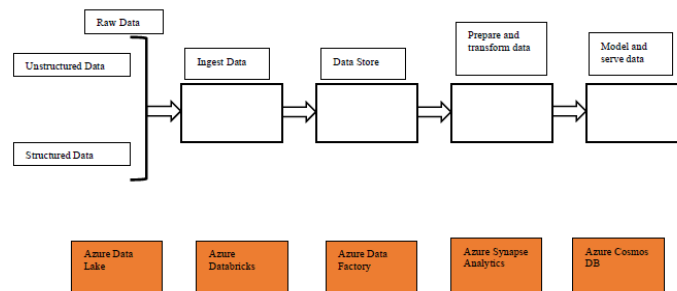


1. [Marks: 5] Explain below the 5 components shown in orange boxes. Explain which Azure components you will use where in this big data architecture and why.



Azure Data Lake:

Scalable raw data storage used to store large volumes of structured and unstructured raw data and act as the first landing zone from a variety of resources.

Azure Data Factory:

Cloud-based data integration service for orchestrating and automating data movement and workflows.

Azure Synapse Analytics:

Data platform act as analytical data warehouse and used for storing transformed data and supporting large-scale querying and reporting

Azure Databricks:

Apache spark-based analytics platform optimized for Microsoft Azure provided collaborative environment for sorting, cleaning, joining and outputting the data.

Azure Cosmos DB:

Globally distributed, multi-model database service for real-time application and aim to serve processed data to end-user applications at low latency

In this big data architecture:

Ingest Data → *Azure Data Factory*: Ingest the unstructured and structured data for preparation of data storing

Data Store → *Azure Data Lake*: Store the data collected from Azure Data Factory

Prepare and Transform Data → *Azure Databricks*: From Azure Data Factory, process and transform the data for data analysis

Model and Serve Data → *Azure Synapse Analytics*: Provide limitless analytics service for large-scale dataset and combine with data warehousing as big data analysis and modeling service

2. [Marks: 5] Explain how Stream Analytics works in Azure. Mention at least two common use cases or applications for this service.

In Microsoft Azure, Azure Stream Analytics works as a real-time analytics service for processing data streams from various sources like real-time IoT device or applications. It digests high-velocity data from the data source and uses a SQL-like language to filter, aggregate, and detect patterns within the data stream. It can output the data to multiple data sinks like data lakes or Azure SQL database and enabling real-time insight and actions.

Two common use case for Azure Stream Analytics:

1. Real-time IoT device monitoring:
process real-time data from IoT devices or sensors. It can be used to tracking the temperature of sensors or set up alerts when any data is monitored
2. Social media sentiment monitoring and analysis:
It can monitor ingest real-time Twitter(X) or Instagram feeds for trending topics or public sentiment and provide visualizations

3. [Marks: 10] Deploy all the resources in Azure Portal. Implement a Stream Analytics job by using the Azure portal. See this for reference - <https://learn.microsoft.com/en-us/azure/streamanalytics/stream-analytics-quick-create-portal>

For query use below:

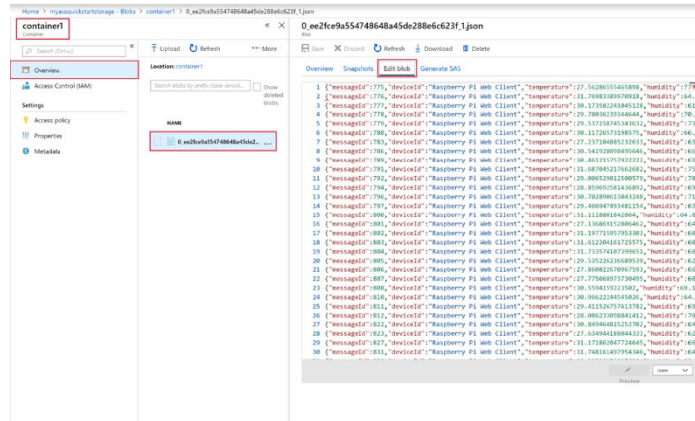
SELECT *

INTO BlobOutput

FROM IoTHubInput

HAVING Temperature > 25

See the below screenshot and show the top 30 results for your output.



Create stream analytics jobs, resource group, storage account, IOT Hub etc. as below

The screenshot shows the Azure Portal 'Recent' page. The table lists the following resources:

Name	Type	Location	Resource Group	Subscription	Last accessed
1628a5	Stream Analytics job	East US 2	DefaultResourceGroup-eastus2	Azure subscription 1	15 minutes ago
DefaultResourceGroup-eastus2	Resource group		DefaultResourceGroup-eastus2	Azure subscription 1	15 minutes ago
1628a5	Storage account		DefaultResourceGroup-eastus2	Azure subscription 1	18 minutes ago
1628a5	IoT Hub	East US	DefaultResourceGroup-eastus2	Azure subscription 1	21 minutes ago
ml-1628a5-yolov	Machine learning online endpoint	East US 2	zeying-zhou-rg	Azure subscription 1	8 hours ago
Azure subscription 1	Subscription			Azure subscription 1	19 hours ago

Query as below:

1628a5 | Query

Stream Analytics job

Start job

Open in VS Code

Diagnostics settings

Refresh

Query language docs

Share feedback

Tutorial

2 actions required

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Job topology

Inputs

Functions

Query

Outputs

No-code editor (preview)

Settings

Developer tools

Monitoring

Automation

Help

Inputs (2)

1628a5-1

Outputs (2)

1628a5-1

Functions (0)

Test query

Save query

Discard changes

```

2 Here are links to help you get started with Stream Analytics Query Language:
3 Common query patterns - https://go.microsoft.com/fwlink/?LinkID=619153
4 Query language - https://docs.microsoft.com/stream-analytics-query/query-language-elements-azure-stream-analytics
5 */
6 SELECT *
7 INTO BlobOutput
8 FROM IoTHubInput
9 HAVING Temperature > 25

```

Input preview

Test results

Job simulation (preview)

While sampling data, no data was received from 2 partitions.

Table

Raw

Refresh

Select time range

Upload sample input

Download sample data

sent query:

The screenshot shows the Raspberry Pi Online Simulator interface. On the left, a breadboard circuit is connected to a Raspberry Pi 4. The circuit includes a red LED, a 220Ω resistor, and a DS18B20 temperature sensor. The sensor is connected to the Pi's I2C pins (SDA and SCL). The code editor on the right contains the following Python script:

```

1 #
2 # Run Raspberry Pi Desktop - Floppystick Sample Date - Copyright (c) 2017 - licensed MIT
3 #
4 #
5 # led = gpiochip <leding on>
6 #
7 # i2c = i2cchip <gpiochip> <i2c device> <i2c device>
8 #
9 # i2caddr = i2cchip <i2c device> <i2c device> <i2c device>
10 #
11 # i2caddr = i2cchip <i2c device> <i2c device>
12 #
13 # i2caddr = i2cchip <i2c device> <i2c device>
14 #
15 # i2caddr = i2cchip <i2c device> <i2c device>
16 #
17 # i2caddr = i2cchip <i2c device> <i2c device>
18 #
19 # i2caddr = i2cchip <i2c device> <i2c device>
20 #
21 # i2caddr = i2cchip <i2c device> <i2c device>
22 #
23 # i2caddr = i2cchip <i2c device> <i2c device>
24 #
25 # i2caddr = i2cchip <i2c device> <i2c device>
26 #
27 # i2caddr = i2cchip <i2c device> <i2c device>
28 #
29 # i2caddr = i2cchip <i2c device> <i2c device>
30 #
31 # i2caddr = i2cchip <i2c device> <i2c device>
32 #
33 # i2caddr = i2cchip <i2c device> <i2c device>
34 #
35 # i2caddr = i2cchip <i2c device> <i2c device>
36 #
37 # i2caddr = i2cchip <i2c device> <i2c device>
38 #
39 # i2caddr = i2cchip <i2c device> <i2c device>
40 #
41 # i2caddr = i2cchip <i2c device> <i2c device>
42 #
43 # i2caddr = i2cchip <i2c device> <i2c device>
44 #
45 # i2caddr = i2cchip <i2c device> <i2c device>
46 #
47 # i2caddr = i2cchip <i2c device> <i2c device>
48 #
49 # i2caddr = i2cchip <i2c device> <i2c device>
50 #
51 # i2caddr = i2cchip <i2c device> <i2c device>
52 #
53 # i2caddr = i2cchip <i2c device> <i2c device>
54 #
55 # i2caddr = i2cchip <i2c device> <i2c device>
56 #
57 # i2caddr = i2cchip <i2c device> <i2c device>
58 #
59 # i2caddr = i2cchip <i2c device> <i2c device>
60 #
61 # i2caddr = i2cchip <i2c device> <i2c device>
62 #
63 # i2caddr = i2cchip <i2c device> <i2c device>
64 #
65 # i2caddr = i2cchip <i2c device> <i2c device>
66 #
67 # i2caddr = i2cchip <i2c device> <i2c device>
68 #
69 # i2caddr = i2cchip <i2c device> <i2c device>
70 #
71 # i2caddr = i2cchip <i2c device> <i2c device>
72 #
73 # i2caddr = i2cchip <i2c device> <i2c device>
74 #
75 # i2caddr = i2cchip <i2c device> <i2c device>
76 #
77 # i2caddr = i2cchip <i2c device> <i2c device>
78 #
79 # i2caddr = i2cchip <i2c device> <i2c device>
80 #
81 # i2caddr = i2cchip <i2c device> <i2c device>
82 #
83 # i2caddr = i2cchip <i2c device> <i2c device>
84 #
85 # i2caddr = i2cchip <i2c device> <i2c device>
86 #
87 # i2caddr = i2cchip <i2c device> <i2c device>
88 #
89 # i2caddr = i2cchip <i2c device> <i2c device>
90 #
91 # i2caddr = i2cchip <i2c device> <i2c device>
92 #
93 # i2caddr = i2cchip <i2c device> <i2c device>
94 #
95 # i2caddr = i2cchip <i2c device> <i2c device>
96 #
97 # i2caddr = i2cchip <i2c device> <i2c device>
98 #
99 # i2caddr = i2cchip <i2c device> <i2c device>
100 #

```

Top 30 Result similar to the provided screenshot:

Microsoft Azure Upgrade Search resources, services, and docs (G+) Copilot

Home > 1628a5 >

1628a5 Container

Search Upload Change access level

Overview Authentication method: Access key (Switch to Microsoft Entra user account) Location: 1628a5

Diagnose and solve problems Access Control (IAM) Settings

Search blobs by prefix (case-...) Show deleted blobs Add filter

Name

0_b4f317c51fdd46d089b558c... ***

0_b4f317c51fdd46d089b558c1266545d2_1.json

Overview Versions Snapshots Edit Generate SAS

```
1 {"messageId":137,"deviceId":"Raspberry Pi Web Client","temperature":28.29419891976918,"humidity":74.99730805649862,"EventProcessed":true}
2 {"messageId":138,"deviceId":"Raspberry Pi Web Client","temperature":25.928187295335203,"humidity":72.05698104844154,"EventProcessed":true}
3 {"messageId":141,"deviceId":"Raspberry Pi Web Client","temperature":28.056476120846604,"humidity":68.44382094693071,"EventProcessed":true}
4 {"messageId":143,"deviceId":"Raspberry Pi Web Client","temperature":25.59779331363464,"humidity":70.96470788872492,"EventProcessed":true}
5 {"messageId":144,"deviceId":"Raspberry Pi Web Client","temperature":27.668318701841113,"humidity":77.20033910648651,"EventProcessed":true}
6 {"messageId":145,"deviceId":"Raspberry Pi Web Client","temperature":30.41480008815595,"humidity":74.72594873717682,"EventProcessed":true}
7 {"messageId":146,"deviceId":"Raspberry Pi Web Client","temperature":29.455991171872704,"humidity":70.43151579087306,"EventProcessed":true}
8 {"messageId":147,"deviceId":"Raspberry Pi Web Client","temperature":29.43481314397131,"humidity":63.56514635650544,"EventProcessed":true}
9 {"messageId":148,"deviceId":"Raspberry Pi Web Client","temperature":29.42036702570004,"humidity":76.10019636508481,"EventProcessed":true}
10 {"messageId":149,"deviceId":"Raspberry Pi Web Client","temperature":31.84176092581463,"humidity":79.6372578562488,"EventProcessed":true}
11 {"messageId":150,"deviceId":"Raspberry Pi Web Client","temperature":25.75423840173954,"humidity":62.791545169828424,"EventProcessed":true}
12 {"messageId":151,"deviceId":"Raspberry Pi Web Client","temperature":27.794160287749975,"humidity":65.92995727086105,"EventProcessed":true}
13 {"messageId":152,"deviceId":"Raspberry Pi Web Client","temperature":28.243807062254252,"humidity":65.73780074751548,"EventProcessed":true}
14 {"messageId":154,"deviceId":"Raspberry Pi Web Client","temperature":31.92193863758927,"humidity":65.51378791578657,"EventProcessed":true}
15 {"messageId":155,"deviceId":"Raspberry Pi Web Client","temperature":30.969677724544976,"humidity":69.5991652695911,"EventProcessed":true}
16 {"messageId":156,"deviceId":"Raspberry Pi Web Client","temperature":31.46297753746157,"humidity":62.76487901172665,"EventProcessed":true}
17 {"messageId":157,"deviceId":"Raspberry Pi Web Client","temperature":27.660780731151297,"humidity":69.19458260893938,"EventProcessed":true}
18 {"messageId":159,"deviceId":"Raspberry Pi Web Client","temperature":28.639216406060394,"humidity":71.02459328116849,"EventProcessed":true}
19 {"messageId":160,"deviceId":"Raspberry Pi Web Client","temperature":31.92898722274147,"humidity":77.22870626583418,"EventProcessed":true}
20 {"messageId":163,"deviceId":"Raspberry Pi Web Client","temperature":25.842716380900413,"humidity":64.28618167228004,"EventProcessed":true}
21 {"messageId":164,"deviceId":"Raspberry Pi Web Client","temperature":29.596278491354678,"humidity":72.862936128122,"EventProcessed":true}
22 {"messageId":165,"deviceId":"Raspberry Pi Web Client","temperature":28.409417244337243,"humidity":61.91562999876396,"EventProcessed":true}
23 {"messageId":167,"deviceId":"Raspberry Pi Web Client","temperature":28.0108493088044772,"humidity":68.557999519584,"EventProcessed":true}
24 {"messageId":168,"deviceId":"Raspberry Pi Web Client","temperature":29.90156031606052,"humidity":60.758638858001454,"EventProcessed":true}
25 {"messageId":170,"deviceId":"Raspberry Pi Web Client","temperature":30.389241933054517,"humidity":69.212351080008095,"EventProcessed":true}
26 {"messageId":171,"deviceId":"Raspberry Pi Web Client","temperature":28.37044241538437,"humidity":74.98091044634951,"EventProcessed":true}
27 {"messageId":176,"deviceId":"Raspberry Pi Web Client","temperature":28.146925266335245,"humidity":74.28004282770246,"EventProcessed":true}
28 {"messageId":177,"deviceId":"Raspberry Pi Web Client","temperature":26.37030371774628,"humidity":76.9538268472931,"EventProcessed":true}
29 {"messageId":178,"deviceId":"Raspberry Pi Web Client","temperature":31.0584362480063,"humidity":63.7153268130612,"EventProcessed":true}
30 {"messageId":179,"deviceId":"Raspberry Pi Web Client","temperature":27.300985514034355,"humidity":60.640213319819324,"EventProcessed":true}
31 {"messageId":182,"deviceId":"Raspberry Pi Web Client","temperature":31.231224857009956,"humidity":74.6445881121089,"EventProcessed":true}
32 {"messageId":185,"deviceId":"Raspberry Pi Web Client","temperature":29.677450042714593,"humidity":75.44596591627632,"EventProcessed":true}
```

Json Preview