

Lab 10 - Real-time stream processing with Stream Analytics

In this lab, you will learn how to process streaming data with Azure Stream Analytics. You will ingest vehicle telemetry data into Event Hubs, then process that data in real time, using various windowing functions in Azure Stream Analytics. They will output the data to Azure Synapse Analytics. Finally, you will learn how to scale the Stream Analytics job to increase throughput.

After completing this lab, you will be able to:

- Use Stream Analytics to process real-time data from Event Hubs
- Use Stream Analytics windowing functions to build aggregates and output to Synapse Analytics
- Scale the Azure Stream Analytics job to increase throughput through partitioning
- Repartition the stream input to optimize parallelization

Technology overview

Azure Stream Analytics

As more and more data is generated from a variety of connected devices and sensors, transforming this data into actionable insights and predictions in near real-time is now an operational necessity. [Azure Stream Analytics](#) seamlessly integrates with your real-time application architecture to enable powerful, real-time analytics on your data no matter what the volume.

Azure Stream Analytics enables you to develop massively parallel Complex Event Processing (CEP) pipelines with simplicity. It allows you to author powerful, real-time analytics solutions using very simple, declarative [SQL like language](#) with embedded support for temporal logic. Extensive array of [out-of-the-box connectors](#), advanced debugging and job monitoring capabilities help keep costs down by significantly lowering the developer skills required. Additionally, Azure Stream Analytics is highly extensible through support for custom code with [JavaScript User Defined functions](#) further extending the streaming logic written in SQL.

Getting started in seconds is easy with Azure Stream Analytics as there is no infrastructure to worry about, and no servers, virtual machines, or clusters to manage. You can instantly [scale-out the processing power](#) from one to hundreds of streaming units for any job. You only pay for the processing used per job.

[Guaranteed event delivery](#) and an enterprise grade SLA, provide the three 9's of availability, making sure that Azure Stream Analytics is suitable for mission critical workloads. Automated checkpoints enable fault tolerant operation with fast restarts with no data loss.

Azure Event Hubs

[Azure Event Hubs](#) is a big data pipeline that can ingest millions of events per second. It facilitates the capture, retention, and replay of telemetry and event stream data, using standard protocols such as HTTPS, AMQP, AMQP over websockets, and Kafka. The data can come from many concurrent sources and up to 20 consumer groups can allow applications to read entire event hub independently at their own pace.

Scenario overview

Contoso Auto is collecting vehicle telemetry and wants to use Event Hubs to rapidly ingest and store the data in its raw form, then do some processing in near real-time. In the end, they want to create a dashboard that automatically updates with new data as it flows in after being processed. What they would like to see on the dashboard are various visualizations of detected anomalies, like engines overheating, abnormal oil pressure, and aggressive driving, using components such as a map to show anomalies related to cities, as well as various charts and graphs depicting this information in a clear way.

In this experience, you will use Azure Event Hubs to ingest streaming vehicle telemetry data as the entry point to a near real-time analytics pipeline built on Event Hubs, Azure Stream Analytics, and Azure Synapse Analytics. Azure Stream Analytics extracts the vehicle sensor data from Event Hubs, performs aggregations over windows of time, then sends the aggregated data to Azure Synapse Analytics for data analysis. A vehicle telemetry data generator will be used to send vehicle telemetry data to Event Hubs.

Lab setup and pre-requisites

Before starting this lab, you must complete at least the setup steps in **Lab 4: Explore, transform, and load data into the Data Warehouse using Apache Spark**.

This lab uses the dedicated SQL pool you created in the previous lab. You should have paused the SQL pool at the end of the previous lab, so resume it by following these instructions:

1. Open Azure Synapse Studio (<https://web.azuresynapse.net/>).
2. Select the **Manage** hub.
3. Select **SQL pools** in the left-hand menu. If the **SQLPool01** dedicated SQL pool is paused, hover over its name and select ▷.

The screenshot shows the Azure Synapse Studio interface. On the left, the 'Analytics pools' menu is open, and 'SQL pools' is selected, marked with a red box and a red circle with the number 1. The main area displays the 'SQL pools' section. It includes a 'New' button, a 'Refresh' button, and a 'System-assigned managed identity' toggle. Below this is a 'Filter by name' search bar. A message states 'Showing 1-2 of 2 items (1 Serverless, 1 Dedicated)'. A table lists the pools:

Name	Type	Status	Size
Built-in	Serverless	Online	Auto
SQLPool01	Dedicated	Paused	DW100c

For the 'SQLPool01' row, a context menu is open, showing a 'Resume' button (a right-pointing triangle) highlighted with a red box and a red circle with the number 2. The 'Paused' status in the 'Status' column for 'SQLPool01' is also highlighted with a red box.

4. When prompted, select **Resume**. It will take a minute or two to resume the pool.
5. Continue to the next exercise while the dedicated SQL pool resumes.

Important: Once started, a dedicated SQL pool consumes credits in your Azure subscription until it is paused. If you take a break from this lab, or decide not to complete it; follow the instructions at the end of the lab to pause your SQL pool!






Exercise 1 - Configure services

Azure Event Hubs is a Big Data streaming platform and event ingestion service, capable of receiving and processing millions of events per second. We are using it to temporarily store vehicle telemetry data that is processed and ready to be sent to the real-time dashboard. As data flows into Event Hubs, Azure Stream Analytics will query the data, applying aggregates and tagging anomalies, then send it to Azure Synapse Analytics.

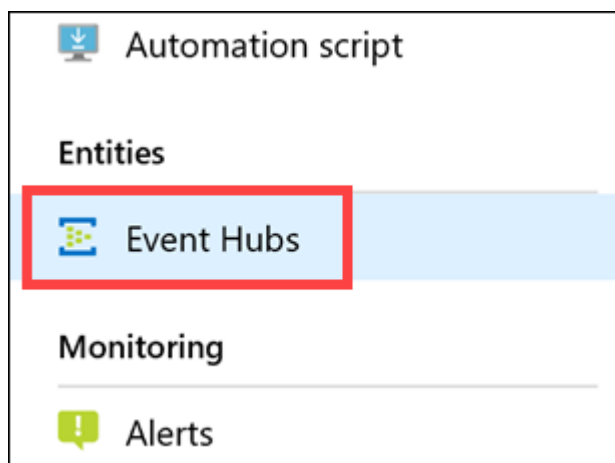
Task 1: Configure Event Hubs

In this task, you will create and configure a new event hub within the provided Event Hubs namespace. This will be used to capture vehicle telemetry after it has been processed and enriched by the Azure function you will create later on.

1. Browse to the [Azure portal](#).
2. Select **Resource groups** in the left-hand menu. Then select the **data-engineering-synapse-xxxxxxx** resource group.
3. Select the **eventhubxxxxxxx** Event Hubs Namespace.

<input type="checkbox"/> Name ↑↓	Type ↑↓
<input type="checkbox"/>  asadatalakede44	Storage account
<input type="checkbox"/>  asade44	Stream Analytics job
<input type="checkbox"/>  asaworkspacede44	Synapse workspace
<input type="checkbox"/>  ContosoAuto (asaworkspacede44/ContosoAuto)	Dedicated SQL pool
<input type="checkbox"/>  eventhubde44	Event Hubs Namespace

4. In the Event Hubs Namespace blade, select **Event Hubs** in the left-hand menu.



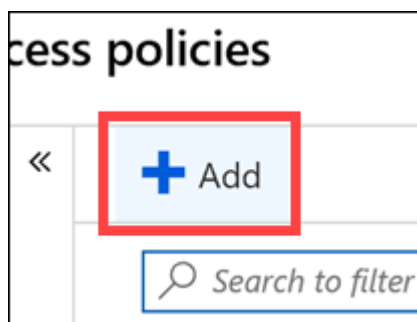
5. Select the **telemetry** event hub.

+ Event Hub Refresh	
<input type="text" value="Search to filter items..."/>	
NAME	STATUS
telemetry	Active

6. Select **Shared access policies** in the left-hand menu.

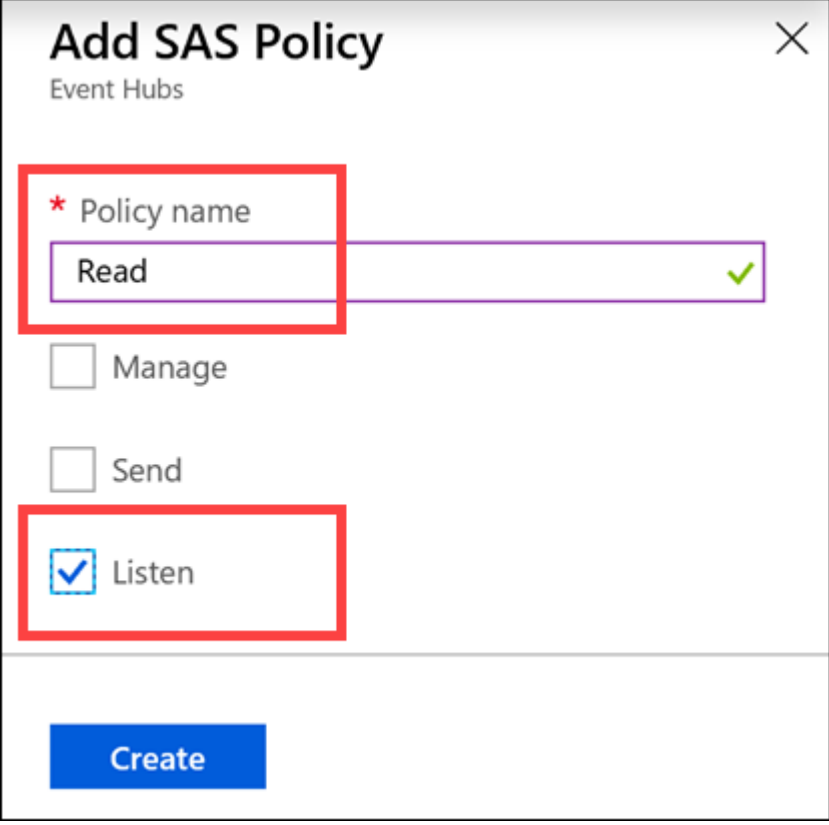


7. Select + **Add** in the top toolbar to create a new shared access policy.



8. In the **Add SAS Policy** blade, configure the following:

- **Name:** Read
- **Managed:** Unchecked
- **Send:** Unchecked
- **Listen:** Checked



Add SAS Policy ×

Event Hubs

* Policy name

Read ✓

☐ Manage

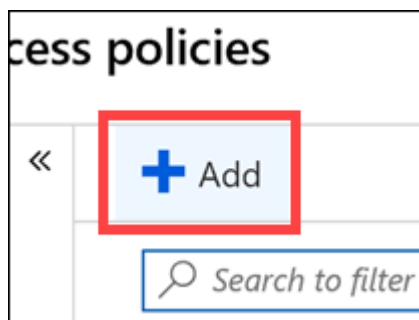
☐ Send

☒ Listen

Create

It is a best practice to create separate policies for reading, writing, and managing events. This follows the principle of least privilege to prevent services and applications from performing unauthorized operations.

9. Select **Create** on the bottom of the form when you are finished entering the values.
10. Select **+ Add** in the top toolbar to create a second new shared access policy.



11. In the **Add SAS Policy** blade, configure the following:

- **Name:** Write
- **Managed:** Unchecked
- **Send:** Checked
- **Listen:** Unchecked

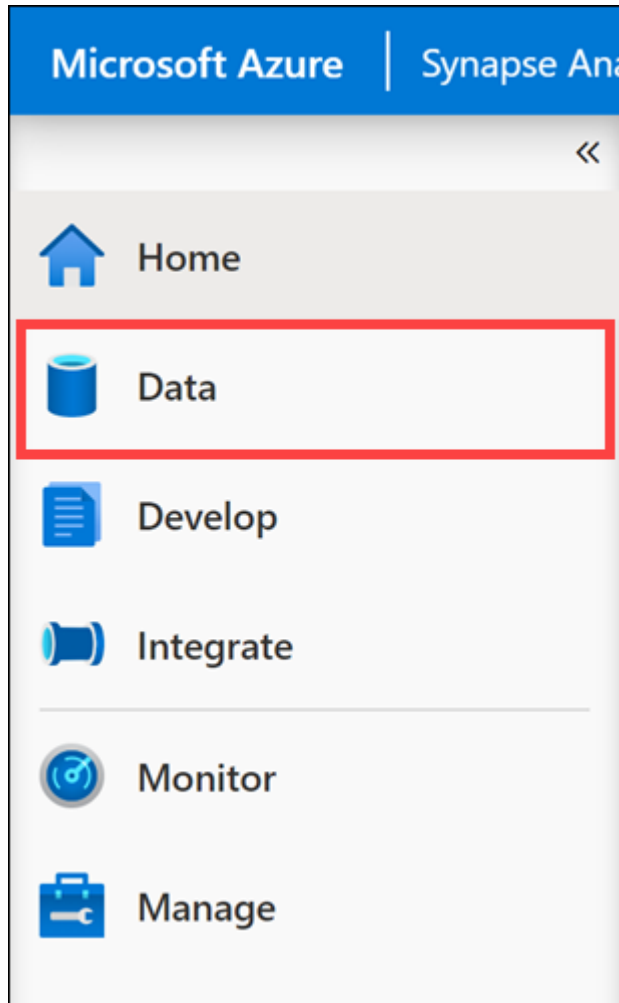
12. Select **Create** on the bottom of the form when you are finished entering the values.
13. Select your **Write** policy from the list. Copy the **Connection string - primary key** value by selecting the Copy button to the right of the field. Save this value in Notepad or similar text editor for later.

Task 2: Configure Synapse Analytics

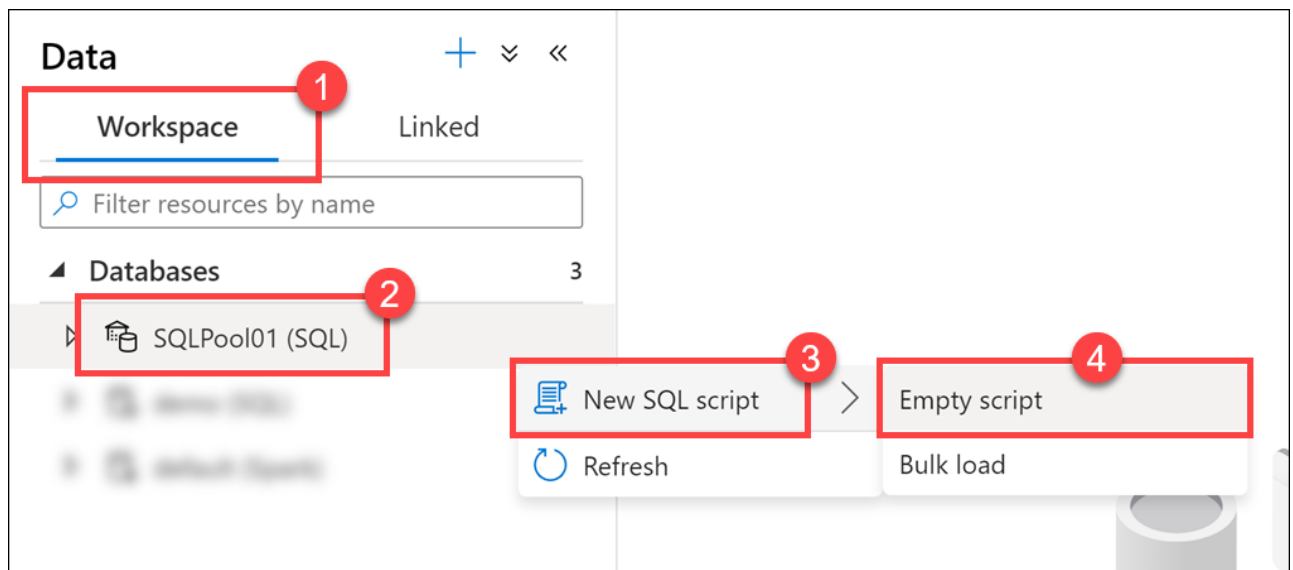
Azure Synapse is an end-to-end analytics platform which combines SQL data warehousing, big data analytics, and data integration into a single integrated environment. It empowers users to gain quick access and insights across all of their data, enabling a whole new level of performance and scale that is simply unmatched in the industry.

In this task, you will create a table in a Synapse dedicated SQL pool to store aggregate vehicle data provided by a Stream Analytics job that processes vehicle telemetry ingested by Event Hubs.

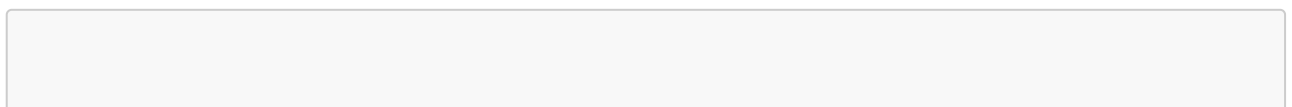
1. In the Azure Synapse Studio, select the **Data** hub.



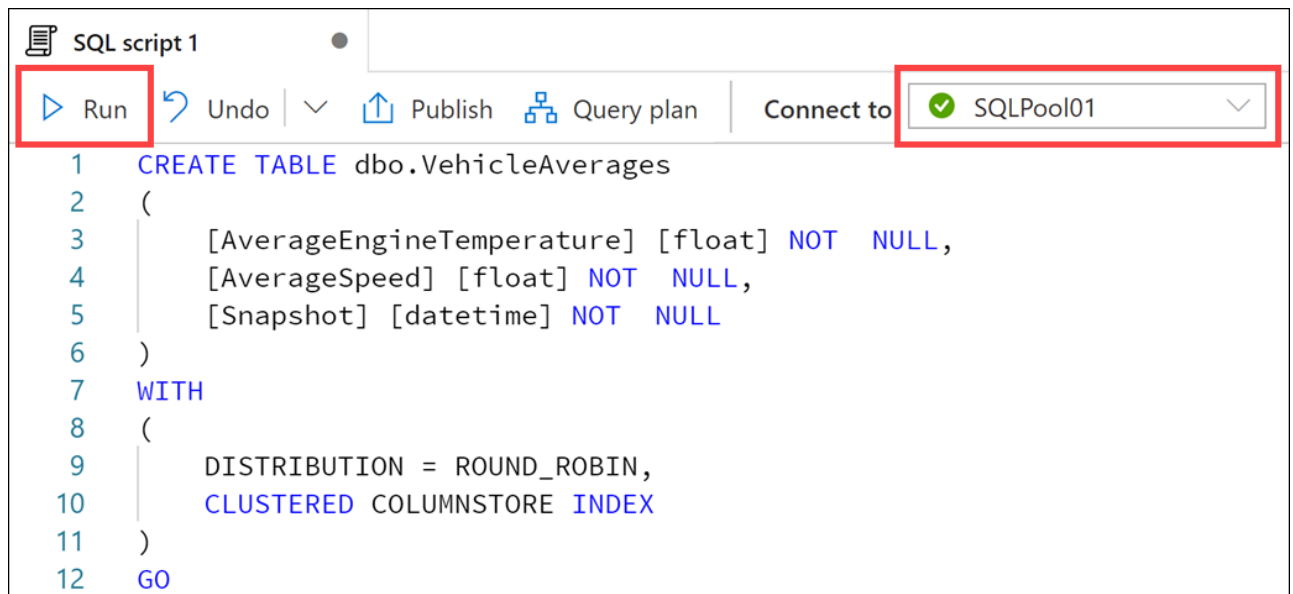
2. Select the **Workspace** tab, expand the **SQL database group** and right-click **SQLPool01**. Then select **New SQL script**, and select **Empty script**.



3. Make sure the script is connected to **SQLPool01**, then replace the script with the following and select **Run** to create a new table:



```
CREATE TABLE dbo.VehicleAverages
(
    [AverageEngineTemperature] [float] NOT NULL,
    [AverageSpeed] [float] NOT NULL,
    [Snapshot] [datetime] NOT NULL
)
WITH
(
    DISTRIBUTION = ROUND_ROBIN,
    CLUSTERED COLUMNSTORE INDEX
)
GO
```




Task 3: Configure Stream Analytics

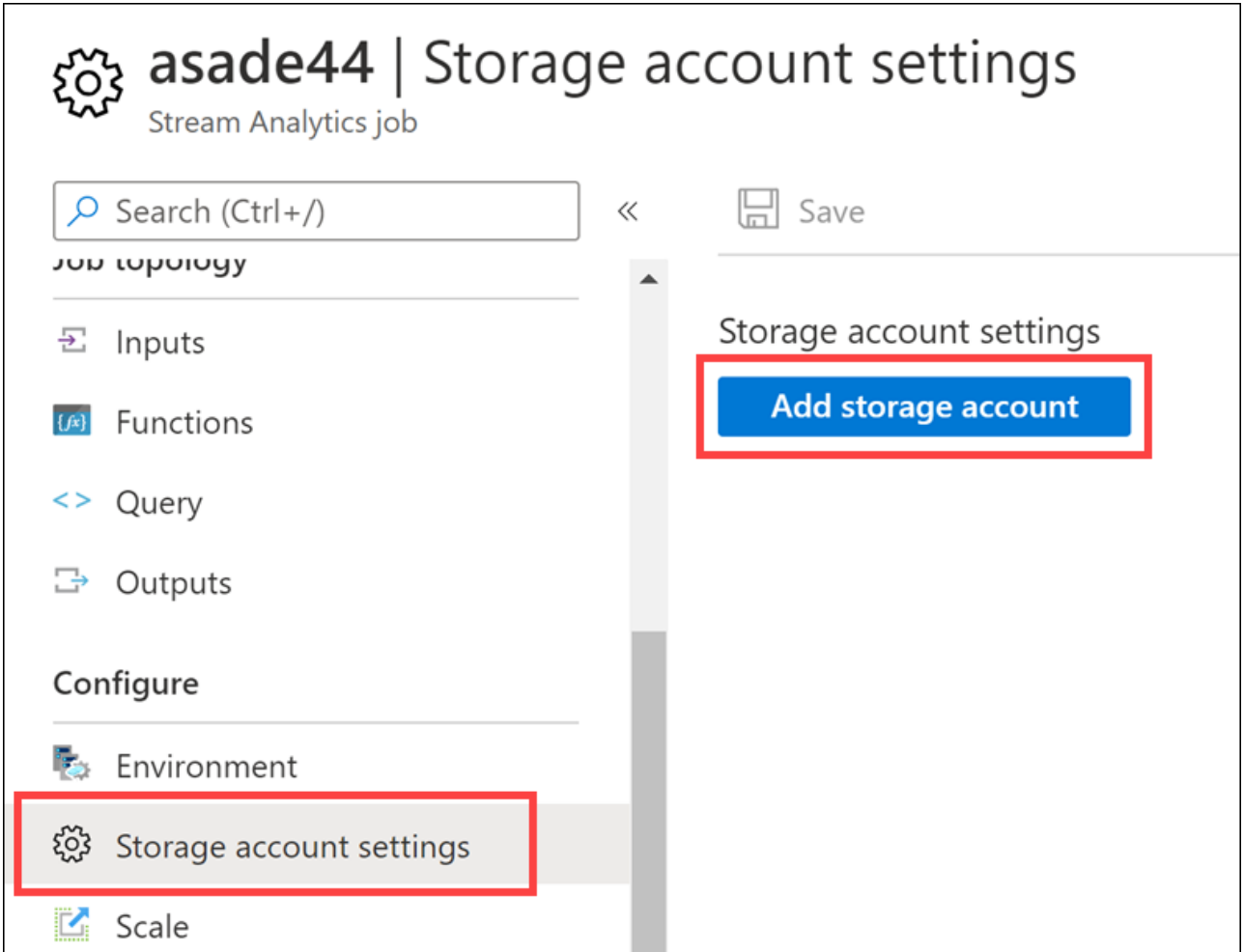
Azure Stream Analytics is an event-processing engine that allows you to examine high volumes of data streaming from devices. Incoming data can be from devices, sensors, web sites, social media feeds, applications, and more. It also supports extracting information from data streams, identifying patterns, and relationships. You can then use these patterns to trigger other actions downstream, such as create alerts, feed information to a reporting tool, or store it for later use.

In this task, you will configure Stream Analytics to use the event hub you created as a source, query and analyze that data.

1. In the Azure portal, in the **data-engineering-synapse-xxxxxxx** resource group, select the **asxxxxxxx** Stream Analytics job.

<input type="checkbox"/> Name ↑↓	Type ↑↓
<input type="checkbox"/>  asadatalakejdh20210617	Storage account
<input type="checkbox"/>  asajdh20210617	Stream Analytics job
<input type="checkbox"/>  asakeyvaultjdh20210617	Key vault
<input type="checkbox"/>  asastorejdh20210617	Storage account
<input type="checkbox"/>  asaworkspacejdh20210617	Synapse workspace

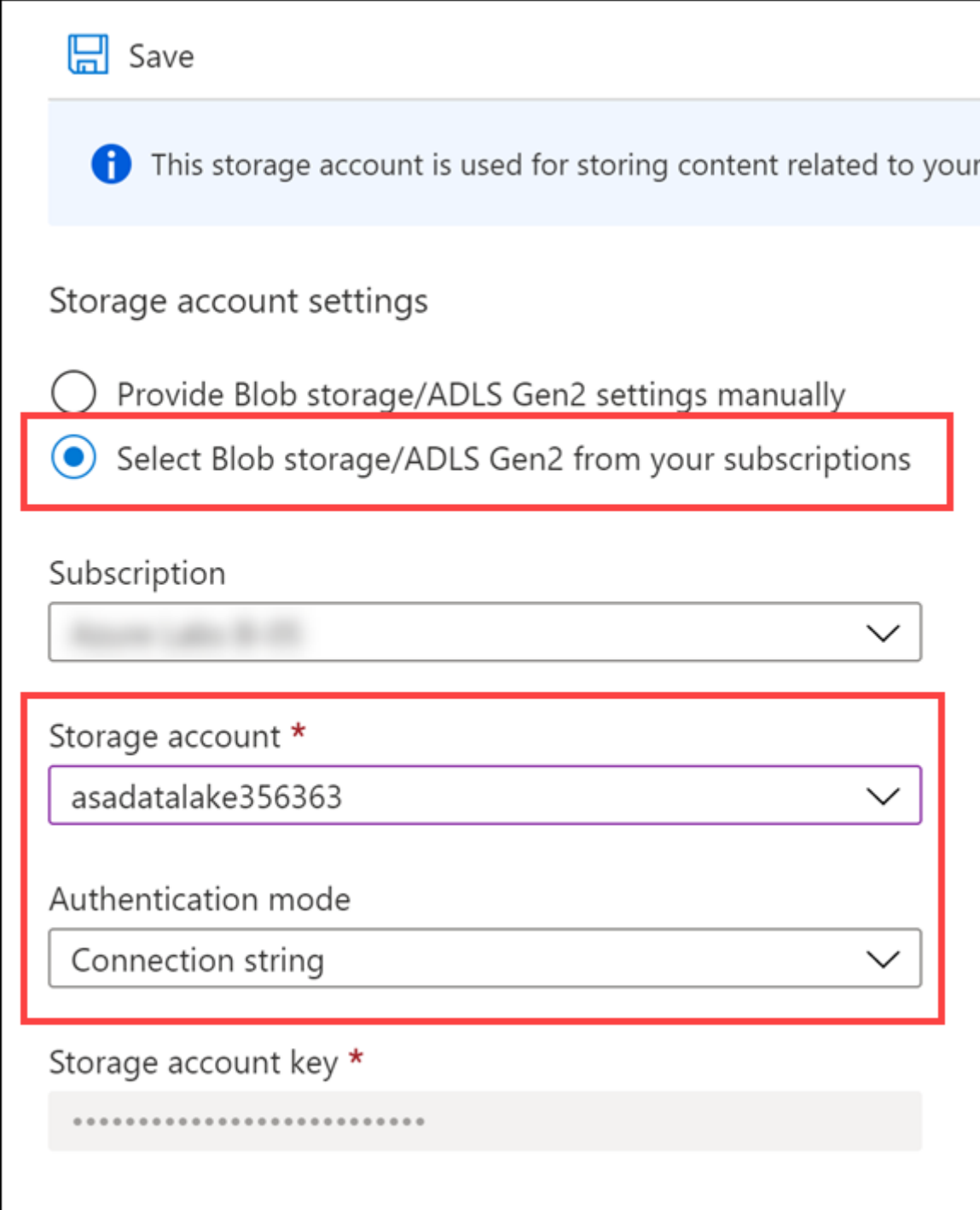
2. Within the Stream Analytics job, select **Storage account settings** in the left-hand menu, then select **Add storage account**. Since we will use Synapse Analytics as one of the outputs, we need to first configure the job storage account.



The screenshot shows the 'asade44 | Storage account settings' page for a Stream Analytics job. The left-hand navigation menu includes 'Inputs', 'Functions', 'Query', 'Outputs', 'Configure', 'Environment', 'Storage account settings' (highlighted with a red box), and 'Scale'. The main content area displays 'Storage account settings' with a red box around the 'Add storage account' button. A search bar and a 'Save' button are also visible at the top of the main content area.

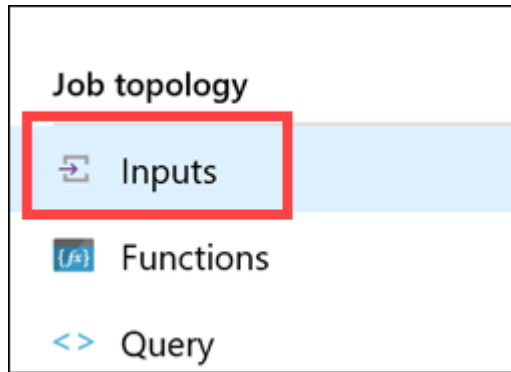
3. In the **Storage account settings** form, configure the following:
- **Select storage account from your subscriptions:** Selected.
 - **Subscription:** Make sure the subscription you are using for this lab is selected.
 - **Storage account:** Select the storage account named **asadatalakexxxxxxx**.

- **Authentication mode:** Select "Connection string".

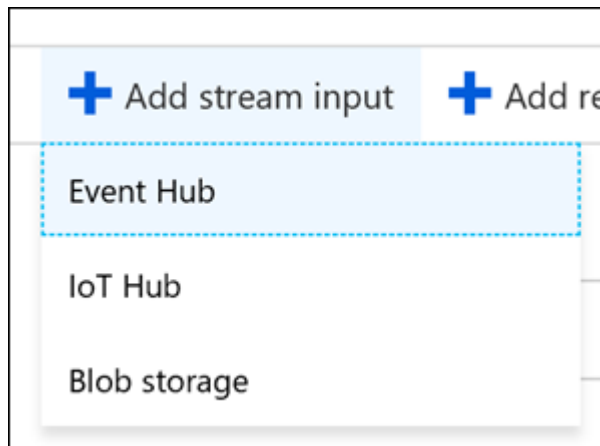


The screenshot shows the 'Storage account settings' configuration page in the Azure portal. At the top, there is a 'Save' button with a floppy disk icon. Below it is a light blue information banner with an 'i' icon and the text: 'This storage account is used for storing content related to your'. The main section is titled 'Storage account settings'. It contains two radio button options: 'Provide Blob storage/ADLS Gen2 settings manually' (unselected) and 'Select Blob storage/ADLS Gen2 from your subscriptions' (selected). The selected option is highlighted with a red rectangular box. Below the radio buttons is a 'Subscription' dropdown menu showing a blurred selection. Another red rectangular box highlights the 'Storage account' dropdown menu, which is set to 'asadatalake356363', and the 'Authentication mode' dropdown menu, which is set to 'Connection string'. Below these, there is a 'Storage account key' field with a red asterisk, which is currently masked with dots.

4. Select **Save**, then **Yes** when prompted to save the storage account settings.
5. Within the Stream Analytics job, select **Inputs** within the left-hand menu.



6. Select + **Add stream input** in the top toolbar, then select **Event Hub** to create a new Event Hub input.



7. In the **New Input** blade, configure the following:

- **Name:** `eventhub`
- **Select Event Hub from your subscriptions:** Selected
- **Subscription:** Make sure the subscription you are using for this lab is selected.
- **Event Hub namespace:** Select the `eventhubxxxxxxx` Event Hub namespace.
- **Event Hub name:** Select **Use existing**, then select `telemetry`, which you created earlier.
- **Event Hub consumer group:** Select **Use existing**, then select `$Default`.
- **Authentication mode:** Select **Connection string**.
- **Event Hub policy name:** Select **Use existing**, then select **Read**.
- Leave all other values at their defaults.



☐ Provide Event Hub settings manually

☒ Select Event Hub from your subscriptions

Subscription

Subscription Analytics Services and Tools

Event Hub namespace * ⓘ

eventhubde44

Event Hub name * ⓘ

☐ Create new ☒ Use existing

telemetry

Event Hub consumer group * ⓘ

☐ Create new ☒ Use existing

\$Default

Authentication mode

Connection string

Event Hub policy name * ⓘ

☐ Create new ☒ Use existing

Read

Event Hub policy key

.....

Event serialization format * ⓘ

JSON



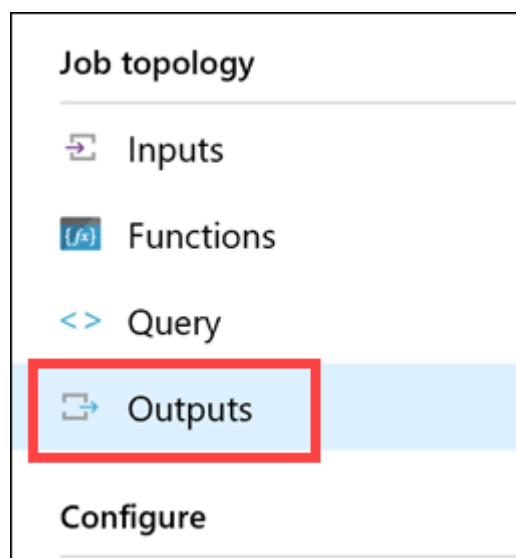
Encoding ⓘ

UTF-8

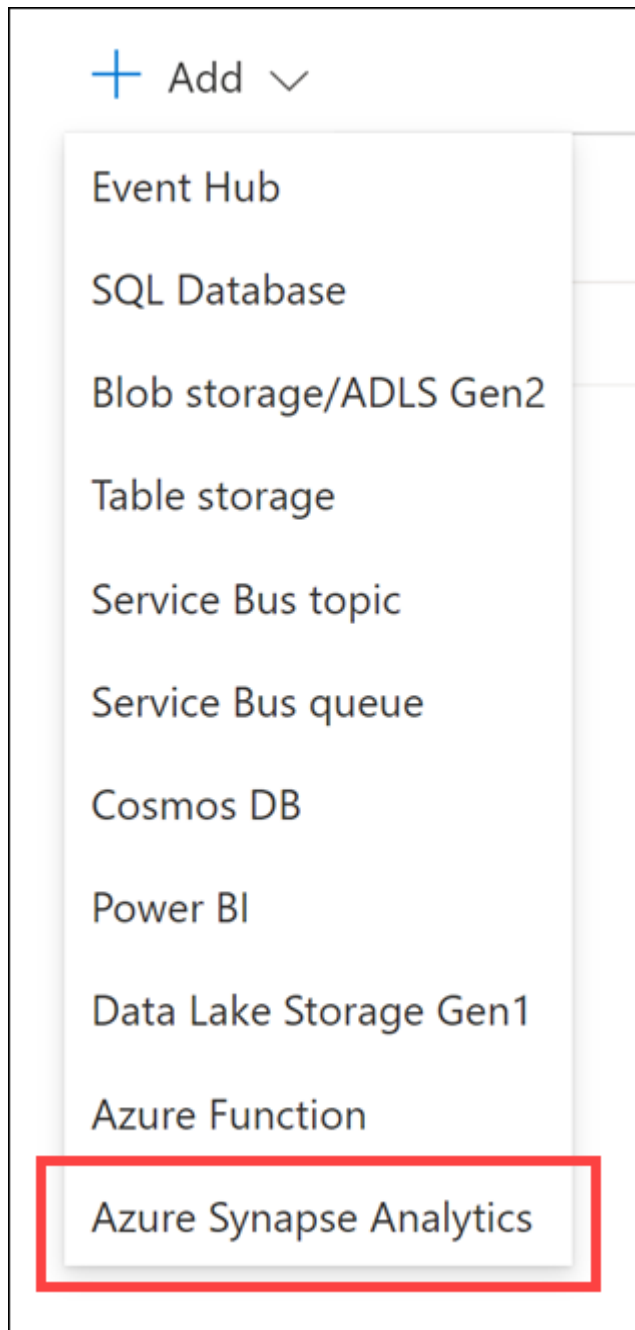
Event compression type ⓘ

None

8. Select **Save** on the bottom of the form when you are finished entering the values.
9. Within the Stream Analytics job blade, select **Outputs** within the left-hand menu.



10. Select **+ Add** in the top toolbar, then select **Azure Synapse Analytics** to create a new Synapse Analytics output.



11. In the **New Output** blade, configure the following:

- **Output alias:** `synapse`
- **Select Azure Synapse Analytics from your subscriptions:** Selected.
- **Subscription:** Select the subscription you are using for this lab.
- **Database:** Select **SQLPool01**. Make sure your correct Synapse workspace name appears under **Server name**.
- **Authentication mode:** Select **Connection string**.
- **Username:** `asa.sql.admin`
- **Password:** Enter the SQL admin password value you entered when deploying the lab environment, or which was provided to you as part of your hosted lab environment. If you are

unsure about your SQL admin username, navigate to the Synapse workspace in the Azure resource group. The SQL admin username is shown in the Overview pane.

- **Server name:** asaworkspacexxxxxxx
- **Table:** `dbo.VehicleAverages`

Azure Synapse Analytics

×

New output

Output alias *

synapse

☐

 Provide SQL Database settings manually

☒

 Select SQL Database from your subscriptions

Subscription

Synapse Analytics Demos and Labs

Database *

SQLPool01

Authentication mode

Connection string

Username *

asa.sql.admin

Password *

.....

Server name

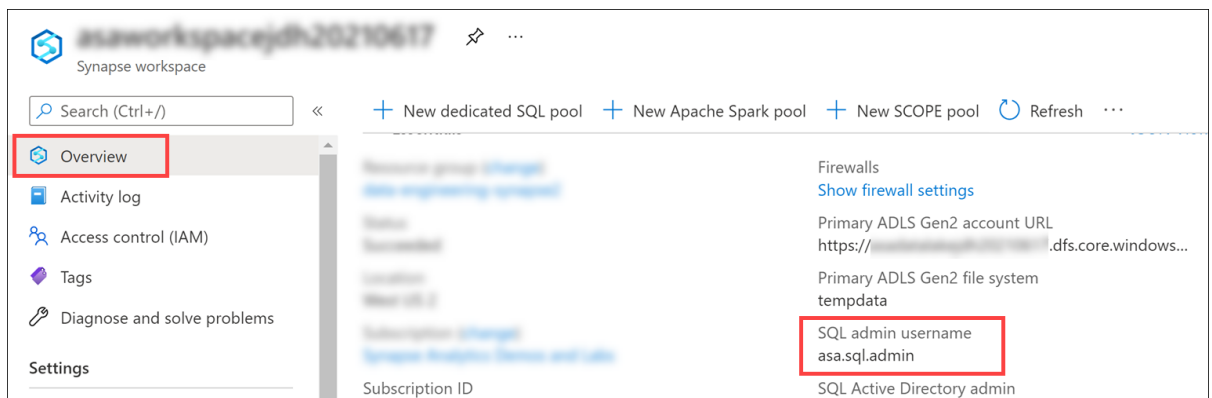
Server name

asagaworkspace01

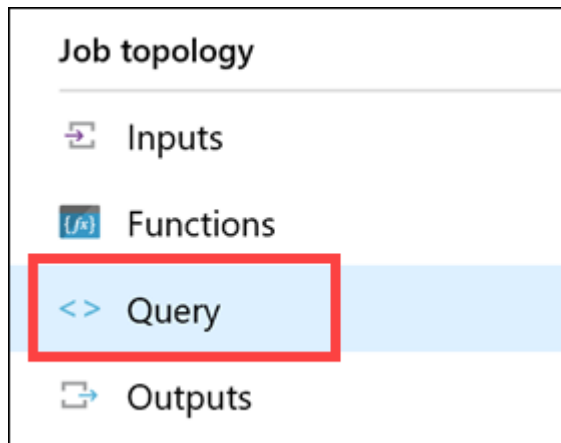
Table *

dbo.VehicleAverages ✓

Note: If you are unsure about your SQL admin username, navigate to the Synapse workspace in the Azure resource group. The SQL admin username is shown in the Overview pane.



12. Select **Save** on the bottom of the form when you are finished entering the values.
13. Within the Stream Analytics job blade, select **Query** within the left-hand menu.



14. Enter the following query:

```
WITH
VehicleAverages AS (
  select
    AVG(engineTemperature) averageEngineTemperature,
    AVG(speed) averageSpeed,
    System.Timestamp() as snapshot
  FROM
    eventhub TIMESTAMP BY [timestamp]
```



```

GROUP BY
    TumblingWindow(Duration(minute, 2))
)
-- INSERT INTO SYNAPSE ANALYTICS
SELECT
    *
INTO
    synapse
FROM
    VehicleAverages

```

Query language docs ▾ Open in Visual Studio ▾ UserVoice

Inputs (1)
eventhub

Outputs (1)
synapse

Test query Save query Discard changes

```

1 WITH
2 VehicleAverages AS (
3     select
4         AVG(engineTemperature) averageEngineTemperature,
5         AVG(speed) averageSpeed,
6         System.Timestamp() as snapshot
7     FROM
8         eventhub TIMESTAMP BY [timestamp]
9     GROUP BY
10        TumblingWindow(Duration(minute, 2))
11 )
12 -- INSERT INTO SYNAPSE ANALYTICS
13 SELECT
14     *
15 INTO
16     synapse
17 FROM
18     VehicleAverages
19

```

The query aggregates the average engine temperature and speed of all vehicles over the past two minutes, using **TumblingWindow(Duration(minute, 2))**, and outputs these fields to the **synapse** output.

15. Select **Save query** in the top toolbar when you are finished updating the query.
16. Within the Stream Analytics job blade, select **Overview** within the left-hand menu. On top of the Overview blade, select **Start**.

asade44
Stream Analytics job

Search (Ctrl+/) << Start Stop Delete

Overview

Activity log

Access control (IAM)

Tags

Created

Essentials

Resource group (change) : ms-dataengineering-14

Status : Created

17. In the Start job blade that appears, select **Now** for the job output start time, then select **Start**. This will start the Stream Analytics job so it will be ready to start processing and sending your events to Azure Synapse Analytics.

Start job ×

asade44

Streaming units ⓘ
3

Environment ⓘ
Standard

Job output start time ⓘ
Now Custom

Start

Exercise 2 - Generate and aggregate data

Task 1: Run data generator

The data generator console application creates and sends simulated vehicle sensor telemetry for an array of vehicles (denoted by VIN (vehicle identification number)) directly to Event Hubs. For this to happen, you first need to configure it with the Event Hub connection string.

In this task, you will configure and run the data generator. The data generator saves simulated vehicle telemetry data to Event Hubs, prompting your Stream Analytics job to aggregate and analyze the enriched data and send it to Synapse Analytics.

1. On your lab VM, use Windows Explorer to view the **c:\dp-203\data-engineering-ilt-deployment\Allfiles** folder.
2. Extract the **TransactionGenerator.zip** archive to a subfolder named **TransactionGenerator**.
3. In the extracted **TransactionGenerator** folder, open the **appsettings.json** file. Paste your **telemetry** Event Hub connection string value next to **EVENT_HUB_CONNECTION_STRING**. Make sure you have

quotes ("") around the value, as shown. **Save** the file.

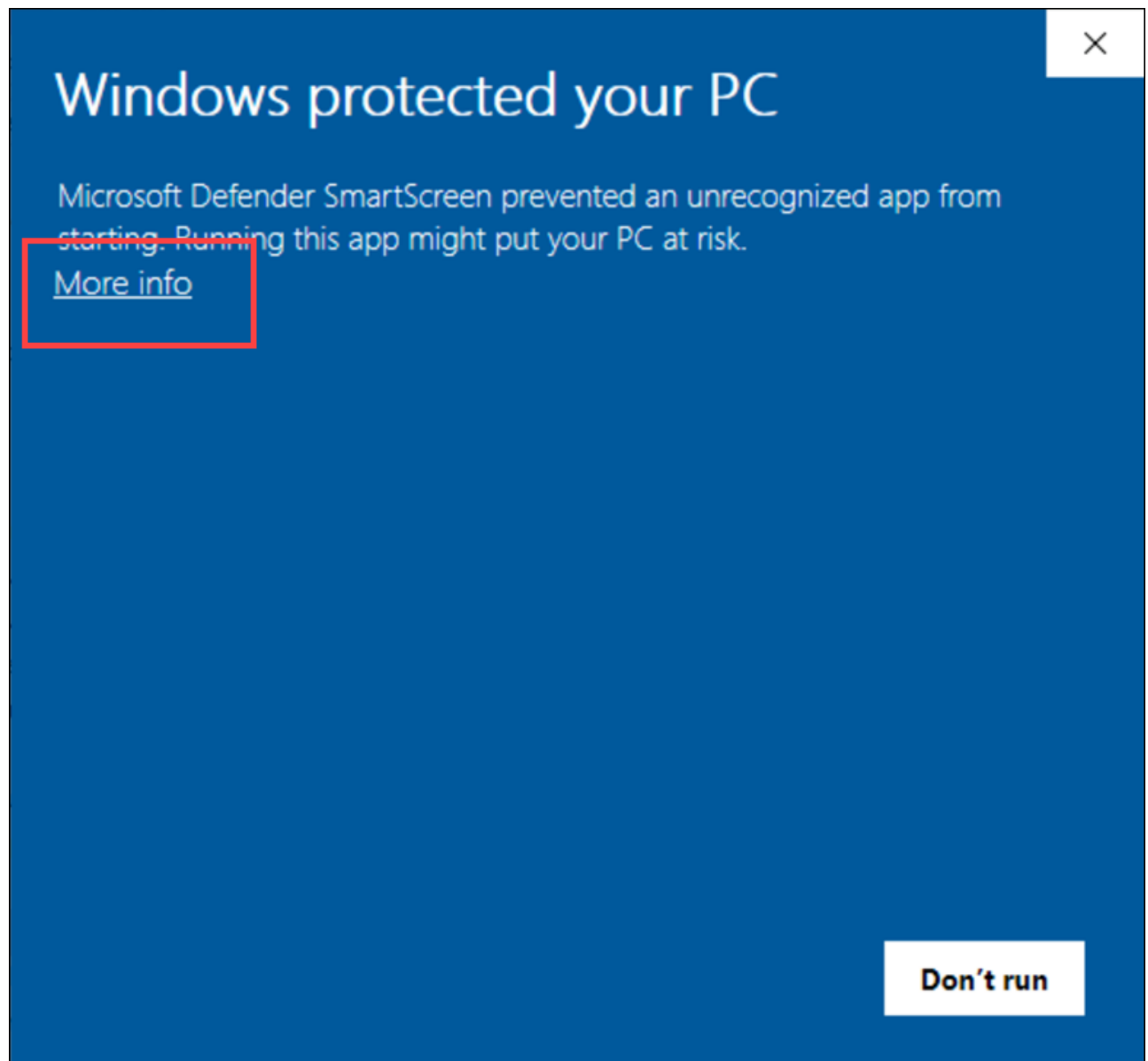
```
1 {
2   "EVENT_HUB_CONNECTION_STRING": "Endpoint=sb://eventhubde44.servicebus.windows.net/;SharedAccessKeyName=Write;SharedAccessKey=0u52KB/R1KGY/avjytFMg7F2s",
3
4   "SECONDS_TO_LEAD": "0",
5   "SECONDS_TO_RUN": "1800"
6 }
7
```

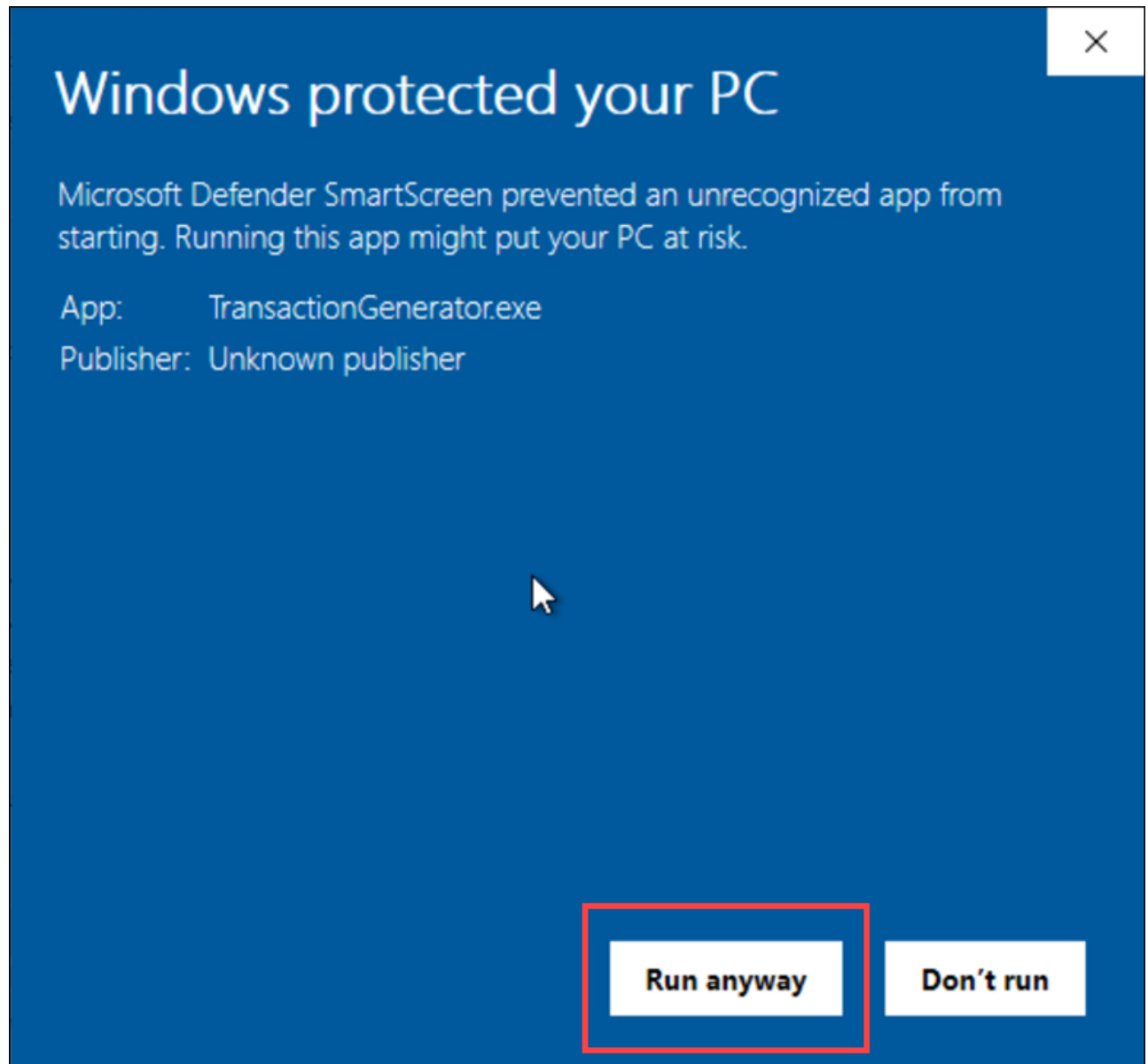
Note: Make sure that the connection string ends with *EntityPath=telemetry* (eg. *Endpoint=sb://YOUR_EVENTHUB_NAMESPACE.servicebus.windows.net/;SharedAccessKeyName=Write;SharedAccessKey=REDACTED/S/U=;EntityPath=telemetry*). If not, then you did not copy the connection string from the **Write** policy of your event hub.

SECONDS_TO_LEAD is the amount of time to wait before sending vehicle telemetry data. Default value is 0.

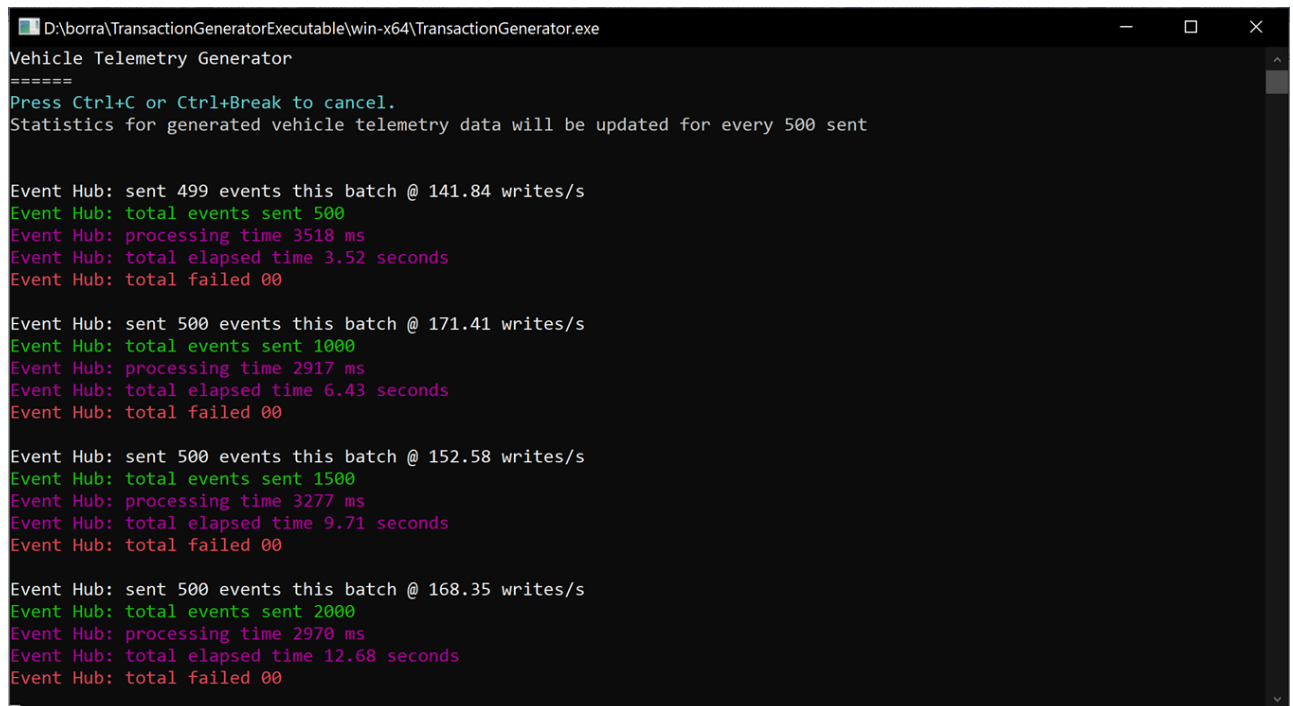
SECONDS_TO_RUN is the maximum amount of time to allow the generator to run before stopping transmission of data. The default value is 1800. Data will also stop transmitting when you enter Ctrl+C while the generator is running, or if you close the window.

4. In the extracted **TransactionGenerator** folder, run **TransactionGenerator.exe**.
5. If a **Windows protected your PC** dialog is displayed, select **More info**, then **Run anyway**.





6. A new console window will open, and you should see it start to send data after a few seconds. Once you see that it is sending data to Event Hubs, *minimize* the window and keep it running in the background. Allow this to run for a minimum of three minutes before moving onto the next step.



```
D:\borra\TransactionGeneratorExecutable\win-x64\TransactionGenerator.exe
Vehicle Telemetry Generator
=====
Press Ctrl+C or Ctrl+Break to cancel.
Statistics for generated vehicle telemetry data will be updated for every 500 sent

Event Hub: sent 499 events this batch @ 141.84 writes/s
Event Hub: total events sent 500
Event Hub: processing time 3518 ms
Event Hub: total elapsed time 3.52 seconds
Event Hub: total failed 00

Event Hub: sent 500 events this batch @ 171.41 writes/s
Event Hub: total events sent 1000
Event Hub: processing time 2917 ms
Event Hub: total elapsed time 6.43 seconds
Event Hub: total failed 00

Event Hub: sent 500 events this batch @ 152.58 writes/s
Event Hub: total events sent 1500
Event Hub: processing time 3277 ms
Event Hub: total elapsed time 9.71 seconds
Event Hub: total failed 00

Event Hub: sent 500 events this batch @ 168.35 writes/s
Event Hub: total events sent 2000
Event Hub: processing time 2970 ms
Event Hub: total elapsed time 12.68 seconds
Event Hub: total failed 00
```

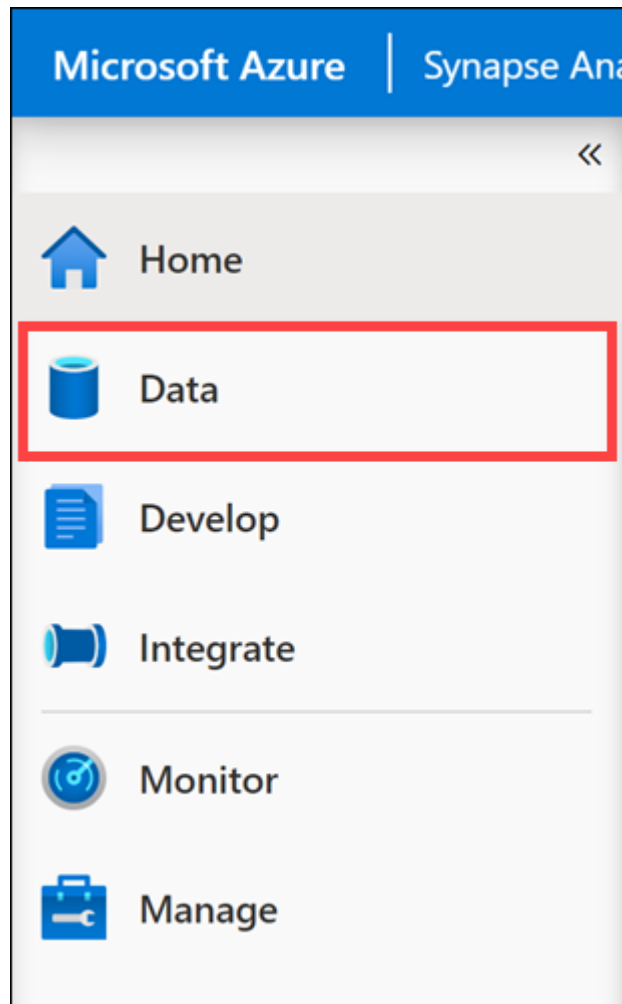
After every 500 records are requested to be sent, you will see output statistics.

Task 2: View aggregate data in Synapse Analytics

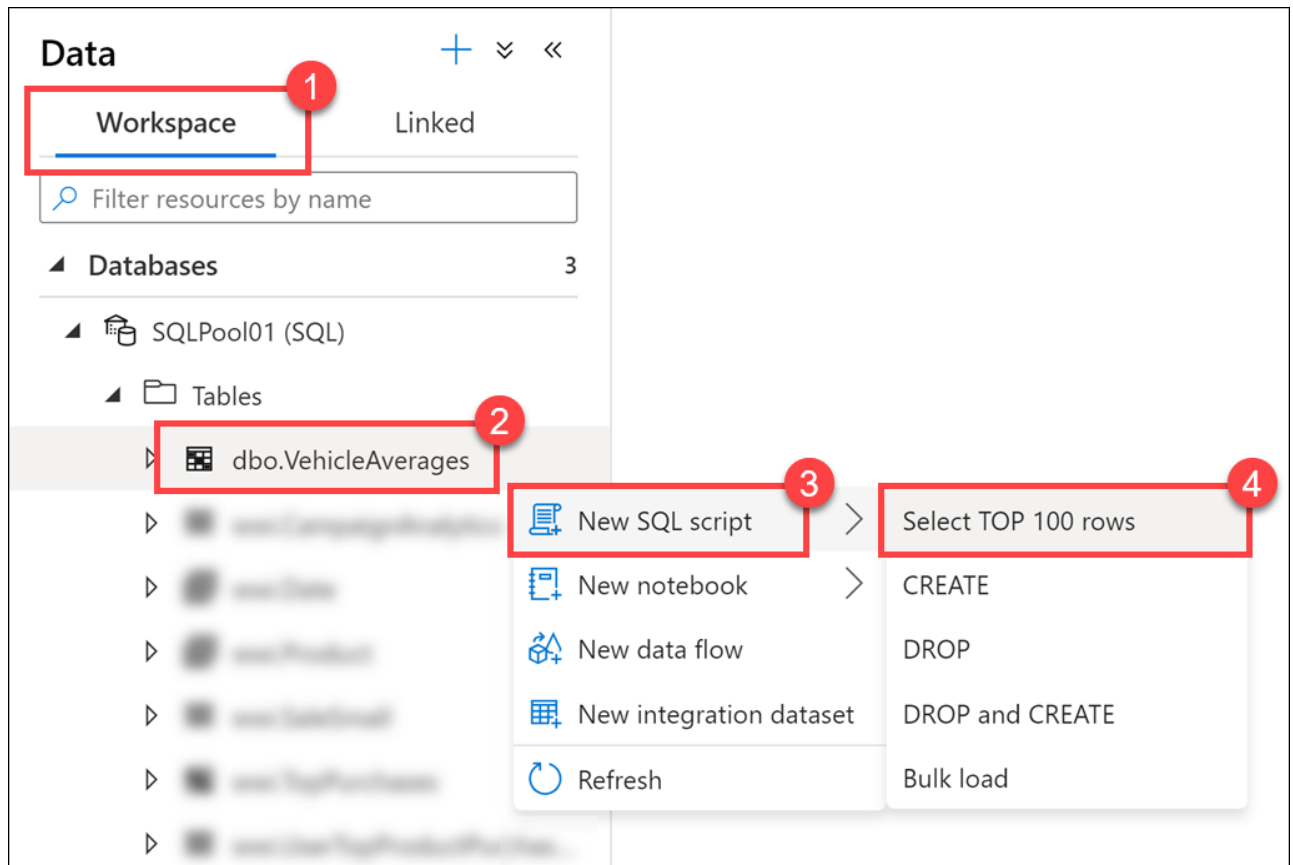
As you recall, when you created the query in Stream Analytics, you aggregated the engine temperature and vehicle speed data over two-minute intervals and saved the data to Synapse Analytics. This capability demonstrates the Stream Analytics query's ability to write data to multiple outputs at varying intervals. Writing to a Synapse Analytics dedicated SQL pool enables us to retain the historic and current aggregate data as part of the data warehouse without requiring an ETL/ELT process.

In this task, you will view the anomaly data within Synapse Analytics.

1. In Synapse Studio, select **Data** in the left-hand menu to navigate to the Data hub.



2. Select the **Workspace** tab, expand the **SQLPool01** database, expand **Tables**, then right-click on the **dbo.VehicleAverages** table (if you do not see the table listed, refresh the tables list). Select **New SQL script**, then **Select TOP 100 rows**.



3. Run the query and view the results. Observe the aggregate data stored in **AverageEngineTemperature** and **AverageSpeed**. The **Snapshot** value changes in two-minute intervals between these records.

SQL script 1

Run Undo Publish Query plan Connect to SQLPool01 Use database SQLPool01

```

1 SELECT TOP (100) [AverageEngineTemperature]
2 , [AverageSpeed]
3 , [Snapshot]
4 FROM [dbo].[VehicleAverages]

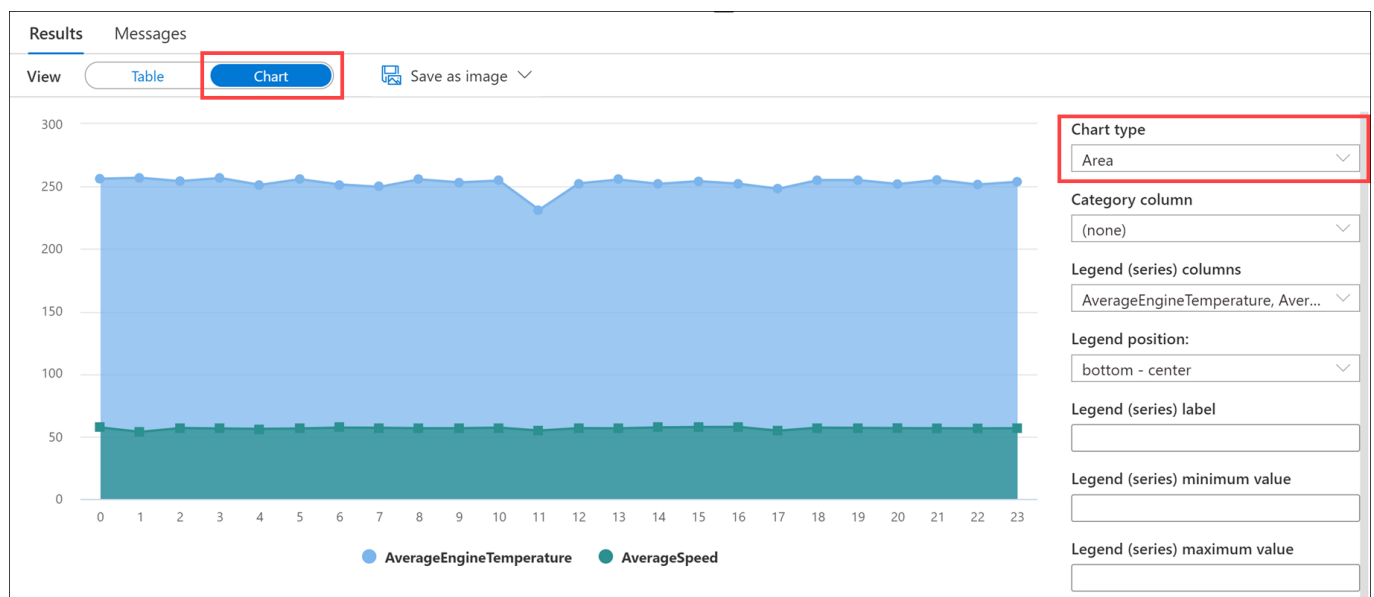
```

Results Messages

View Table Chart Export results

AverageEngineTemperature	AverageSpeed	Snapshot
253.471058209685	56.7696070438611	2021-06-23T04:14:00.0000000
252.195573770492	56.1544262295082	2021-06-23T04:10:00.0000000
251.669111518586	56.3575595932655	2021-06-23T04:12:00.0000000
252.580983606557	56.5922950819672	2021-06-23T04:20:00.0000000
252.751553810926	57.1414785737651	2021-06-23T04:24:00.0000000
252.394976037019	56.4805817220294	2021-06-23T04:22:00.0000000
252.20492011159	56.8582297742835	2021-06-23T04:08:00.0000000
257.897328881469	56.7337228714524	2021-06-23T04:16:00.0000000
251.311151375391	57.2698072805139	2021-06-23T04:18:00.0000000
254.0347739692	56.8137108792846	2021-06-23T04:26:00.0000000

4. Select the **Chart** view in the Results output, then set the chart type to **Area**. This visualization shows the average engine temperature correlated with the average speed over time. Feel free to experiment with the chart settings. The longer the data generator runs the more data points are generated. The following visualization is for an example of a session that ran over 10 mins, and may not represent what you see on the screen.



Important: Cleanup

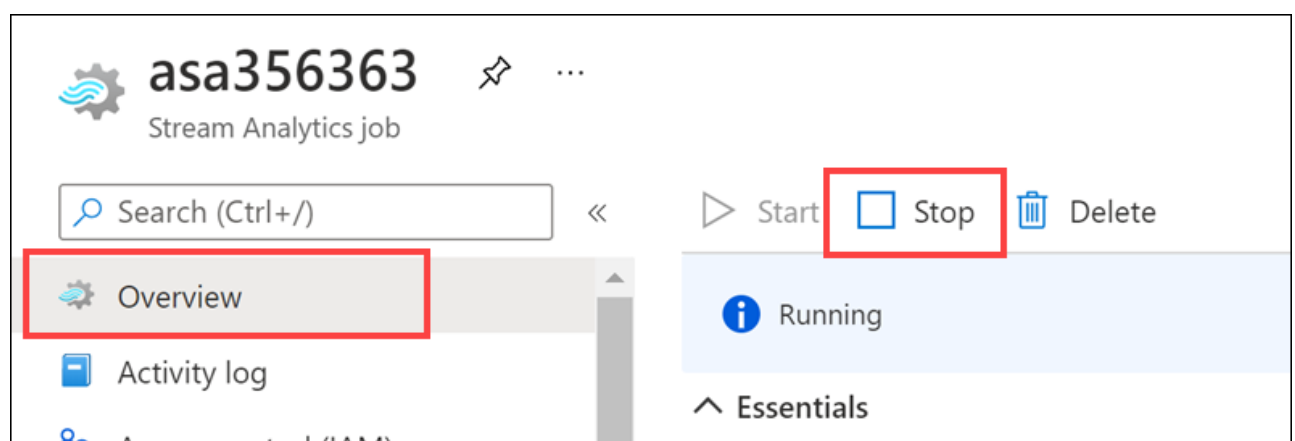
Complete these steps to stop the data generator and free up resources you no longer need.

Task 1: Stop the data generator

1. Go back to the console/terminal window in which your data generator is running. Close the window to stop the generator.

Task 2: Stop the Stream Analytics job

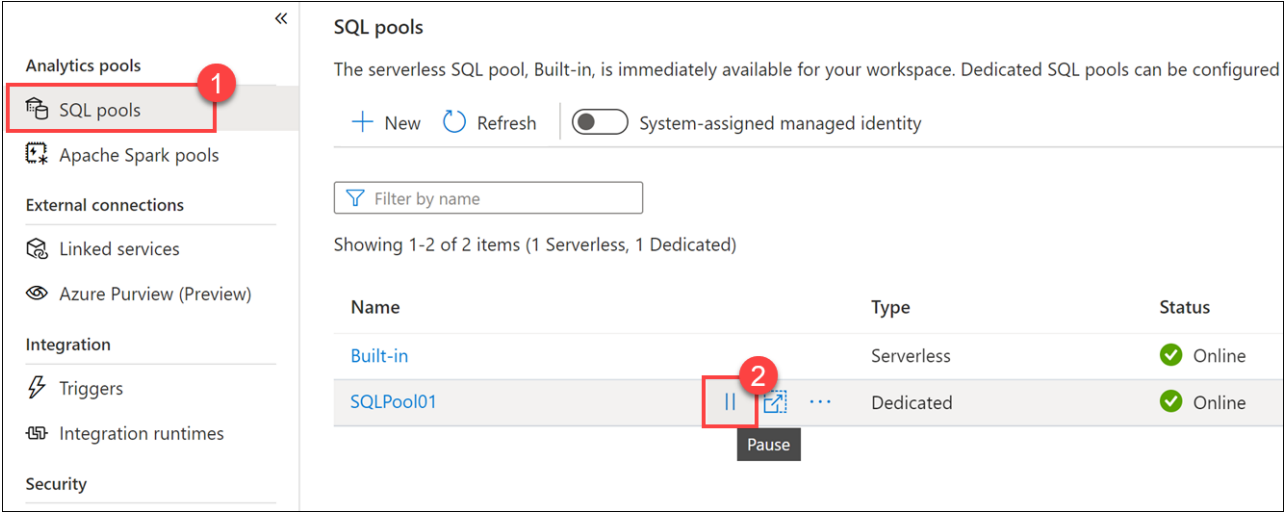
1. Navigate to the Stream Analytics job in the Azure portal.
2. In the Overview pane, select **Stop**, then select **Yes** when prompted.



Task 3: Pause the dedicated SQL pool

Complete these steps to free up resources you no longer need.

- 1. In Synapse Studio, select the **Manage** hub.
- 2. Select **SQL pools** in the left-hand menu. Hover over the **SQLPool01** dedicated SQL pool and select **||**.



- 3. When prompted, select **Pause**.