

Assignment 5

Panourgia Evangelia (t8190130)
Papadatos Ioannis (t8190314)
Professor: Chatziantoniou Damianos

Latest Update: May 8, 2022

School of Management Science and Technology,
Athens University of Economics and Business

Contents

0. Introduction.....	3
1. Create a trial account.....	3
2. Setup an Event Hub.....	3
3. Generate a Security Access Signature.....	4
4. Edit Generator.html.....	5
5. Feed the Event Hub.....	5
6. Setup a Storage account.....	5
7. Upload the Reference Data files to your storage account.....	6
8. Setup a Stream Analytics Job.....	7
9. Input setup.....	8
10. Output setup.....	10

II Queries

Query 1: Show the total “Amount” of “Type = 0” transactions at “ATM Code = 21” of the last 10 minutes. Repeat as new events keep flowing in (use a sliding window).

Query 2: Show the total “Amount” of “Type = 1” transactions at “ATM Code = 21” of the last hour. Repeat once every hour (use a tumbling window).

Query 3: Show the total “Amount” of “Type = 1” transactions at “ATM Code = 21” of the last hour. Repeat once every 30 minutes (use a hopping window).

Query 4: Show the total “Amount” of “Type = 1” transactions per “ATM Code” of the last one hour (use a sliding window).

Query 5: Show the total “Amount” of “Type = 1” transactions per “Area Code” of the last hour. Repeat once every hour (use a tumbling window).

Query 6: Show the total “Amount” per ATM’s “City” and Customer’s “Gender” of the last hour. Repeat once every hour (use a tumbling window).

Query 7: Alert (SELECT “1”) if a Customer has performed two transactions of “Type = 1” in a window of an hour (use a sliding window).

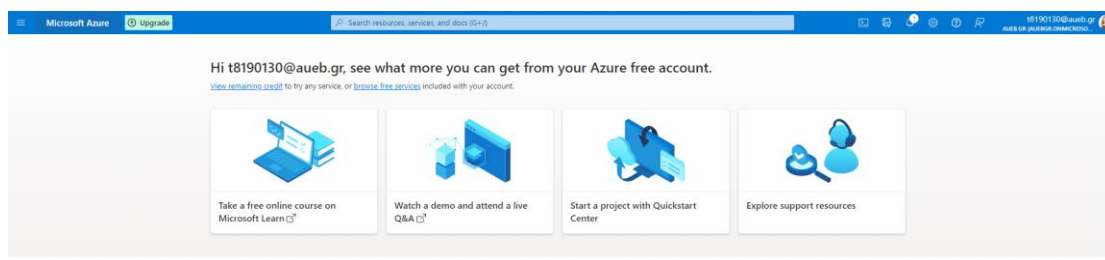
Query 8: Alert (SELECT “1”) if the “Area Code” of the ATM of the transaction is not the same as the “Area Code” of the “Card Number” (Customer’s Area Code) - (use a sliding window)

0.Introduction

Azure Stream Analytics is a fully managed, real-time analytics service designed to help you analyze and process fast moving streams of data that can be used to get insights, build reports or trigger alerts and actions [1]. For this assignment, I demonstrate proficiency in using Stream Analytics by setting up a processing environment and running queries using real-time randomly generated data and reference data.

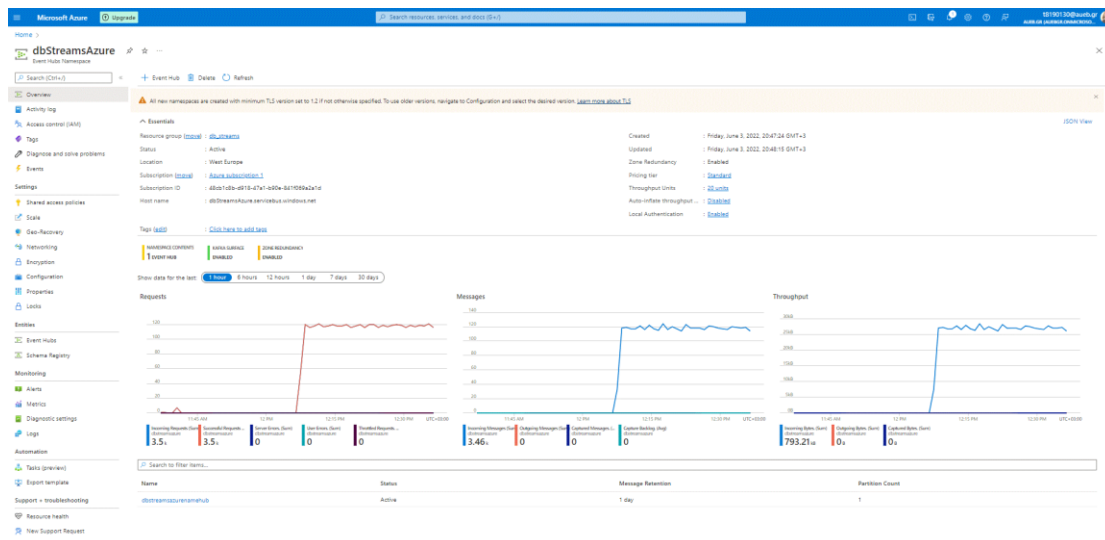
1. Create a Trial Account

I created a Microsoft Azure account using my university e-mail address.



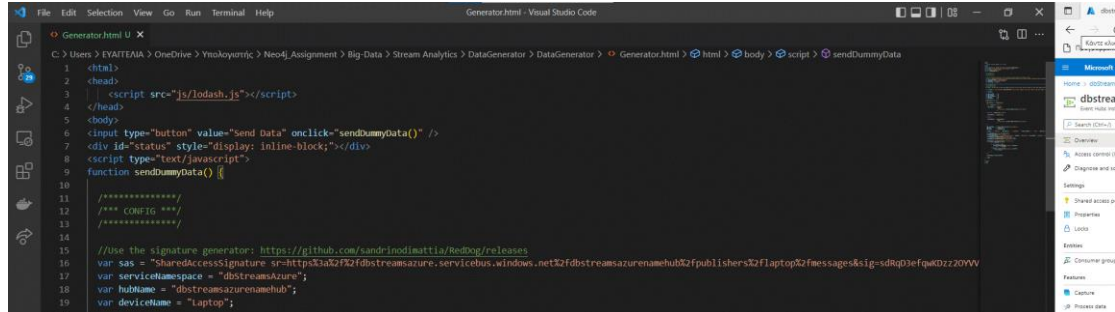
2. Setup an event hub

Event Hubs is a fully managed, real-time data ingestion service that's simple, trusted, and scalable [2].



4. Edit Generator.html

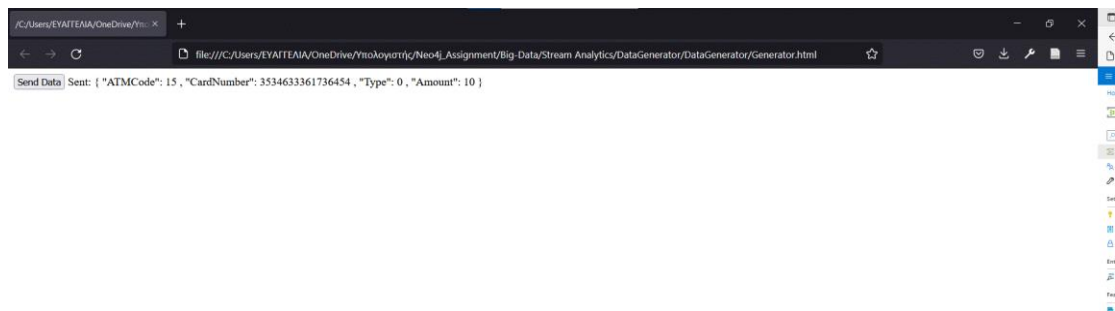
We edited Generator.html. We opened it with visual studio and we updated the CONFIG variables with our security access signature bearing in mind the suitable variables values that we created in step 3 (the previous step).



```
1 <html>
2 <head>
3   <script src="js/lodash.js"></script>
4 </head>
5 <body>
6   <input type="button" value="Send Data" onclick="sendDummyData()" />
7   <div id="status" style="display: inline-block;"></div>
8   <script type="text/javascript">
9     function sendDummyData() {
10
11       /*****
12        *** CONFIG ***
13        *****/
14
15       //Use the signature generator: https://github.com/sandrinodimattia/RedDog/releases
16       var sas = "SharedAccessSignature sr=https://dbstream.azure.servicebus.windows.net/2fbdstreamazurenamehub/2fpublishers2flaptop2fmessages&sig=sdRqD3efqkD2220VYV";
17       var serviceNameSpace = "dbstreamazure";
18       var hubName = "dbstreamazurenamehub";
19       var deviceId = "laptop";
```

5. Feed the Event Hub

We fed the Event Hub with the use of Generator.html. Specifically, we opened Generator.html in a web browser and pressed the “Send Data” button.



6. Set up a Storage account

An Azure storage account contains all of your Azure Storage data objects, including blobs, file shares, queues, tables, and disks. The storage account provides a unique namespace for your Azure Storage data that's accessible from anywhere in the world over HTTP or HTTPS. Data in your storage account is durable and highly available, secure, and massively scalable [3]. For the assignment a storage account was created in order to store the reference data to be used in the queries, as well as their output.

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

Home > Storage accounts >

Create a storage account

Basics Advanced Networking Data protection Encryption Tags Review + create

Create new

Instance details

If you need to create a legacy storage account type, please click here.

Storage account name

Region

Performance ☒ Standard: Recommended for most scenarios (general-purpose v2 account)
☐ Premium: Recommended for scenarios that require low latency.

Redundancy

☒ Make read access to data available in the event of regional unavailability.

Review + create Previous Next: Advanced >

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

Home > Storage accounts >

Create a storage account

Running final validation...

Basics Advanced Networking Data protection Encryption Tags Review + create

Basics

Subscription	Azure subscription 1
Resource Group	db_streams
Location	westeurope
Storage account name	evastorage1
Deployment model	Resource manager
Performance	Standard
Replication	Read-access geo-redundant storage (RA-GRS)

Advanced

Secure transfer	Enabled
Allow storage account key access	Enabled
Allow cross-tenant replication	Enabled
Default to Azure Active Directory	Disabled

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

Home >

evastorage1_1654533299219 | Overview

Deployment

Search (Ctrl+J) Delete Cancel Redeploy Refresh

We'd love your feedback! →

Your deployment is complete

Deployment name: evastorage1_1654533299219
Subscription: Azure subscription 1
Resource group: db_streams

Start time: 6/6/2022, 7:35:10 PM
Correlation ID: 7b3e05b2-5a72-476c-bbfb-706604115c4d

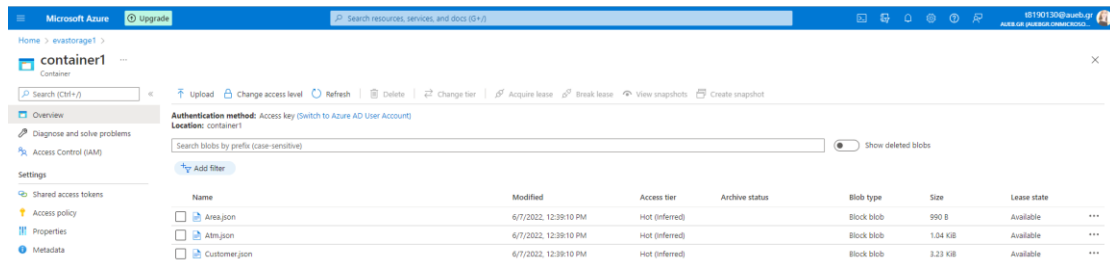
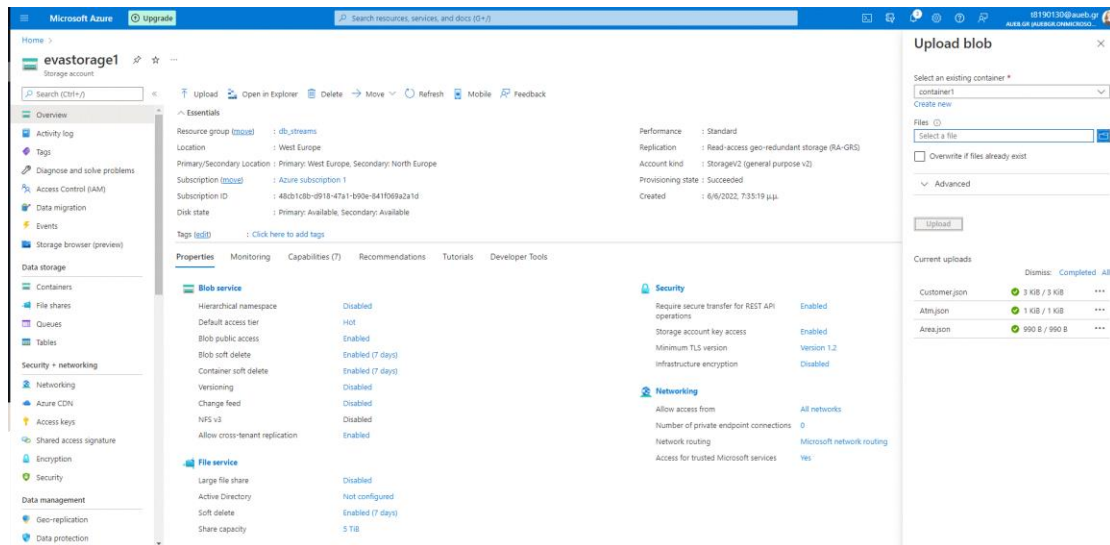
Deployment details (Download)
Next steps
Go to resource

Cost Management
Get notified to stay within your budget and prevent unexpected charges on your bill.
Set up cost alerts >

Microsoft Defender for Cloud
Secure your apps and infrastructure
Go to Microsoft Defender for Cloud >

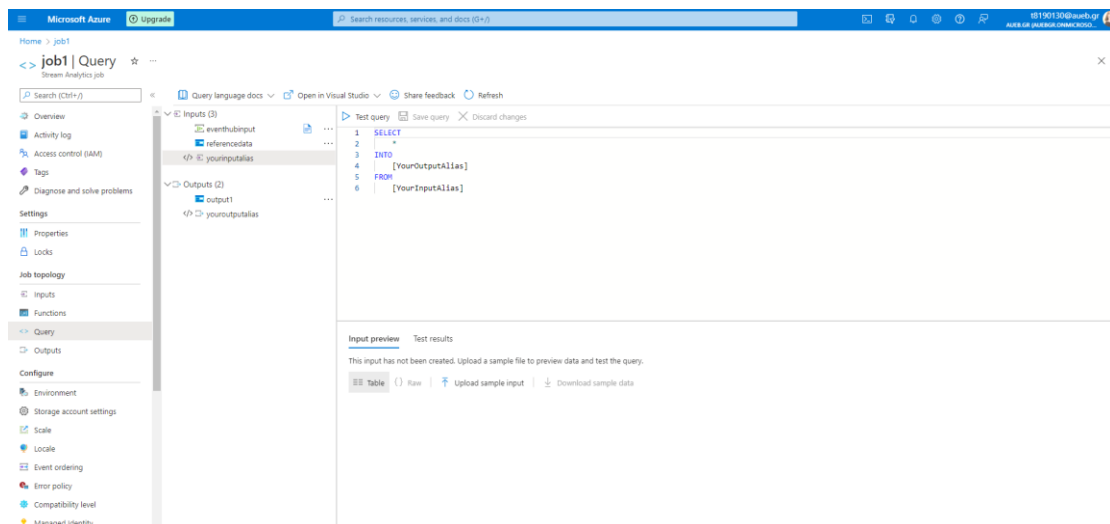
7. Upload the Reference Data files to your storage account

A blob container was created and the reference JSON files were uploaded.



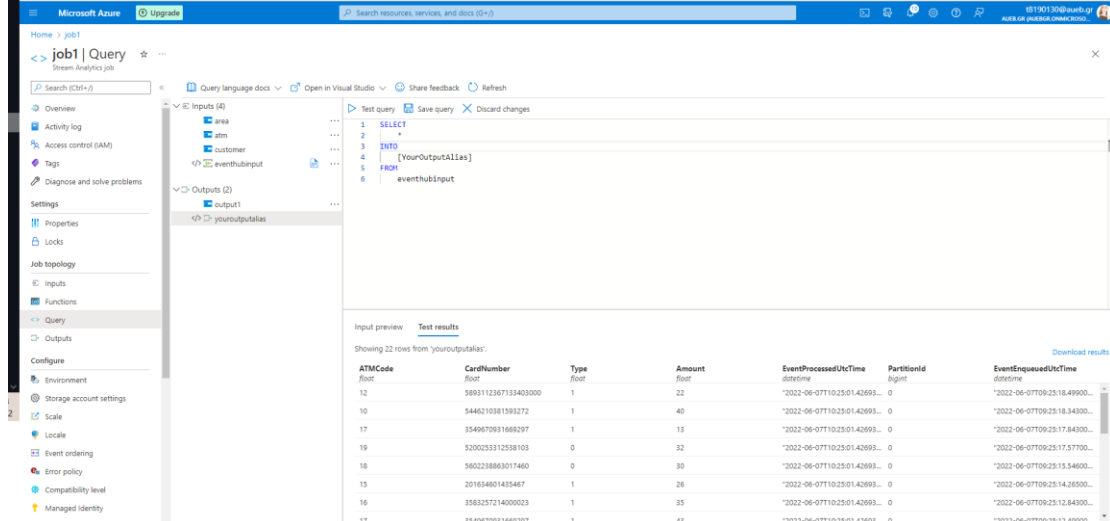
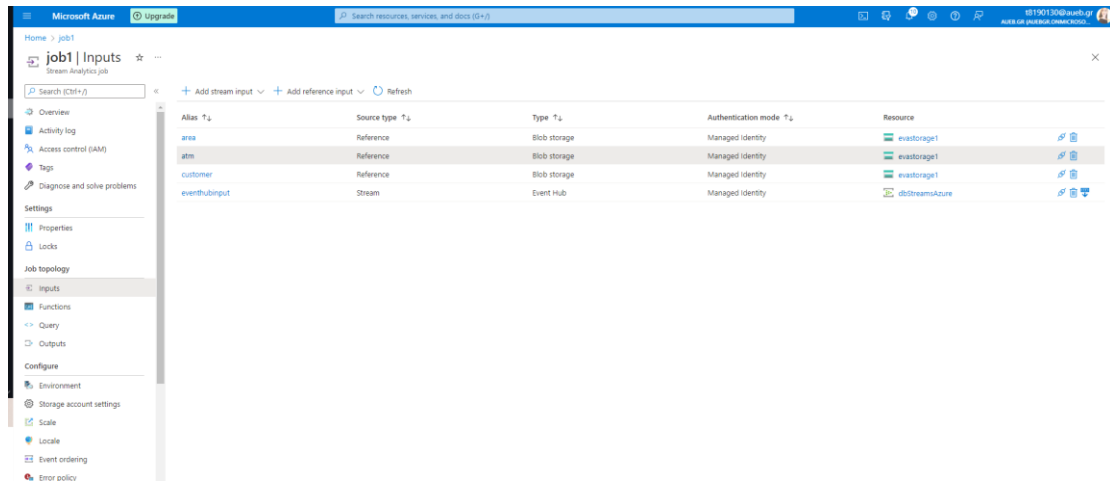
8. Setup a Stream Analytics Job

We created a stream analytics job bearing in mind the instructions from link[4].



9. Input setup

The reference data, as well as the Event Hub stream were set as inputs for the job. Successful setup of the stream input was confirmed by sampling data from it.

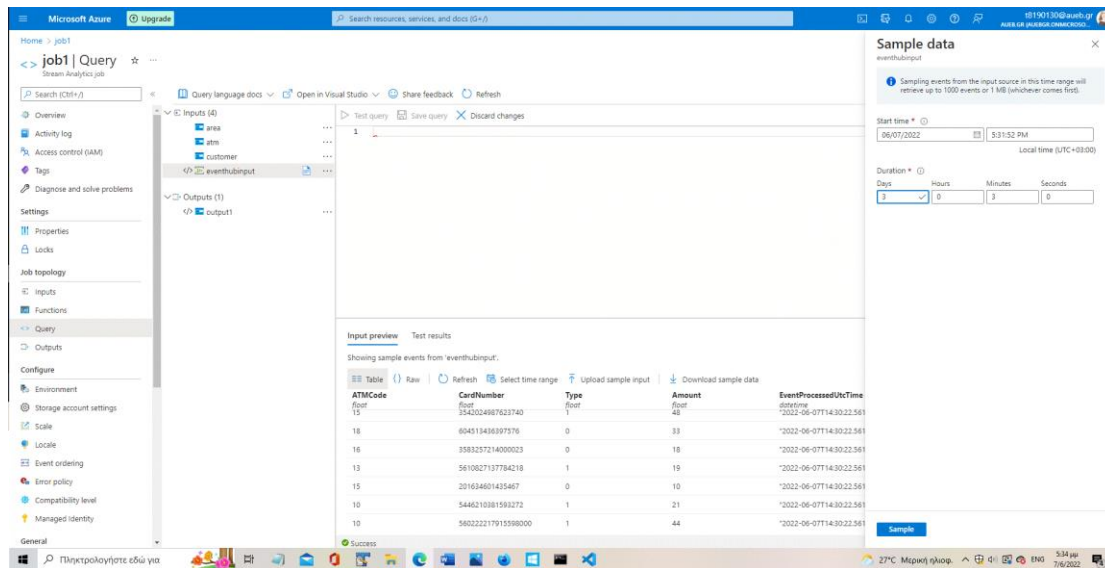


Alias	Source type	Type	Authentication mode	Resource
area	Reference	Blob storage	Managed identity	evastorage1
atm	Reference	Blob storage	Managed identity	evastorage1
customer	Reference	Blob storage	Managed identity	evastorage1
eventhubinput	Stream	Event Hub	Managed identity	dbstreamazure

```
1 SELECT
2 *
3 INTO
4 [YourOutputAllies]
5 FROM
6 eventhubinput
```

ATMCode	CardNumber	Type	Amount	EventProcessedUtcTime	PartitionId	EventEnqueuedUtcTime
12	5693112367133403000	1	22	"2022-06-07T10:25:01.42693...	0	"2022-06-07T09:25:18.49903...
10	5446210981593272	1	40	"2022-06-07T10:25:01.42693...	0	"2022-06-07T09:25:18.54300...
17	3549670931669297	1	13	"2022-06-07T10:25:01.42693...	0	"2022-06-07T09:25:17.84300...
19	3200253312538103	0	32	"2022-06-07T10:25:01.42693...	0	"2022-06-07T09:25:17.57700...
18	5602238863017460	0	30	"2022-06-07T10:25:01.42693...	0	"2022-06-07T09:25:15.54600...
15	201634801435467	1	26	"2022-06-07T10:25:01.42693...	0	"2022-06-07T09:25:14.26500...
16	3583257214000023	1	35	"2022-06-07T10:25:01.42693...	0	"2022-06-07T09:25:12.84300...
17	21476700316235357	1	23	"2022-06-07T10:25:01.42693...	0	"2022-06-07T09:25:12.84300...

For the sample I changes the range as following :



10. Output setup

We created a stream analytics job bearing in mind the instructions from link[4].



II Queries

In the interest of reproducibility the jobs were tested using the data sampled in Section 10 of this report, however jobs did also produce blobs in the output container, successfully. Both the sampled data and the results of the queries have been provided alongside this report in the folder **output_json**.

Query 1: Show the total “Amount” of “Type = 0” transactions at “ATM Code = 21” of the last 10 minutes. Repeat as new events keep flowing in (use a sliding window).

```
SELECT
    SUM([Amount]) AS Amount
INTO
    [output1]
FROM
    [eventhubinput]
```

WHERE

```
[ATMCode] = 21 AND [Type] = 0  
GROUP BY [ATMCode], SlidingWindow(minute, 10)
```

The screenshot shows the Microsoft Azure Stream Analytics job configuration interface. The top navigation bar includes the Microsoft Azure logo, a search bar, and user information. The main area is divided into three sections: a left sidebar with navigation options, a central job topology diagram, and a right-hand query editor and test results section.

Left Sidebar: Contains navigation options such as Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Properties, Locks, Job topology, Inputs, Functions, Query (selected), Outputs, Configure, Environment, and Storage account settings.

Job Topology: Shows a diagram with four inputs (area, atm, customer, eventhubinput) and one output (output1).

Query Editor: Displays the following SQL query:

```
1 SELECT  
2     SUM([Amount]) AS Amount  
3 INTO  
4     [output1]  
5 FROM  
6     [eventhubinput]  
7 WHERE  
8     [ATMCode] = 21 AND [Type] = 0  
9 GROUP BY [ATMCode], SlidingWindow(minute, 10)  
10
```

Test Results: Shows the output of the query. The table has one column, Amount, with a value of 165.

Reference [5]

Note : in folder output_json exists the exported output from azure interfaces for all queries.

Query 2: Show the total “Amount” of “Type = 1” transactions at “ATM Code = 21” of the last hour. Repeat once every hour (use a tumbling window).

```
SELECT  
    SUM([Amount]) AS Amount  
INTO  
    [output1]  
FROM  
    [eventhubinput]  
WHERE  
    [ATMCode] = 21 AND [Type] = 1  
GROUP BY [ATMCode], TumblingWindow(hour, 1)
```

Reference [6]

Note : in folder output_json exists the exported output from azure interfaces for all queries.

Query 3: Show the total “Amount” of “Type = 1” transactions at “ATM Code = 21” of the last hour. Repeat once every 30 minutes (use a hopping window).

```
SELECT
    SUM([Amount]) AS Amount
INTO
    [output1]
FROM
    [eventhubinput]
WHERE
    [ATMCode] = 21 AND [Type] = 1
GROUP BY [ATMCode], HoppingWindow(minute, 60, 30)
```

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

Home > job1

job1 | Query ☆ ...

Stream Analytics job

Search (Ctrl+/)

Query language docs Open in Visual Studio Share feedback Refresh

Test query Save query Discard changes

```

1 SELECT
2   SUM([Amount]) AS Amount
3 INTO
4   [output1]
5 FROM
6   [eventhubinput]
7 WHERE
8   [ATMCode] = 21 AND [Type] = 1
9 GROUP BY [ATMCode], HoppingWindow(minute, 60, 30)
10
11
12

```

Input preview Test results

Showing 2 rows from 'output1'.

Amount
143
143

Download results

Success Ln 10, Col 1

Reference [7]

Note : in folder output_json exists the exported output from azure interfaces for all queries.

Query 4: Query 4: Show the total “Amount” of “Type = 1” transactions per “ATM Code” of the last one hour (use a sliding window).

```

SELECT
    [ATMCode], SUM([Amount]) AS Amount
INTO
    [output1]
FROM
    [eventhubinput]
WHERE
    [Type] = 1
GROUP BY [ATMCode], SlidingWindow(hour, 1)

```

Microsoft Azure

Search resources, services, and docs (G+/)

Home > job1

job1 | Query ☆ ...

Stream Analytics job

Search (Ctrl+/)

Query language docs Open in Visual Studio Share feedback Refresh

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Properties

Locks

Job topology

Inputs

Functions

Query

Outputs

Configure

Environment

Storage account settings

Inputs (4)

area

atm

customer

eventhubinput

Outputs (1)

output1

Test query Save query Discard changes

```

1 SELECT
2   [ATMCode], SUM([Amount]) AS Amount
3 INTO
4   [output1]
5 FROM
6   [eventhubinput]
7 WHERE
8   [Type] = 1
9 GROUP BY [ATMCode], SlidingWindow(hour, 1)
10
11
12

```

Input preview Test results

Showing 12 rows from 'output1':

ATMCode	Amount
15	878
19	516
21	143
17	178

Download results

Success Ln 10, Col 1

Reference[5]

Note : in folder output_json exists the exported output from azure interfaces for all queries.

Query 5: Show the total “Amount” of “Type = 1” transactions per “Area Code” of the last hour. Repeat once every hour (use a tumbling window).

```

SELECT
    [atm].[area_code], SUM([eventhubinput].[Amount]) AS Amount
INTO
    [output1]
FROM
    [eventhubinput]
    INNER JOIN [atm] ON [eventhubinput].[ATMCode] = [atm].[atm_code]
WHERE
    [eventhubinput].[Type] = 1
GROUP BY [atm].[area_code], TumblingWindow(hour, 1)

```

The screenshot shows the Microsoft Azure portal interface for a Databricks job. The left sidebar contains navigation options like 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 a query for a job named 'job1'. The query is as follows:

```

1 SELECT
2   [atm].[area_code], SUM([eventhubinput].[Amount]) AS Amount
3 INTO
4   [output1]
5 FROM
6   [eventhubinput]
7 INNER JOIN [atm] ON [eventhubinput].[ATMCode] = [atm].[atm_code]
8 WHERE
9   [eventhubinput].[Type] = 1
10 GROUP BY [atm].[area_code], TumblingWindow(hour, 1)

```

Below the query, the 'Test results' section shows 11 rows from 'output1':

area_code	Amount
6	49
5	878
1	684
3	178
2	316
7	179
11	459
4	441

Reference[6]

Note : in folder output_json exists the exported output from azure interfaces for all queries.

Query 6: Show the total “Amount” per ATM’s “City” and Customer’s “Gender” of the last hour. Repeat once every hour (use a tumbling window).

SELECT

[area].[area_city], [customer].[gender], SUM([eventhubinput].[Amount]) AS Amount

INTO

[output1]

FROM

[eventhubinput]

INNER JOIN [atm] ON [eventhubinput].[ATMCode] = [atm].[atm_code]

INNER JOIN [area] ON [atm].[area_code] = [area].[area_code]

INNER JOIN [customer] ON [eventhubinput].[CardNumber] = [customer].

[card_number]

GROUP BY [area].[area_city], [customer].[gender], TumblingWindow(hour, 1)

The screenshot shows the Microsoft Azure portal interface for a Stream Analytics job. The left sidebar contains navigation options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Properties, Locks, Job topology, Inputs, Functions, Query, Outputs, Configure, Environment, Storage account settings, Scale, Locale, Event ordering, Error policy, Compatibility level, and Managed identity. The main area displays the 'job1 | Query' tab with a search bar and a 'Test query' button. The query editor shows a SQL query that joins 'area', 'atm', and 'customer' tables with 'eventhubinput' and aggregates the 'Amount' by area, city, and gender using a tumbling window. The 'Test results' tab shows a preview of 16 rows from 'output1'.

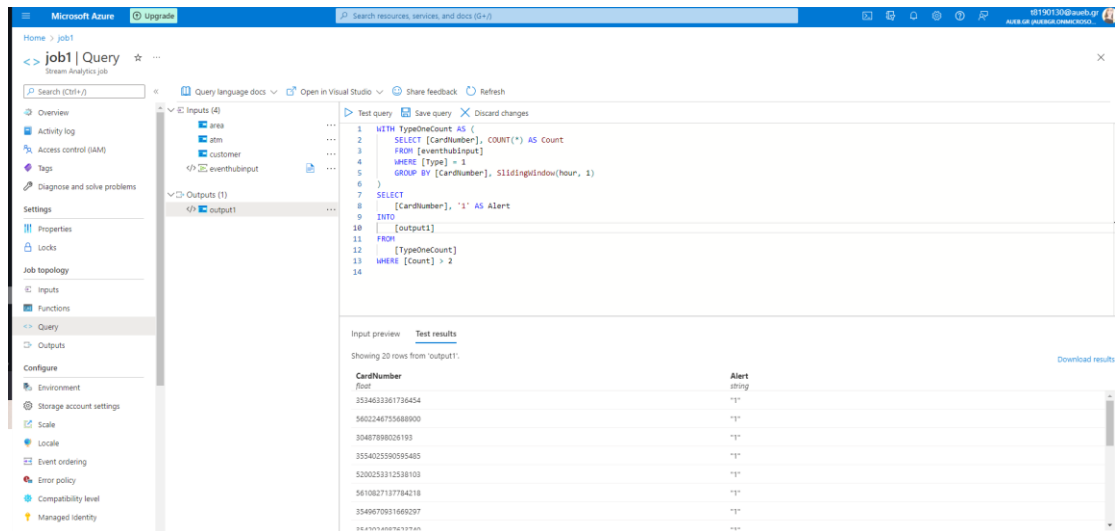
area_city	gender	Amount
"Tacoma"	"Female"	347
"Schaumburg"	"Female"	1172
"Vancouver"	"Male"	290
"Baltimore"	"Female"	112
"Springfield"	"Male"	792
"Canton"	"Male"	210
"Vancouver"	"Female"	39
"Vancouver"	"Female"	ccc

Reference[6]

Note : in folder output_json exists the exported output from azure interfaces for all queries.

Query 7: Alert (SELECT “1”) if a Customer has performed two transactions of “Type = 1” in a window of an hour (use a sliding window).

```
WITH TypeOneCount AS (
    SELECT [CardNumber], COUNT(*) AS Count
    FROM [eventhubinput]
    WHERE [Type] = 1
    GROUP BY [CardNumber], SlidingWindow(hour, 1)
)
SELECT
    [CardNumber], '1' AS Alert
INTO
    [output1]
FROM
    [TypeOneCount]
WHERE [Count] > 2
```

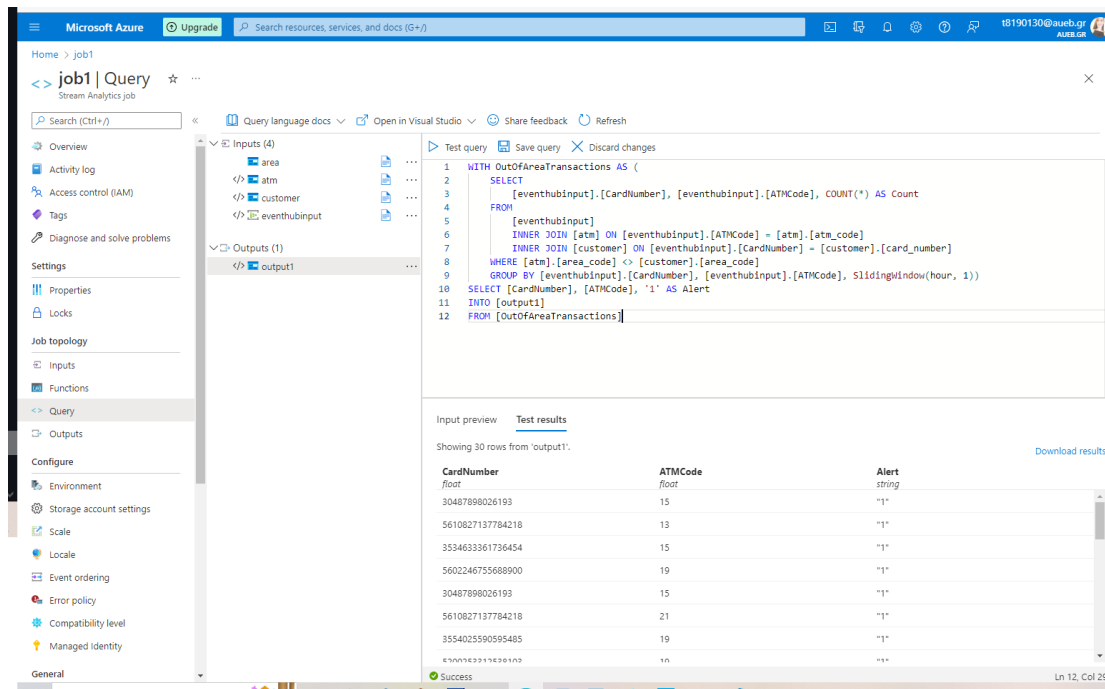


Reference[8]

Note : in folder output_json exists the exported output from azure interfaces for all queries.

Query 8: Alert (SELECT “1”) if the “Area Code” of the ATM of the transaction is not the same as the “Area Code” of the “Card Number” (Customer’s Area Code) - (use a sliding window)

```
WITH OutOfAreaTransactions AS (
    SELECT
        [eventhubinput].[CardNumber], [eventhubinput].[ATMCode], COUNT(
*) AS Count
    FROM
        [eventhubinput]
        INNER JOIN [atm] ON [eventhubinput].[ATMCode] = [atm].[atm_code]
        INNER JOIN [customer] ON [eventhubinput].[CardNumber] = [customer].[card_number]
    WHERE [atm].[area_code] <> [customer].[area_code]
    GROUP BY [eventhubinput].[CardNumber], [eventhubinput].[ATMCode], S
lidingWindow(hour, 1))
SELECT [CardNumber], [ATMCode], '1' AS Alert
INTO [output1]
FROM [OutOfAreaTransactions]
```

Reference[8]

Note : in folder output_json exists the exported output from azure interfaces for all queries.

12. Conclusion

In conclusion, as this assignment is the last one, we would like to mention that this course helped as to learn useful tools – systems for big data. Specifically, we worked with Hadoop, Redis, neo4j and azure in practice. Last but not least, we would like to thank our professor Damianos Chatziantoniou for the course. Good summer!

References

- [Azure Streams Analytics][1] : [Azure Stream Analytics documentation | Microsoft Docs](#)
- [Event Hub][2] : <https://azure.microsoft.com/en-us/services/event-hubs/#overview>
- [Storage Account][3] : <https://docs.microsoft.com/en-us/azure/storage/common/storage-account-overview>
- [Create Stream Analytics Job][4] : <https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-quick-create-portal>
- [Sliding Window][5] : [Sliding Window \(Azure Stream Analytics\) - Stream Analytics Query | Microsoft Docs](#)
- [Tumbling Window][6] : [Tumbling Window \(Azure Stream Analytics\) - Stream Analytics Query | Microsoft Docs](#)

- [Hopping Window][7] : [Hopping Window \(Azure Stream Analytics\) - Stream Analytics Query | Microsoft Docs](#)
- [Sliding Window][8] : [Sliding Window \(Azure Stream Analytics\) - Stream Analytics Query | Microsoft Docs](#)