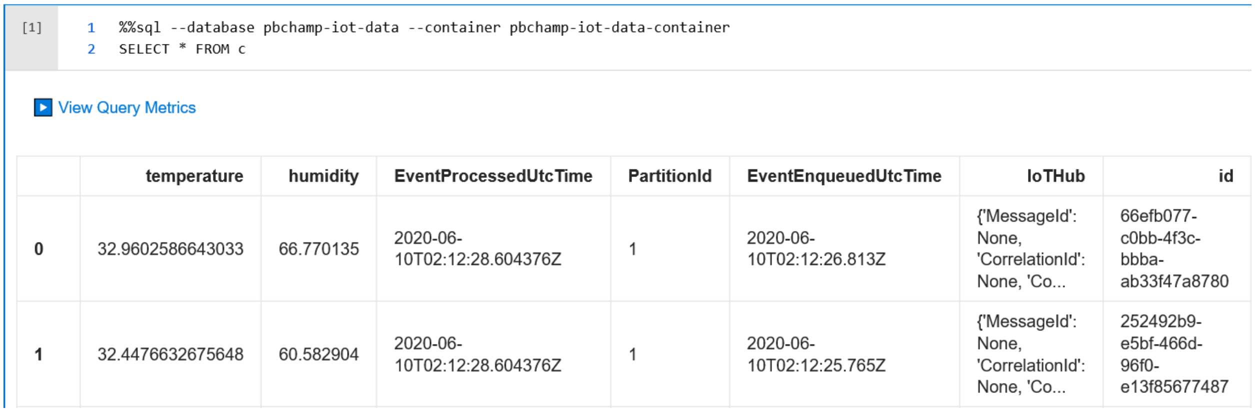


SELECT * FROM c

id	/temperature
66efb077-c0bb-4f3c-bbba...	32.9602586643033
252492b9-e5bf-466d-96f0...	32.4476632675648
768ef6ba-fdc6-412a-b582...	20.0099540705155
566cb027-2fbd-4365-8ee0...	27.6499533309458
b72b66ff-98cd-4162-a623...	20.8892099225754
fd0fc231-2a97-40b2-8788...	24.0835910003974
6cfd8bc5-1649-4738-95e4...	30.4540661620928

```
{
    "temperature": "32.9602586643033",
    "humidity": 66.77013549498241,
    "EventProcessedUtcTime": "2020-06-10T02:12:28.6043760Z",
    "PartitionId": 1,
    "EventEnqueuedUtcTime": "2020-06-10T02:12:26.8130000Z",
    "IoTHub": {
        "MessageId": null,
        "CorrelationId": null,
        "ConnectionDeviceId": "MyPythonDevice",
        "ConnectionDeviceGenerationId": "637273038056061893",
        "EnqueuedTime": "2020-06-10T02:12:26.0000000"
    },
```

- Query Cosmos DB using Azure Notebook

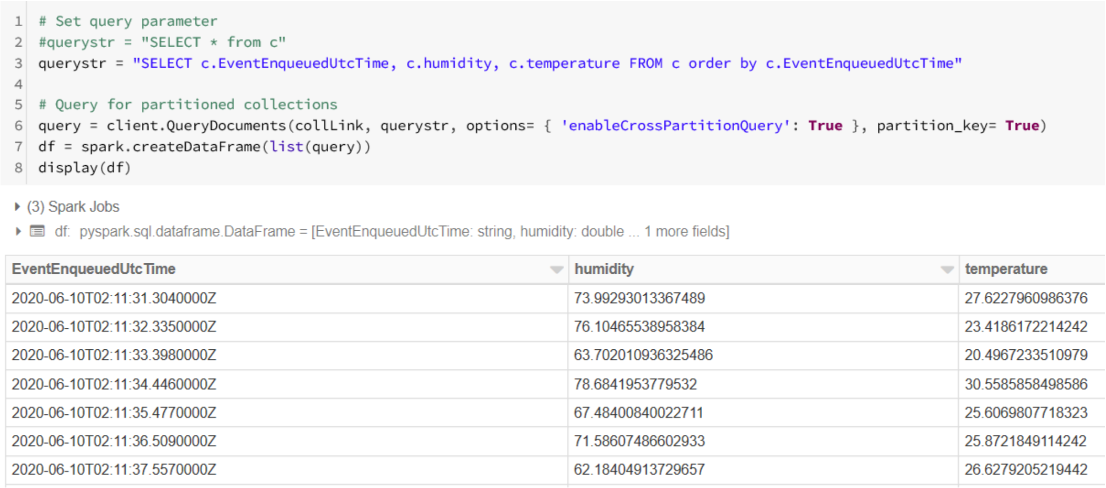


```
%%sql --database pbchamp-iot-data --container pbchamp-iot-data-container
SELECT * FROM c
```

	temperature	humidity	EventProcessedUtcTime	PartitionId	EventEnqueuedUtcTime	IoTHub	id
0	32.9602586643033	66.770135	2020-06-10T02:12:28.604376Z	1	2020-06-10T02:12:26.813Z	{'MessageId': None, 'CorrelationId': None, 'Co...	66efb077-c0bb-4f3c-bbba-ab33f47a8780
1	32.4476632675648	60.582904	2020-06-10T02:12:28.604376Z	1	2020-06-10T02:12:25.765Z	{'MessageId': None, 'CorrelationId': None, 'Co...	252492b9-e5bf-466d-96f0-e13f85677487

Obviously, much more complicated queries could be performed: https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-stream-analytics-query-patterns

- Query Cosmos DB using a Databricks/Spark (Community Edition) Notebook
    - Followed instructions here to set up install and configure the pyDocumentDB library (https://medium.com/big-data-and-cloud-a-z/connecting-databricks-to-cosmos-db-and-accessing-data-in-it-a1e1b9263944)



```
# Set query parameter
#querystr = "SELECT * from c"
querystr = "SELECT c.EventEnqueuedUtcTime, c.humidity, c.temperature FROM c order by c.EventEnqueuedUtcTime"

# Query for partitioned collections
query = client.QueryDocuments(collLink, querystr, options= { 'enableCrossPartitionQuery': True }, partition_key= True)
df = spark.createDataFrame(list(query))
display(df)
```

```
(3) Spark Jobs
df:  pyspark.sql.dataframe.DataFrame = [EventEnqueuedUtcTime: string, humidity: double ... 1 more fields]
```

EventEnqueuedUtcTime	humidity	temperature
2020-06-10T02:11:31.3040000Z	73.99293013367489	27.6227960986376
2020-06-10T02:11:32.3350000Z	76.10465538958384	23.4186172214242
2020-06-10T02:11:33.3980000Z	63.702010936325486	20.4967233510979
2020-06-10T02:11:34.4460000Z	78.6841953779532	30.5585858498586
2020-06-10T02:11:35.4770000Z	67.48400840022711	25.6069807718323
2020-06-10T02:11:36.5090000Z	71.58607486602933	25.8721849114242
2020-06-10T02:11:37.5570000Z	62.18404913729657	26.6279205219442