

# Data Engineering in the Cloud

Build Aggregates and Output Them to Synapse Analytics

Xuemao Zhang  
East Stroudsburg University

January 18, 2025

# Outline

- After the real-time streaming is completed, we build aggregates and output these to Synapse Analytics using Stream Analytics windowing functions
- This lab is identical to the last lab except for the queries in and

# Lab

- Step 1. Create a datalake (storage account with hierarchical namespace enabled)
  - ▶ To lower the cost, you may choose Redundancy as LRS
  - ▶ Create a container in the storage account for the Synapse to use
- Step 2. Create an Azure Synapse workspace and
  - ▶ Create a dedicated SQL pool
  - ▶ Create a table in the SQL pool with the following SQL code

```
CREATE TABLE [dbo].[Orders]
(
  [OrderID] varchar(10),
  [Quantity] int,
  [UnitPrice] decimal(5,2),
  [DiscountCategory] varchar(10)
)
```

# Lab

- Create another table in the SQL pool with the following SQL code

```
CREATE TABLE [dbo].[Stats]
(
  [DiscountCategory] varchar(10),
  [AveQuantiy] decimal(5,2),
  [AvePrice] decimal(5,2),
  [Count] int,
  [TotalQuantiy] int,
  [TotalPrice] decimal(5,2)
)
```

# Lab

- That is, we created two tables for the database

The screenshot displays the Microsoft Azure Synapse Analytics interface. The top navigation bar shows 'Microsoft Azure | Synapse Analytics | synapse729'. Below this, the 'Data' section is active, showing a 'Workspace' view. The left sidebar lists the database structure: 'SQL database' (1) -> 'sqlpool (SQL)' -> 'Tables' -> 'dbo.Orders' and 'dbo.Stats'. The main pane shows the SQL script for creating these tables. The script is as follows:

```
1 CREATE TABLE [dbo].[Orders]
2 (
3     [OrderID] varchar(10),
4     [Quantity] int,
5     [UnitPrice] decimal(5,2),
6     [DiscountCategory] varchar(10)
7 )
8
9
10 CREATE TABLE [dbo].[Stats]
11 (
12     [DiscountCategory] varchar(10),
13     [AveQuantity] decimal(5,2),
14     [AvePrice] decimal(5,2),
15     [Count] int,
16     [TotalQuantity] int,
17     [TotalPrice] decimal(5,2)
18 )
19
20
```

The interface also shows a 'Results' tab at the bottom, which is currently empty. The 'Connect to' dropdown is set to 'sqlpool'.

# Lab

- Step 3. Search for Event Hubs to create an event hubs namespace
  - ▶ Create a Event hub in the event hubs namespace
- Step 4. Create a Stream Analytics job (with **two outputs** shown below)
  - ▶ Add the storage account
  - ▶ Add an input under Job Topology with input from the Event hub
  - ▶ Add one output under Job Topology with output of input data to the Synapse Analytics
  - ▶ Add the 2nd output under Job Topology with output of the statistics of the input data to the Synapse Analytics
  - ▶ Click on the “Query” option under Job topology with code in the next slide
- Data types (Azure Stream Analytics): <https://learn.microsoft.com/en-us/stream-analytics-query/data-types-azure-stream-analytics>

# Lab

```
SELECT
    OrderID, Quantity, UnitPrice, DiscountCategory
INTO
    [sqlpool]
FROM
    [demohub];
```

# Lab



Home > jobstream1

<> jobstream1 | Query ☆ ...

Stream Analytics job

Search



Query can't be edited while a job is running. You can stop the job to edit the query.

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Job topology

Inputs

Functions

<> Query

Outputs

No-code editor (preview)

Settings

Environment

Storage account settings

Scale

Locale

Event ordering



Inputs (1)



demohub



Outputs (2)



sqlpool



statsoutput



Functions (0)

Test query



Save query



Discard changes

```
1 SELECT
2   OrderID, Quantity, UnitPrice, DiscountCategory
3 INTO
4   [sqlpool]
5 FROM
6   [demohub];
7
8 SELECT
9   DiscountCategory,
10  AVG(Quantity) AS AveQuantity,
11  SUM(Quantity * UnitPrice) / SUM(Quantity) AS AvePrice,
12  COUNT(*) AS Count,
13  SUM(Quantity) AS TotalQuantity,
14  SUM(Quantity * UnitPrice) AS TotalPrice
15 INTO
16   [statsoutput]
17 FROM
18   (
19     SELECT
20       DiscountCategory,
```

Input preview

Test results

Job simulation (preview)

Showing sample events from 'demohub'.



Table Raw Refresh Select time range Upload sample input ...

# Lab

- Step 5: Start the Stream Analytics job
- Step 6: Go to Event Hubs Namespace to generate real-time data or use other applications to generate data

```
[
  {
    "OrderID": "1",
    "Quantity": 5,
    "UnitPrice": 120,
    "DiscountCategory": "Category 1"
  },
  {
    "OrderID": "2",
    "Quantity": 10,
    "UnitPrice": 200,
    "DiscountCategory": "Category 2"
  },
  {
    "OrderID": "3",
    "Quantity": 15,
    "UnitPrice": 300,
    "DiscountCategory": "Category 3"
  },
  {
    "OrderID": "4",
    "Quantity": 20,
    "UnitPrice": 400,
    "DiscountCategory": "Category 4"
  },
  {
    "OrderID": "5",
    "Quantity": 25,
    "UnitPrice": 500,
    "DiscountCategory": "Category 5"
  }
]
```

- Step 7. Go to Synapse check the results.

The screenshot shows the Microsoft Azure Synapse Analytics interface. The top navigation bar includes 'Microsoft Azure', 'Synapse Analytics', and 'synapse729'. A search bar is located on the right. The left sidebar shows the 'Data' section with 'Workspace' and 'Linked' tabs. Under 'Workspace', there is a 'Filter resources by name' input field. The 'SQL database' section is expanded, showing 'sqlpool (SQL)' and 'Tables'. Under 'Tables', 'dbo.Orders' is selected, and its 'Columns' are listed: 'OrderID (varchar(10), ...)', 'Quantity (int, null)', 'UnitPrice (decimal(5,2), ...)', and 'DiscountCategory (var...)'. The 'Results' tab is active, displaying a table with the following data:

OrderID	Quantity	UnitPrice	DiscountCateg...
2	10	200.00	Category 2
2	10	200.00	Category 2
1	5	120.00	Category 1
4	20	400.00	Category 4
5	25	500.00	Category 5
1	5	120.00	Category 1
4	20	400.00	Category 4
4	20	400.00	Category 4
3	15	300.00	Category 3

The SQL query editor shows the following query:

```
1 select * from Orders;
```

The 'Results' tab is active, and the 'Table' view is selected. The 'Export results' button is visible.

# License



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#).