

Process on Azure cloud

Step 1: Create an Events Hub resource(namespace)

Home > Event Hubs >

Event Hubs

University of Minnesota (UnivOfMinn.onmicrosoft...)

+ Create Manage view ...

Filter for any field...

Name ↑↓

stockfeeddatademo

stockfeeddatademo

Event Hubs Namespace

Search (Ctrl+ /)

+ Event Hub Delete Refresh

Overview

- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Events

Settings

- Shared access policies
- Scale
- Geo-Recovery
- Encryption

Essentials

Resource group ([move](#))
[data-grp1](#)

Status
Active

Location
East US

Subscription ([move](#))
[Azure subscription 1](#)

Subscription ID
a8174be4-e521-4505-8804-915765d8234c

Host name
stockfeeddatademo.servicebus.windows.net

Created
Wednesday, April 27, 2022, 09:44:05 CDT

Updated
Wednesday, April 27, 2022, 09:44:50 CDT

Zone Redundancy
Enabled

Pricing tier
[Basic](#)

Throughput Units
[1 unit](#)

Auto-inflate throughput units
[Not Supported](#)

Local Authentication
[Enabled](#)

Step2: Create an actual instance of eventhub

stockfeeddatademo

Event Hubs Namespace

Search (Ctrl+ /)

Geo-Recovery

Encryption

Properties

Locks

Entities

- Event Hubs

Monitoring

- Alerts
- Metrics
- Diagnostic settings
- Logs

Automation

[Home](#) > [Event Hubs](#) > [stockfeeddatademo](#) >

Create Event Hub ...

Event Hubs

Name ^{*} ⓘ

pythonfeeddata ✓

Partition Count ⓘ

2

Message Retention ⓘ

1

Capture ⓘ

On

Off

Partition count- the more- the faster the data gets ingested

Capture- can't see what's being streamed unless we use stream analytics

Step3: Give manage access policy using shared access policies option for the newly created event hub instance

pythonfeeddata (stockfeeddatademo/pythonfeeddata) | Shared access policies ...

Search (Ctrl+/) << + Add

Overview

Access control (IAM)

Diagnose and solve problems

Settings

Shared access policies

Properties

Locks

Search to filter items...

Policy	Claims
no policies have been set up yet.	

Policy name ^{*}

RootUserAccessPolicy ✓




☒ Manage

☒ Send

☒ Listen

Step4: Get the connection string to use in the python script

SAS Policy: RootUserAccessPo... ×


 Save  Discard  Delete ...

☒ Manage


☐ Send

☐ Listen


Primary key




Secondary key



Connection string–primary key




Connection string–secondary key



Step5: Create a new stream analytics job

[Home](#) > [Stream Analytics jobs](#) >

New Stream Analytics job ...

 This will create a new Stream Analytics job. You will be charged according to the usage of the job.

Job name *

datafeedstockdemo



Subscription *

Azure subscription 1



Resource group *

data-grp1



[Create new](#)

Location *


East US



Hosting environment 

Cloud


Edge

Streaming units (1 to 192) 



1

☐

Secure all private data assets needed by this job in my Storage account. 

Create

Step6: In the newly created stream analytics job create a “Input” using “Inputs” and select “Event Hub”

The screenshot displays the Azure Stream Analytics portal interface for a job named "datafeedstockdemo". The left-hand navigation pane includes sections for "Overview", "Settings", "Job topology", and "Configure". The "Inputs" option under "Job topology" is highlighted with an orange background. Below the navigation pane, the "Add stream input" button is visible, and its dropdown menu is open, listing "Blob storage/ADLS Gen2", "Event Hub" (which is highlighted in orange), and "IoT Hub".

Step 7: Write python script that pulls the data in real time

The python script is programmed to pull data from the api every 30 seconds and push to it azure

```
[*****100%*****] 1 of 1 completed
SUCCESS
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
```

Step 8: Data is now reflected on Azure

Input preview [Test results](#)

Showing events from 'stream'. This list of events might not be complete. Select a specific time range to show all events during that period. Inferred data types may not be accurate.

Symbol	Name	livePrice	Volume_inMill	fiftydaymovavg	twohundaymovavg	prevdayclosingprice	Symbolindex	DailyPric
string	string	float	float	float	float	float	float	float
"ARKK"	"ARK Innovation ETF"	47.130001068115234	24.789641	61.431200103759764	94.94100002288819	48.869998931884766	2	63.400
"AMD"	"Advanced Micro Devi...	85.5199966430664	82.512504	106.40600006103516	117.75864994049073	89.63999938964844	0	93.889
"AMZN"	"Amazon.com, Inc."	2485.6298828125	13.479948	3054.2452099609377	3277.529260253906	2891.929931640625	1	3326.0
"AMD"	"Advanced Micro Devi...	85.5199966430664	82.512504	106.40600006103516	117.75864994049073	89.63999938964844	0	93.889

While sampling data, no data was received from '1' partitions. Ln 12, Col 14

Step 9: Before that create a storage account and a container

[Home](#) > [Storage accounts](#) >

Create a storage account ...

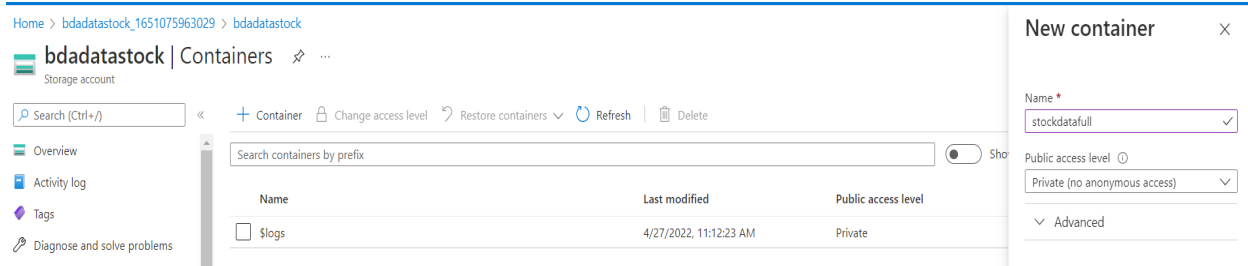
Running final validation...

Basics Advanced Networking Data protection Encryption Tags Review + create

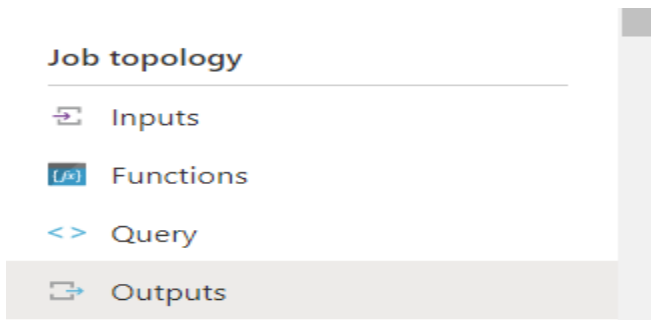
Basics

Subscription

Azure subscription 1



Step 10: In the same stream analytics job create output alias to blob storage.



Step 11: Update the test query in the stream analytics job with the correct aliases and also to only access the latest records everytime python script runs

WITH LastInWindow AS

```
(
    SELECT
        MAX(EventProcessedUtcTime) AS LastEventTime
    FROM
        stream TIMESTAMP BY EventProcessedUtcTime
    GROUP BY
        TumblingWindow(second, 5)
)
```

```
SELECT stream.Symbol, stream.Name, stream.liveprice,
stream.volume_inmill,stream.fiftydaymovavg,stream.twohundaymovavg,
stream.prevdayclosingprice,stream.DailyPrice, stream.Date
INTO FinalTable
FROM
```

stream TIMESTAMP BY EventProcessedUtcTime
 INNER JOIN LastInWindow
 ON DATEDIFF(second, stream, LastInWindow) BETWEEN 0 AND 10
 AND stream.EventProcessedUtcTime =
 LastInWindow.LastEventTime;

Step 12: Save the query and click start on the stream analytics job overview page to start the job that executes your query repeatedly.

The screenshot shows the Azure Stream Analytics job overview page for a job named 'datafeedstockdemo'. The left sidebar contains navigation options: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings (Properties, Locks), Job topology (Inputs, Functions, Query, Outputs), and Configure. The main area displays the job's 'Essentials' and 'Overview' tabs. The 'Essentials' tab shows the job's status as 'Created', its location as 'East US', and its subscription ID. The 'Overview' tab shows the job's inputs, which include an 'eventhubstreamdemo' Event Hub. The 'Outputs' section is currently empty. A 'Query' editor on the right displays a SQL query for a stream analytics job.

Search (Ctrl+/) << Start Stop Delete Refresh

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Properties

Locks

Job topology

Inputs

Functions

Query

Outputs

Configure

Created

Essentials

Resource group (move) : data-grp1

Status : Created

Location : East US

Subscription (move) : Azure subscription 1

Subscription ID : a8174be4-e521-4505-8804-915765d8234c

Created : Wednesday, April 27, 2022, 10:00:53 AM

Started : -

Output watermark : -

Cluster : Shared

Hosting environment : Cloud

Overview

Inputs

1

eventhubstreamdemo

Event Hub

Outputs

Query

```
1 SELECT
2 *
3 INTO
4 [FinalTable]
5 FROM
6 [eventhubstreamdemo]
```

Step 13: Now the files are available in the container

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state	
1198986563_c5f7f7ab8c97459aa0ac588e94c1137a...	4/30/2022, 10:03:36 ...	Hot (Inferred)		Block blob	30.44 KiB	Available	...

Process on Google Cloud Platform

Step 14: Move on to Google cloud

Step 15: Create a new project

Step 16: Create a new storage bucket ready to store files from azure

Step 17: Search for Data Transfer Service tool and create a new Data Transfer Job. Select minimum frequency possible(1 hr)

←

Create a transfer job

✓

Choose a source

Azure container

✓

Choose a destination

Cloud Storage bucket: stockdatabucket

✓

Choose settings

Never delete files

4

Scheduling options

Run job every hour • Starting now

CREATE

CANCEL

Step 18: When you run this job files are transferred from azure to gcp

Latest run of 'azure to gcp'

Status: Success

Start time: May 2, 2022 at 10:05:15 AM GMT-5

End time: May 2, 2022 at 10:05:36 AM GMT-5

Duration: 20 sec

Progress

100%

Changes made after start time may not be discovered until the next transfer

Data transferred

65.41 KB of 65.41 KB

1 of 1 file

Errors

0

Data skipped

0 B

0 files

Average speed estimate

3.27 KB/s

Estimated based on data transferred and duration

Run history

Filter

Filter runs

Status

Start time

Progress

Data transferred

Size

Errors

Duration

End time

✓

Success

May 2, 2022 at 10:05:15 AM GMT-5

100%

65.41 KB

65.41 KB

—

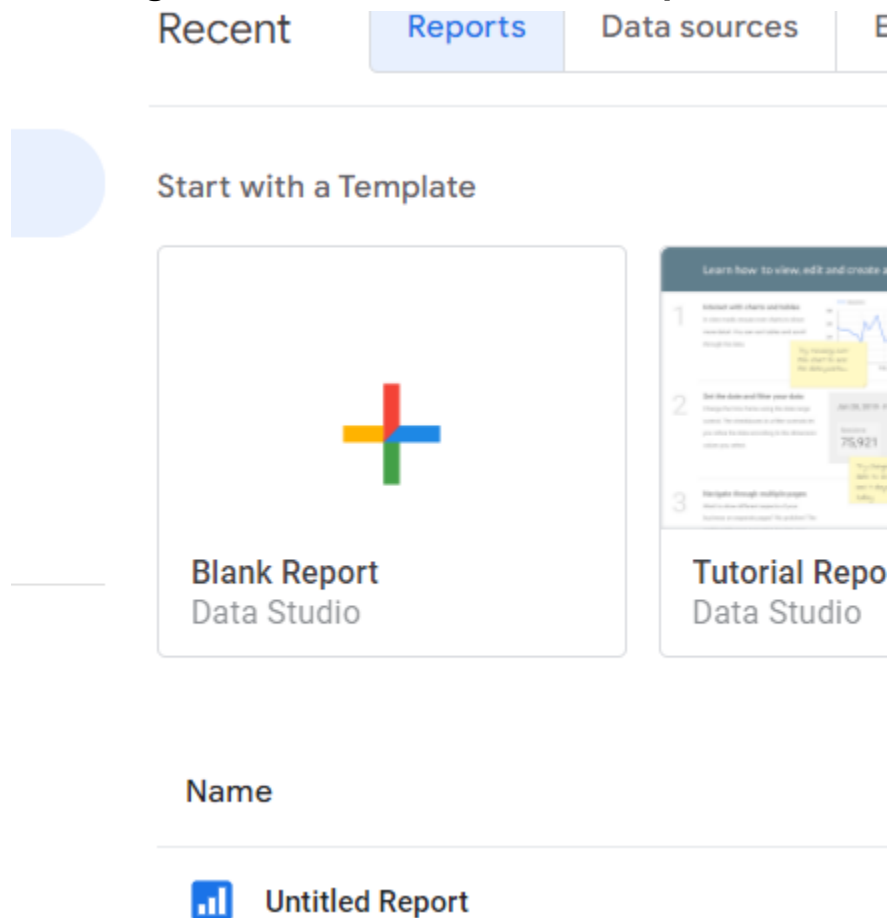
20 sec

May 2, 2022 at 10:05:36 AM GMT-5

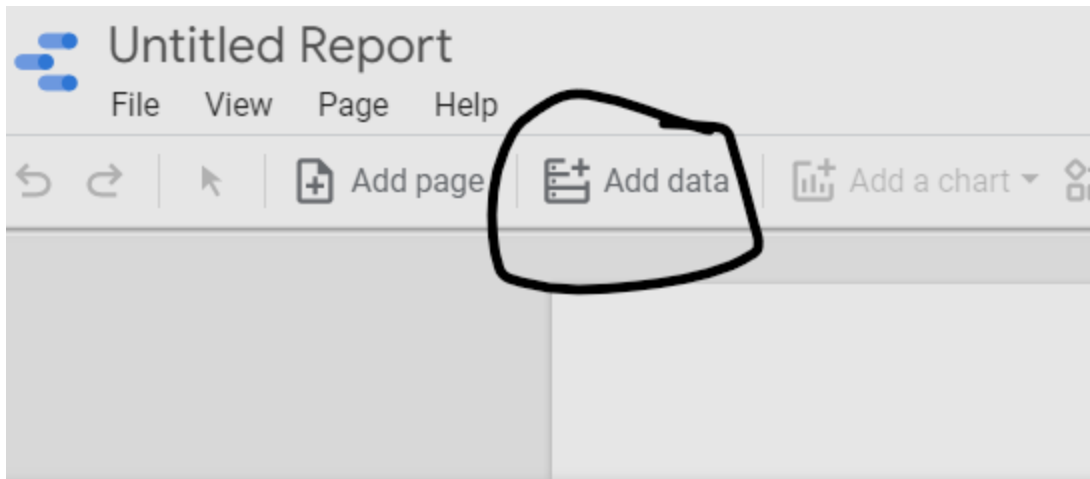
Process on Google Data Studio

Step 19: To perform visualization on the data let's get it into Google Data studio

On Google Data Studio, add a new report



Step 20: Click Add Data from the toolbar and search for Google Cloud Storage



Add data to report

Connect to data

My data sources

🔍 Search

Connect to CSV (comma-separated values) files.



Cloud SQL for MySQL



By Google

Connect to Google Cloud SQL for MySQL databases.



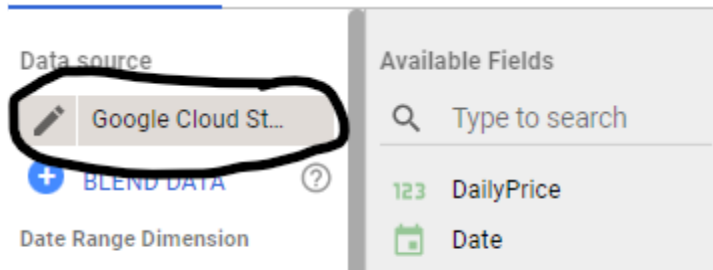
Google Cloud Storage



By Google

See your files in Google Cloud Storage.

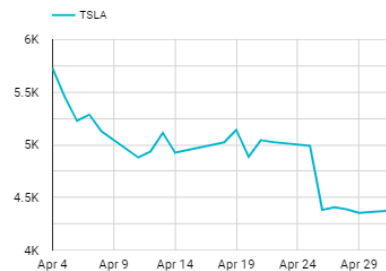
Step 21: Add the name of the Google cloud storage bucket with the data file



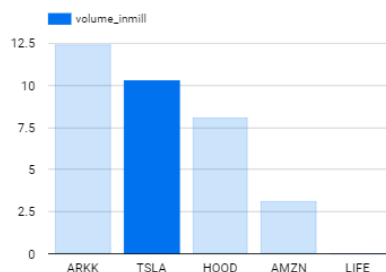
File in GCP becomes data source for the visualizations in google data studio

Step 22: Make visualizations fit for the business problem

Historic Daily Closing Prices of Trending Stocks in Last Month



The Volume of Shares (in Million) of Trending Stocks



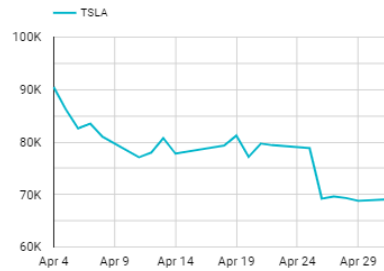
%change in liveprice of selected company
vs it's previous day closing price

+1%

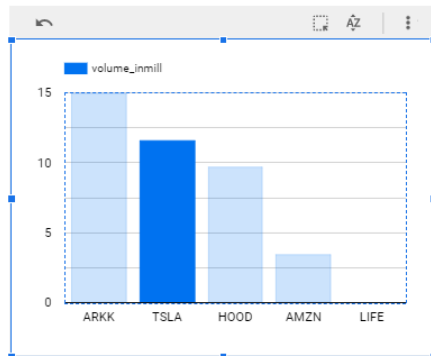
(liveprice are 1% **above** target)

GCP data transfer job runs only every hour, so these plots change every hour

Historic Daily Closing Prices of Trending Stocks in Last Month



The Volume of Shares (in Million) of Trending Stocks



%change in liveprice of selected company
vs its previous day closing price

+0%

(liveprice are 0% **above** target)