**Streaming Data Processing**

**Rohan Chaudhari**

([**chaudharirohan24@gmail.com**](mailto:chaudharirohan24@gmail.com))

**Vishal Jha**

([**vishaljhavj786@gmail.com**](mailto:vishaljhavj786@gmail.com))

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **S.no** | **Topic** | **Page No.** |
| 1 | Project Statement | 3 |
| 2 | Project Overview | 3 |
| 3 | Project Requirements | 3 |
| 3.1 | Azure Subscription | 3 |
| 3.2 | Data Sources | 3 |
| 3.3 | Azure Event Hub | 3 |
| 3.4 | Azure Stream Analytics Job | 3 |
| 3.5 | Azure Data Lake Storage | 3 |
| 3.6 | Azure Data Factory | 4 |
| 3.7 | Azure Databricks | 4 |
| 4 | Architecture Diagram | 4 |
| 5 | Execution Overview | 4 |
| 6 | Project Implementation | 5 |
| 6.1 | Creating a Python application | 5 |
| 6.2 | Azure Event Hub Namespace | 6 |
| 6.3 | Azure Stream Analytics Job | 9 |
| 6.4 | Azure Data Lake Storage Gen 2 | 15 |
| 6.5 | Azure Data Factory Configuration | 18 |
| 6.6 | Azure Explorer & Databricks Configuration | 20 |
| 7 | Conclusion | 24 |

**Project 1 :- Streaming Data Processing**

1. **Project Statement**:

Set up an end-to-end pipeline with Azure Data Factory for ingesting streaming data and Azure Databricks for real-time processing and analysis of the streaming data.

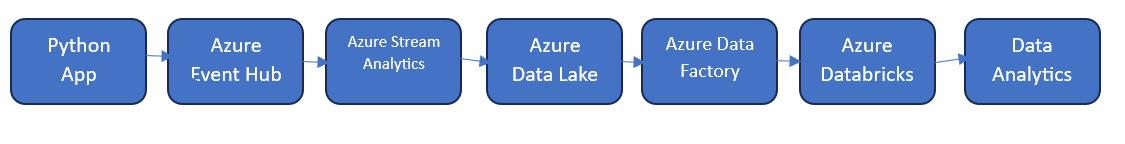
1. **Project Overview**:

The primary goal of this project is to design and implement a end-to-end data pipeline using Azure Data Factory (ADF) and Azure Databricks to ingest, process, and analyse streaming data in real-time. The pipeline will enable the extraction of valuable insights from the streaming data source, providing a foundation for timely decision-making.

1. **Project Requirements**:
   1. Azure Subscription

* You’re required to have an Azure Subscription to perform this project
  1. Data Source
* Created a python application, which fetches the weather data from the google website
  1. Azure Event Hub
* Azure Event Hubs is a cloud-based messaging service provided by Microsoft Azure which enables large-scale, real-time event streaming for ingesting and processing data from various sources.
  1. Azure Stream Analytics
* Azure Stream Analytics is a cloud-based service provided by Microsoft Azure that helps you analyse and gain insights from real-time data streams.
  1. Azure Data Lake
* Azure Data Lake is like a massive storage space in the cloud where you can store and manage enormous amounts of data. It's designed to handle both large files and lots of small files
  1. Azure Data Factory
* Azure Data Factory is like a digital factory that helps you collect, transform, and move your data from one place to another in a smooth and organized way.
  1. Azure Databricks
* Azure Databricks is like a super-smart workspace in the cloud where you can easily analyse and process large amounts of data using Apache Spark.

1. **Architecture Diagram**:

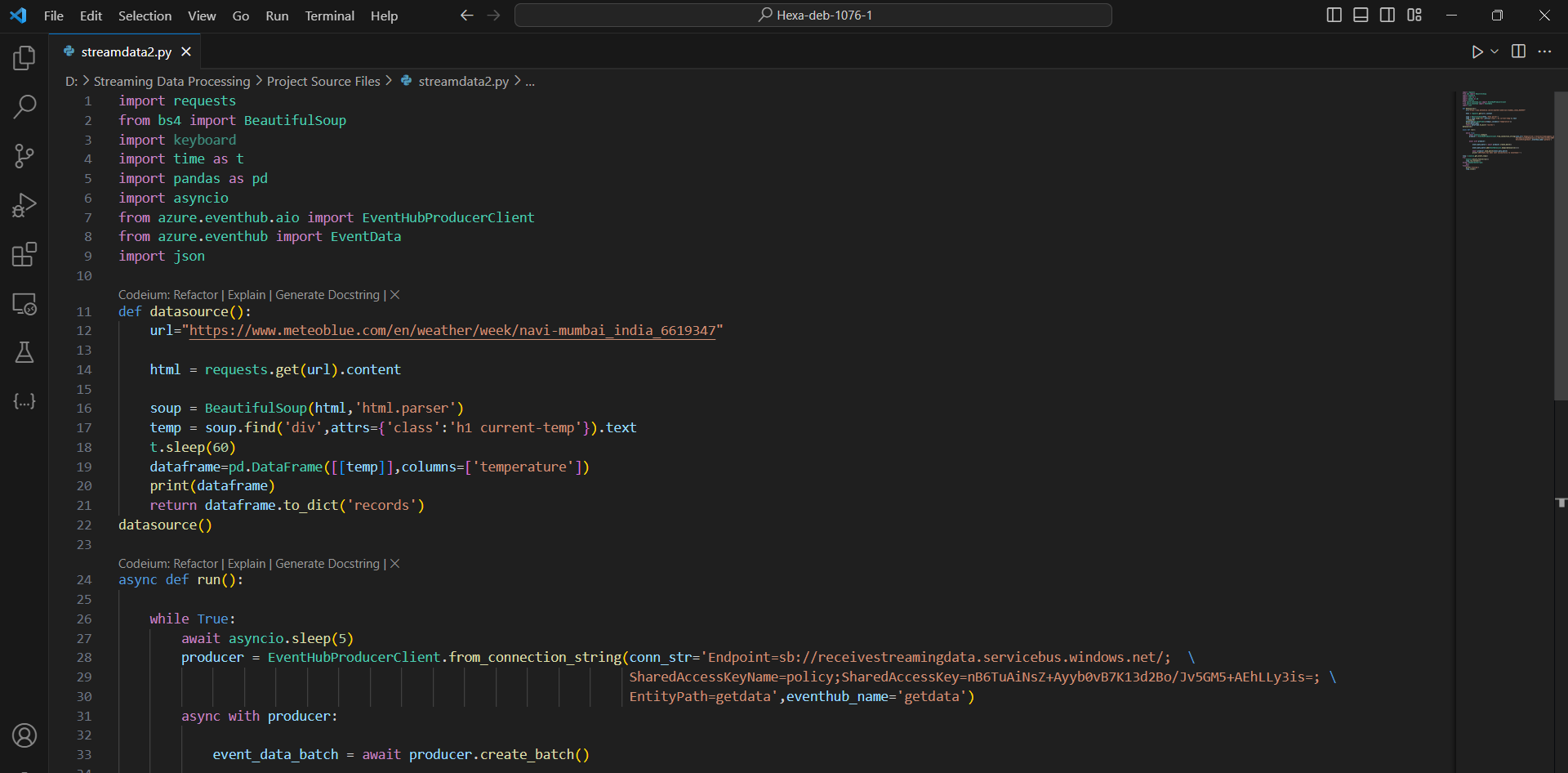


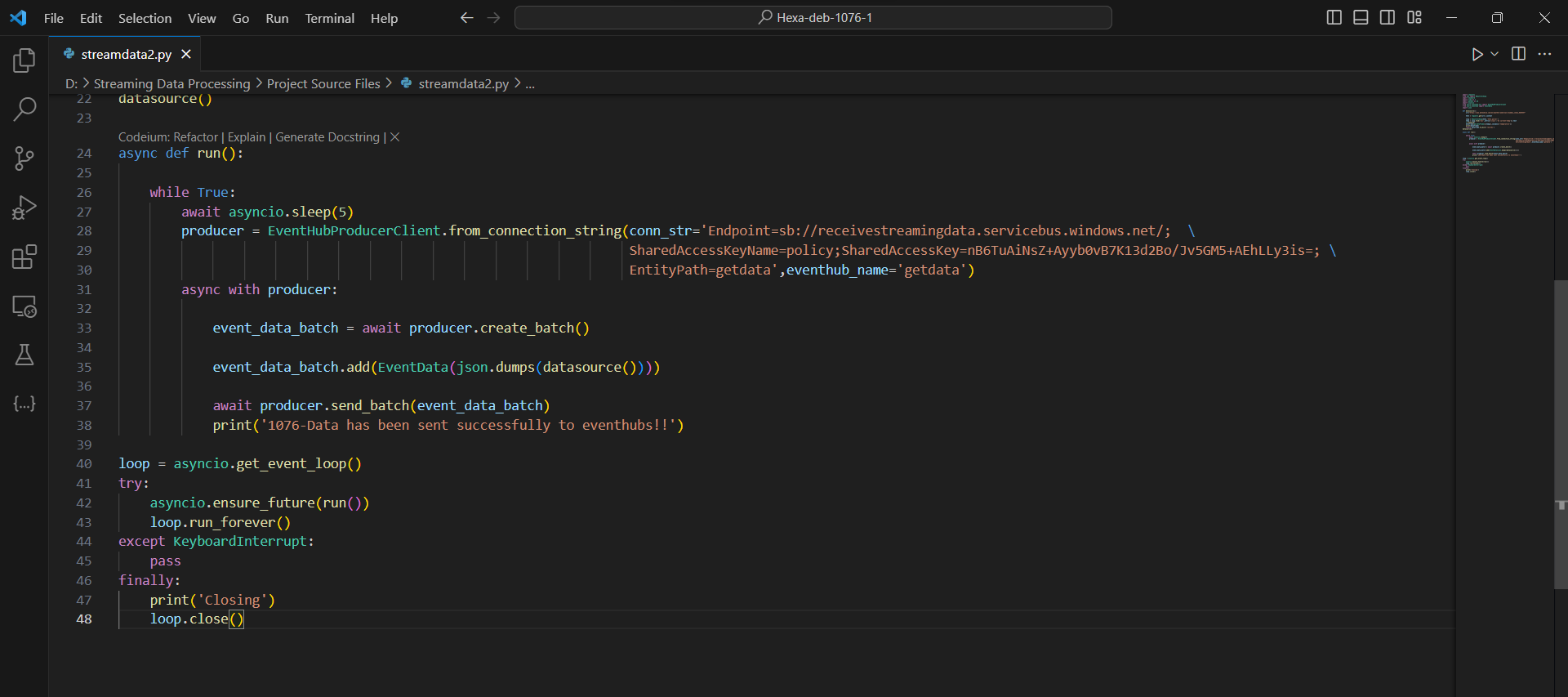
1. **Execution Overview**:

* Creating a python application which fetch a stream data from the webpages
* Connecting the Python application with the Azure Event Hub and receive the streaming data
* Use of Azure Stream Analytics to process the streaming data for storage
* Creating an Azure Data Lake Storage Gen2 to store the received date which is in **JSON** format
* Creating Data Factory for Sink Data storage
* Download and upload the **JSON** file into Azure Databricks
* Use Spark commands to perform data analysis on the **CSV** file

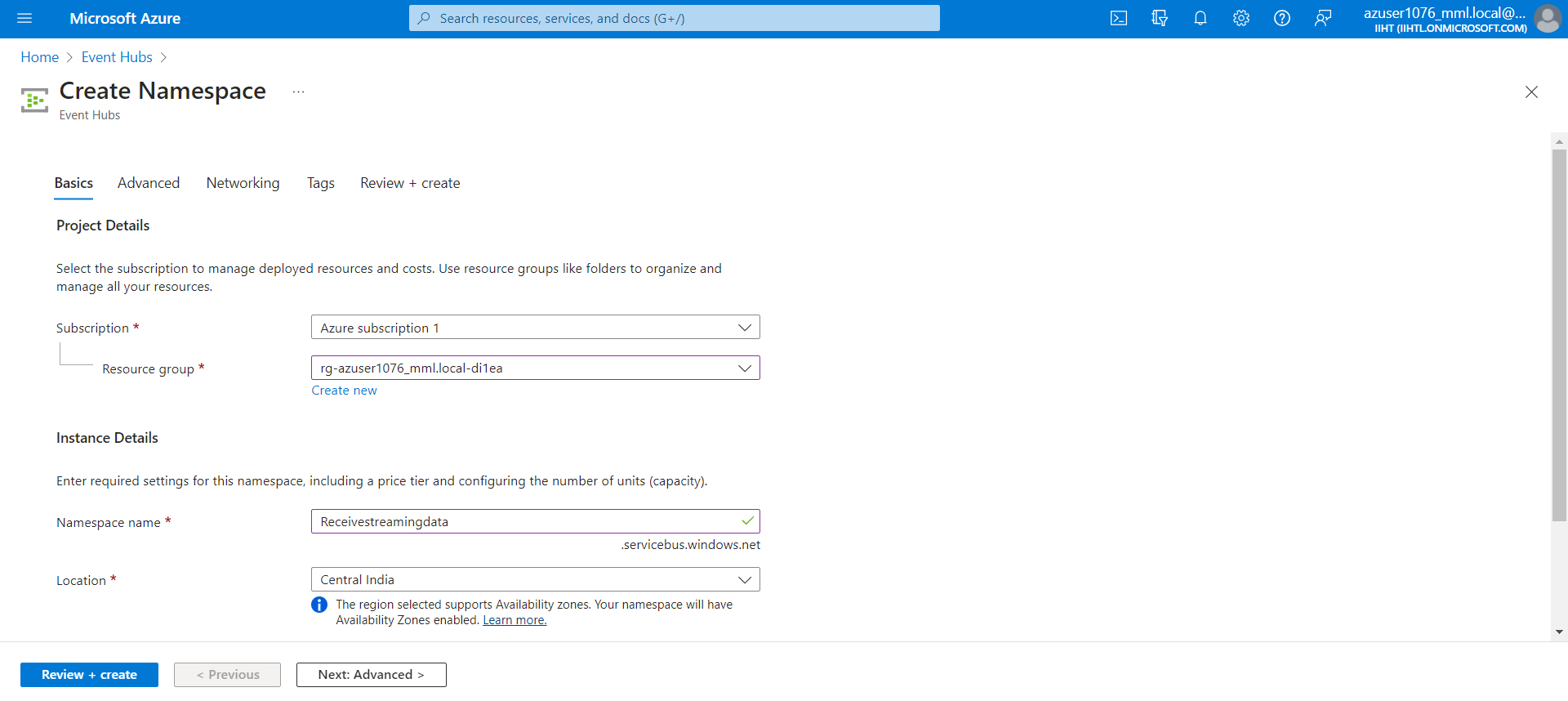
1. **Project Implementation**

6.1) Creating a Python application

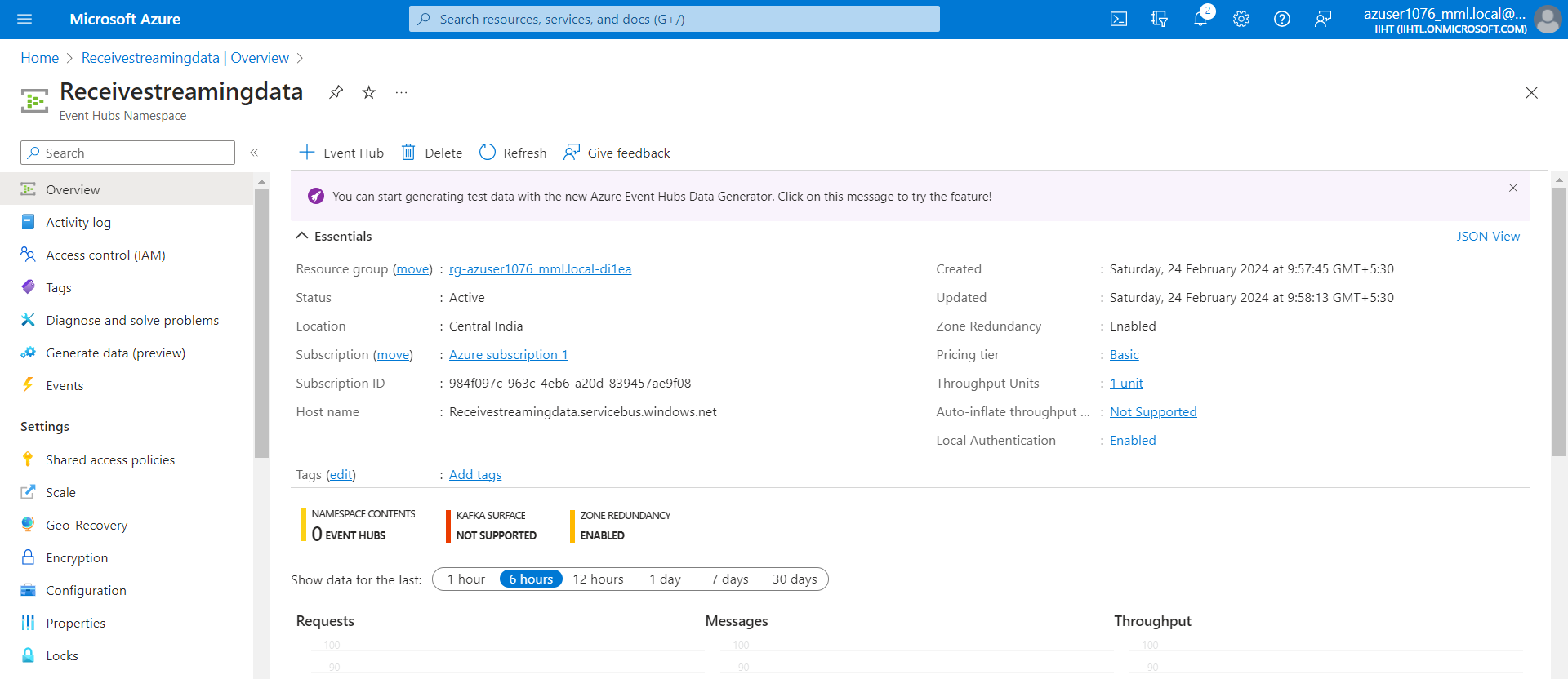




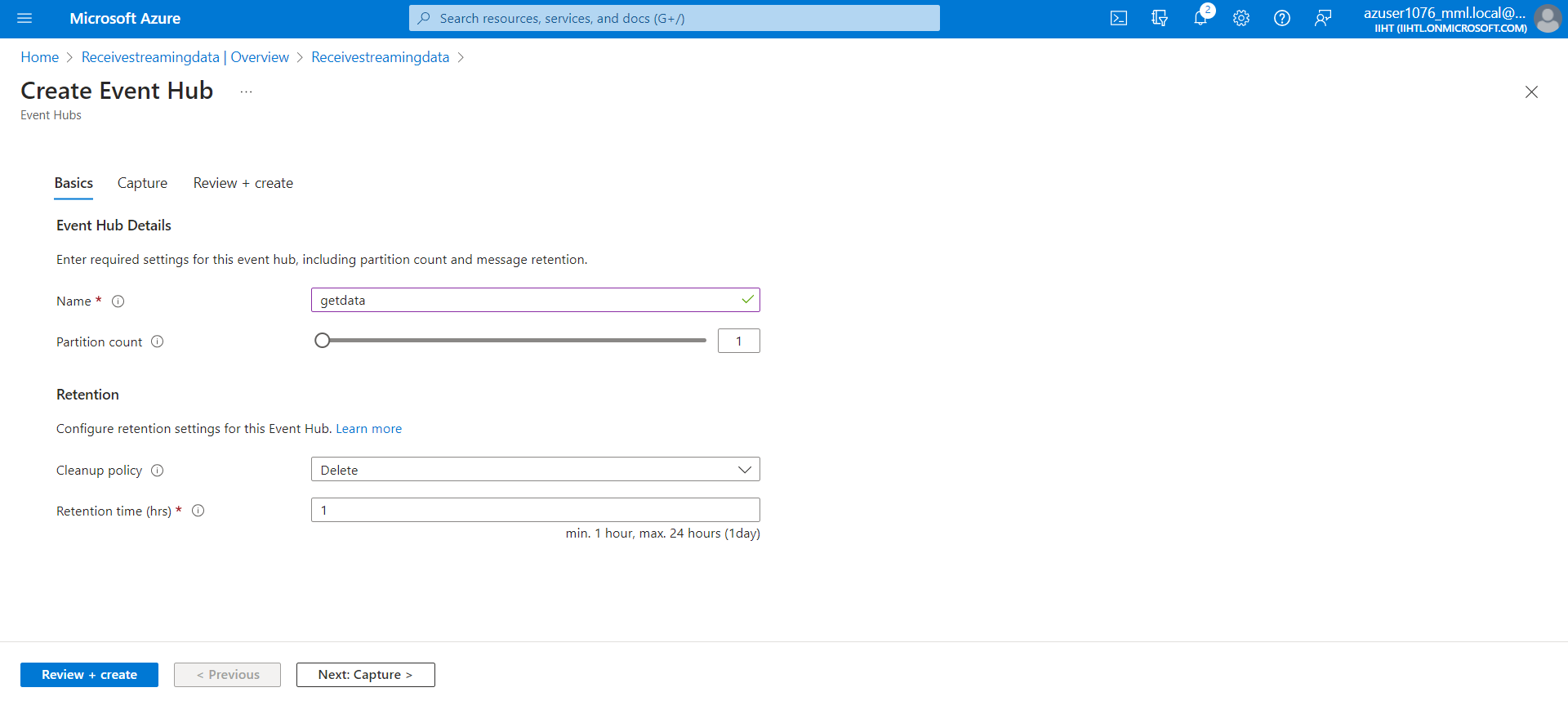
6.2) Creating an Event Hub Namespace



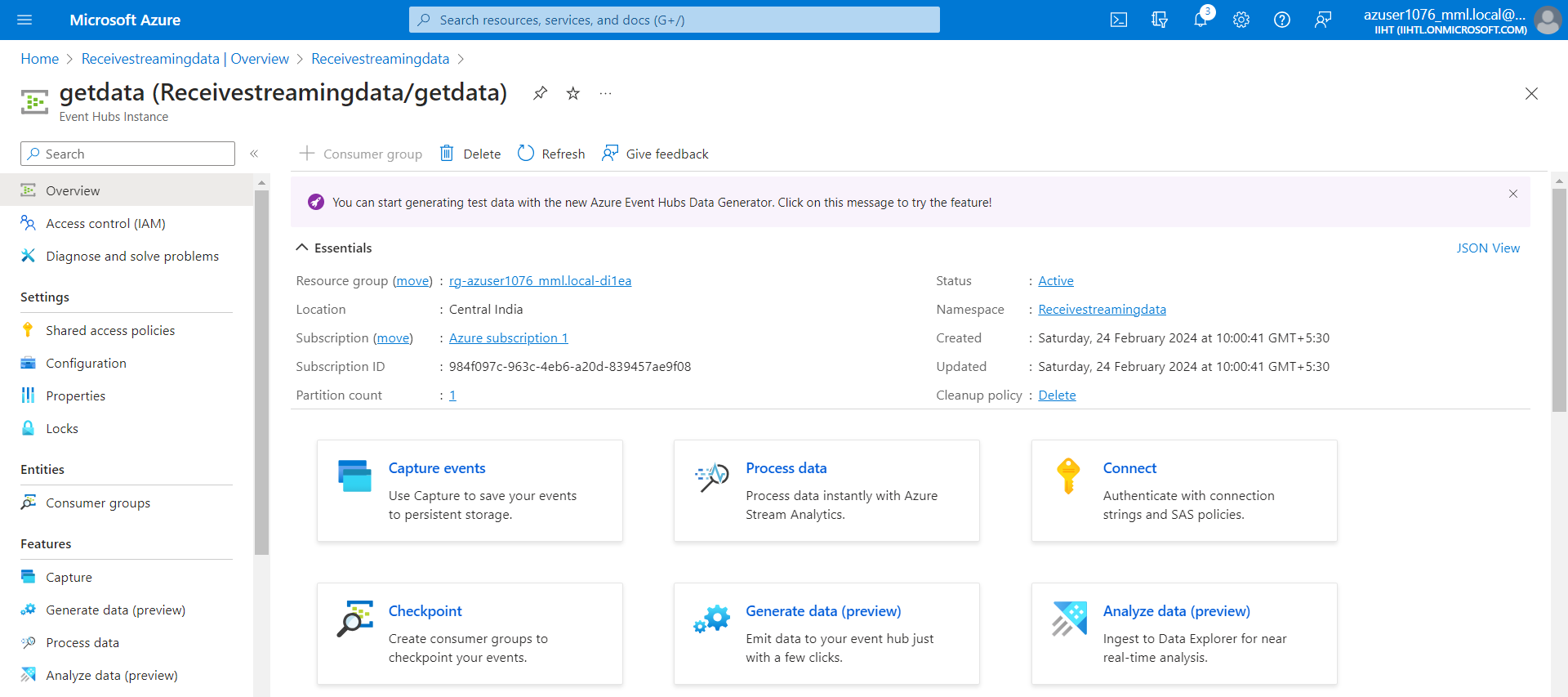
Output :



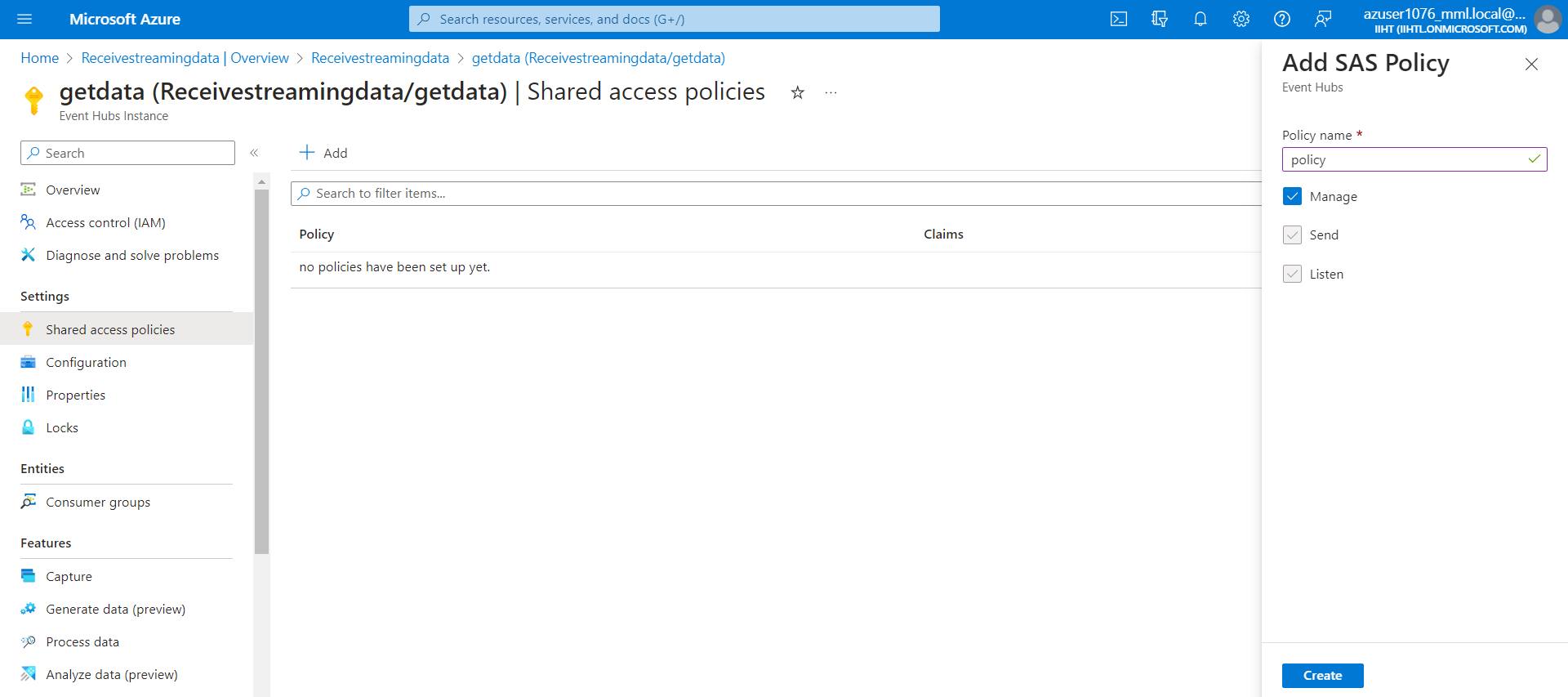
Creating an Event Hub to receive the streaming data



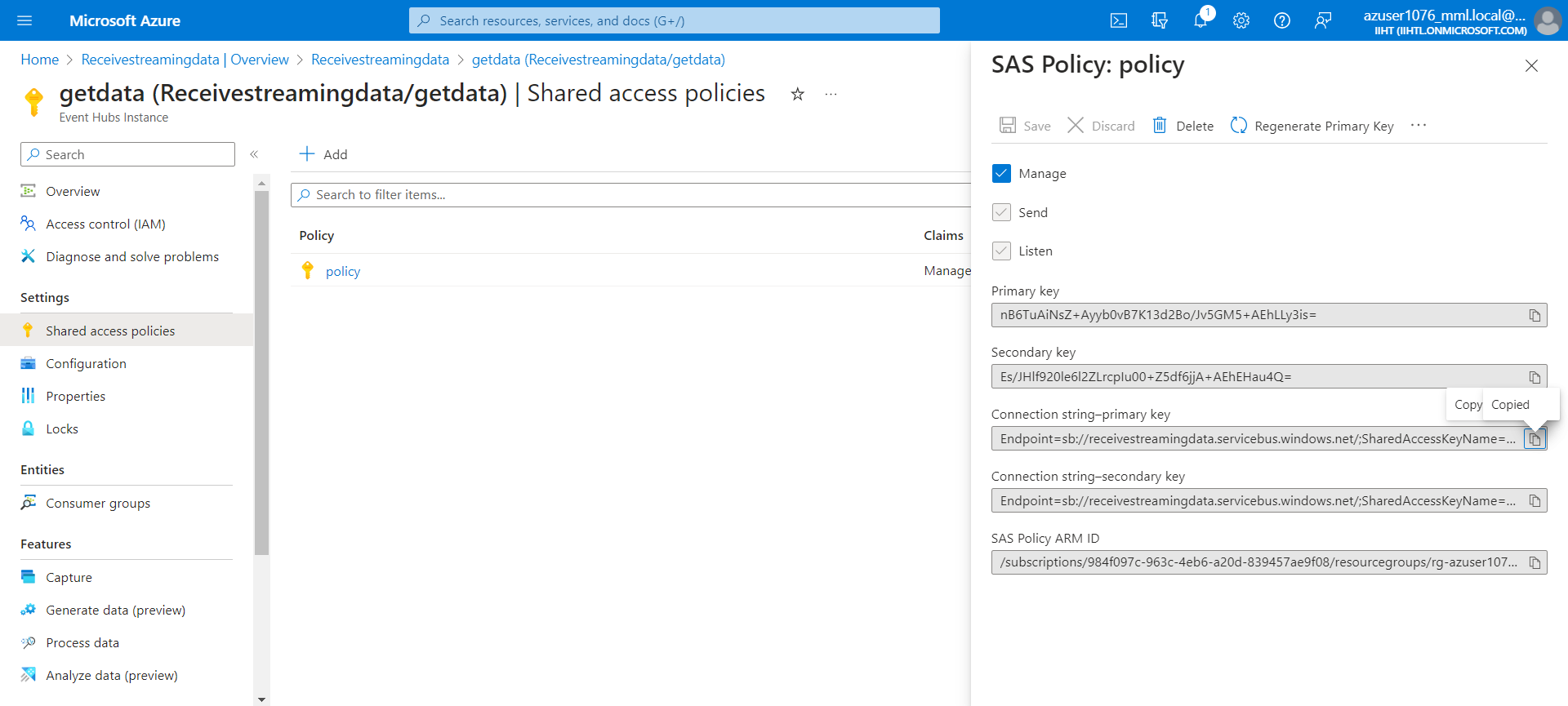
Output:



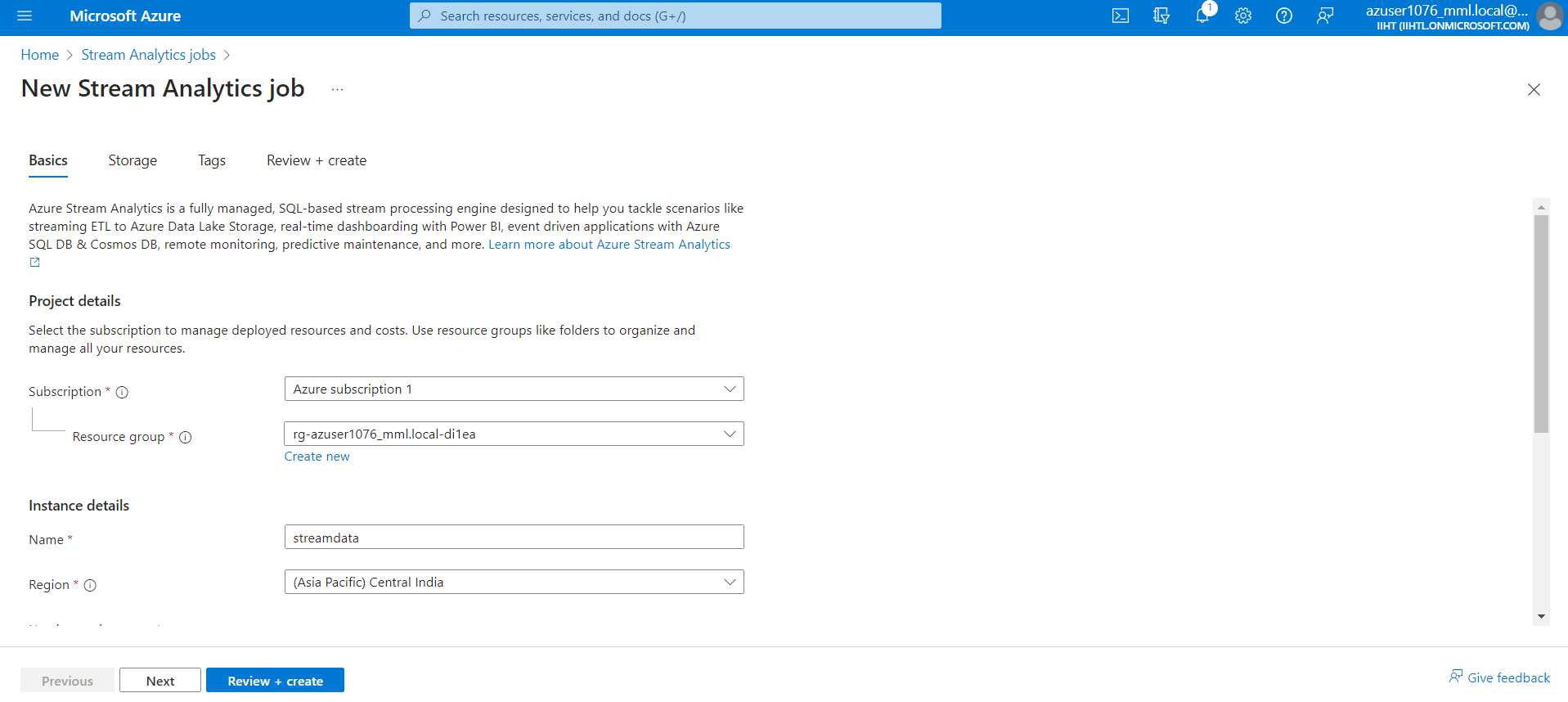
Creating a policy in the created Event Hub



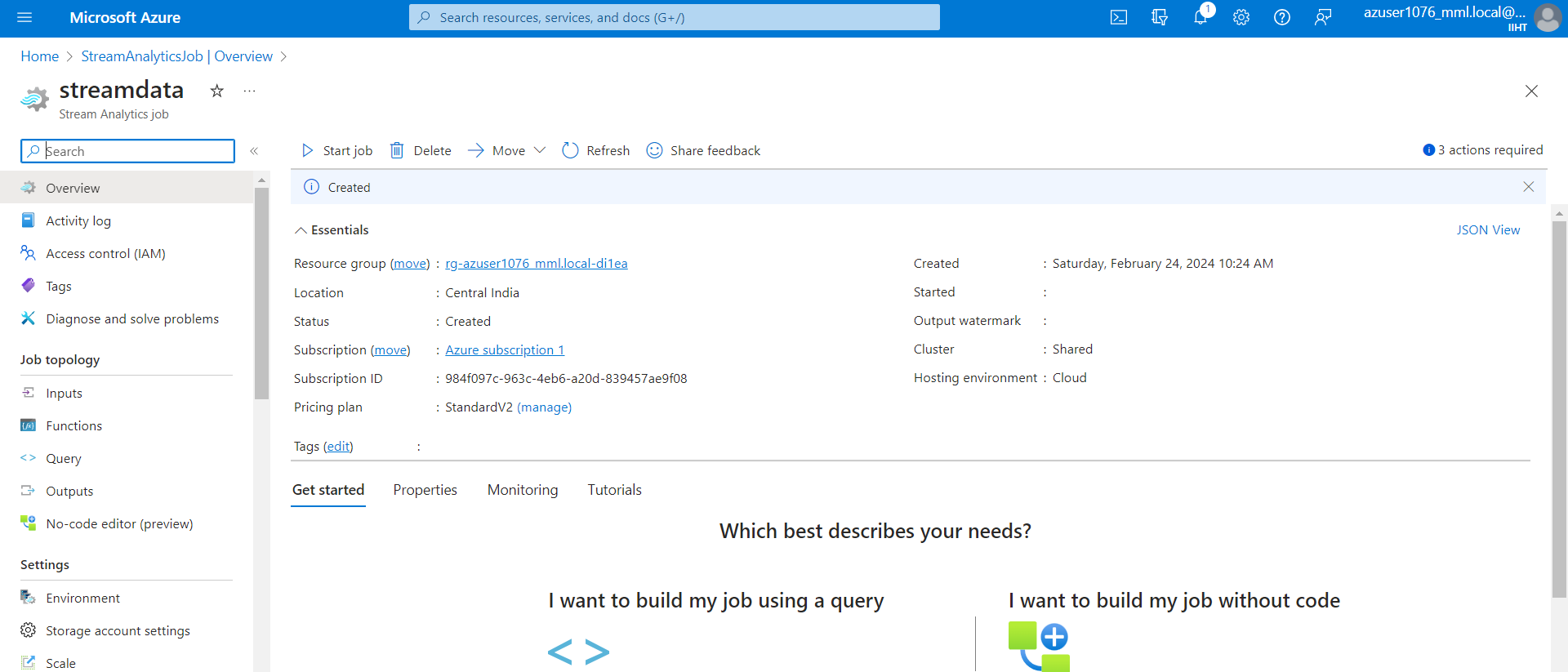
After creating the policy, copy the connection string primary-key and paste that in the python application to establish the connection



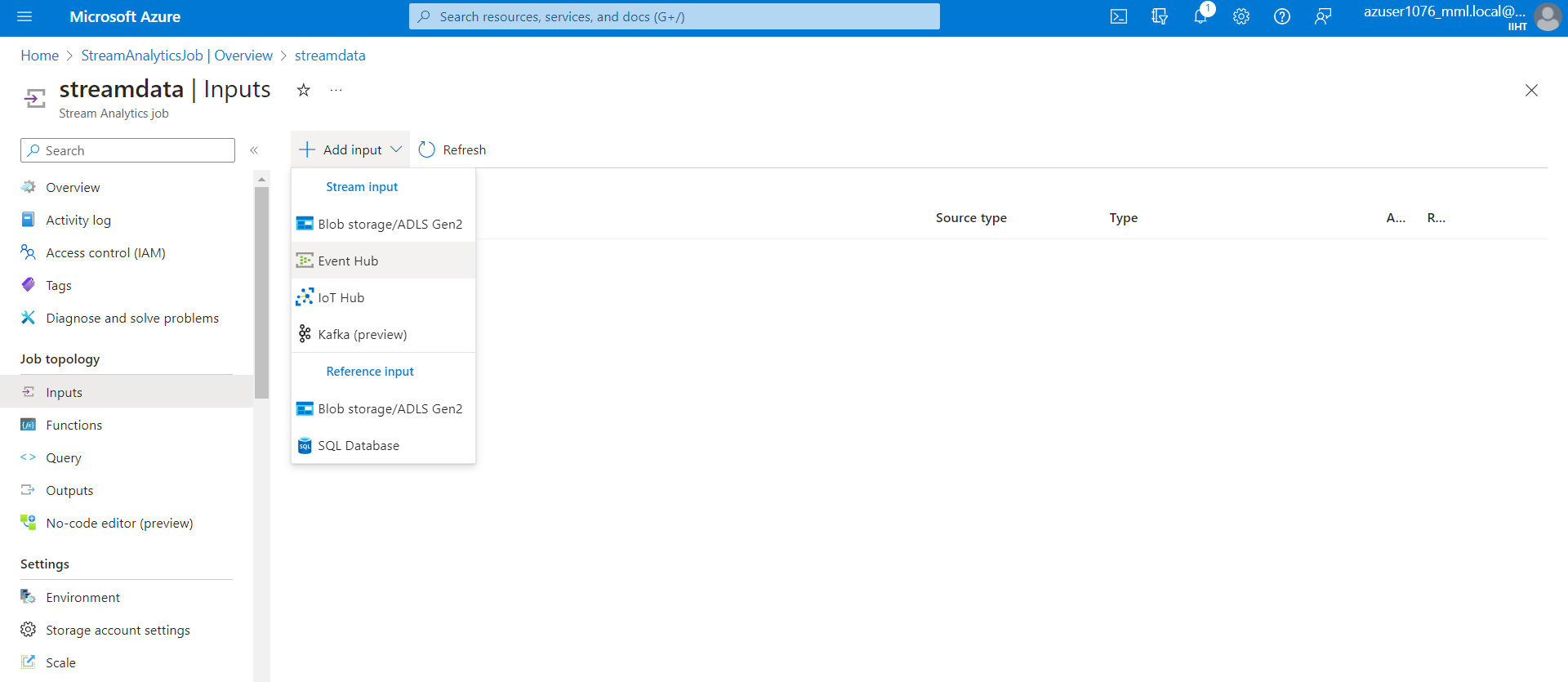
6.3) Creating an Azure Stream Analytics Job



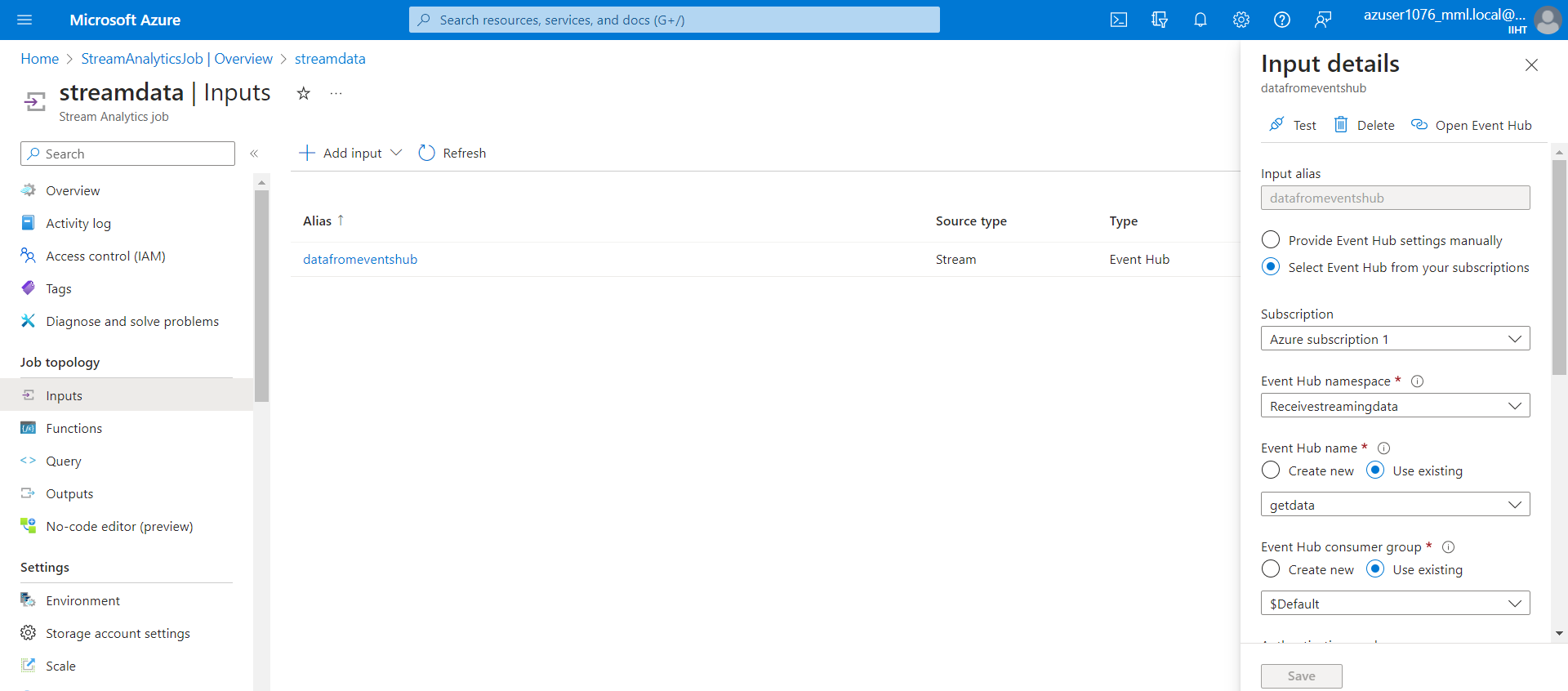
Output:



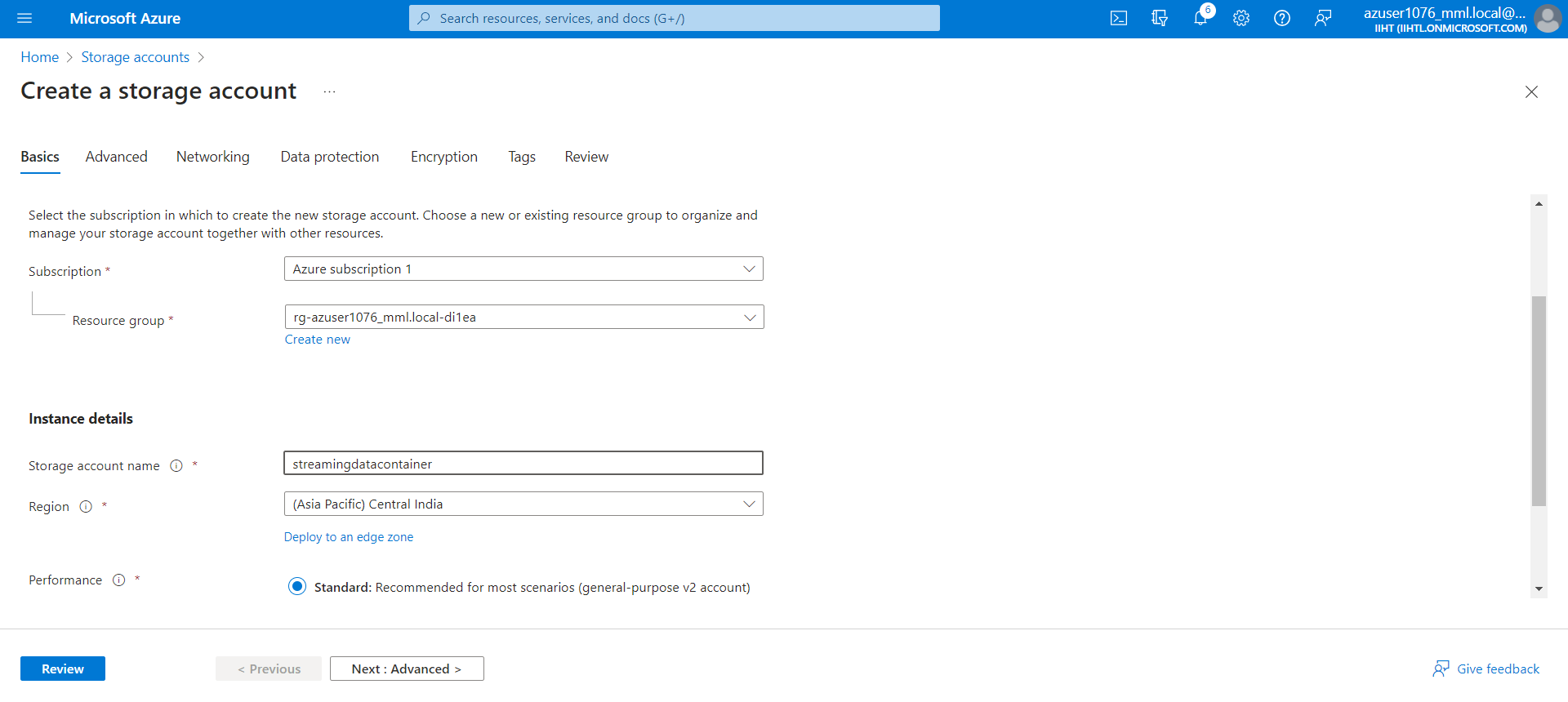
Adding input to the Stream Analytics Job, As the data are receiving through the Event Hub so Choose select the input type as Event Hub



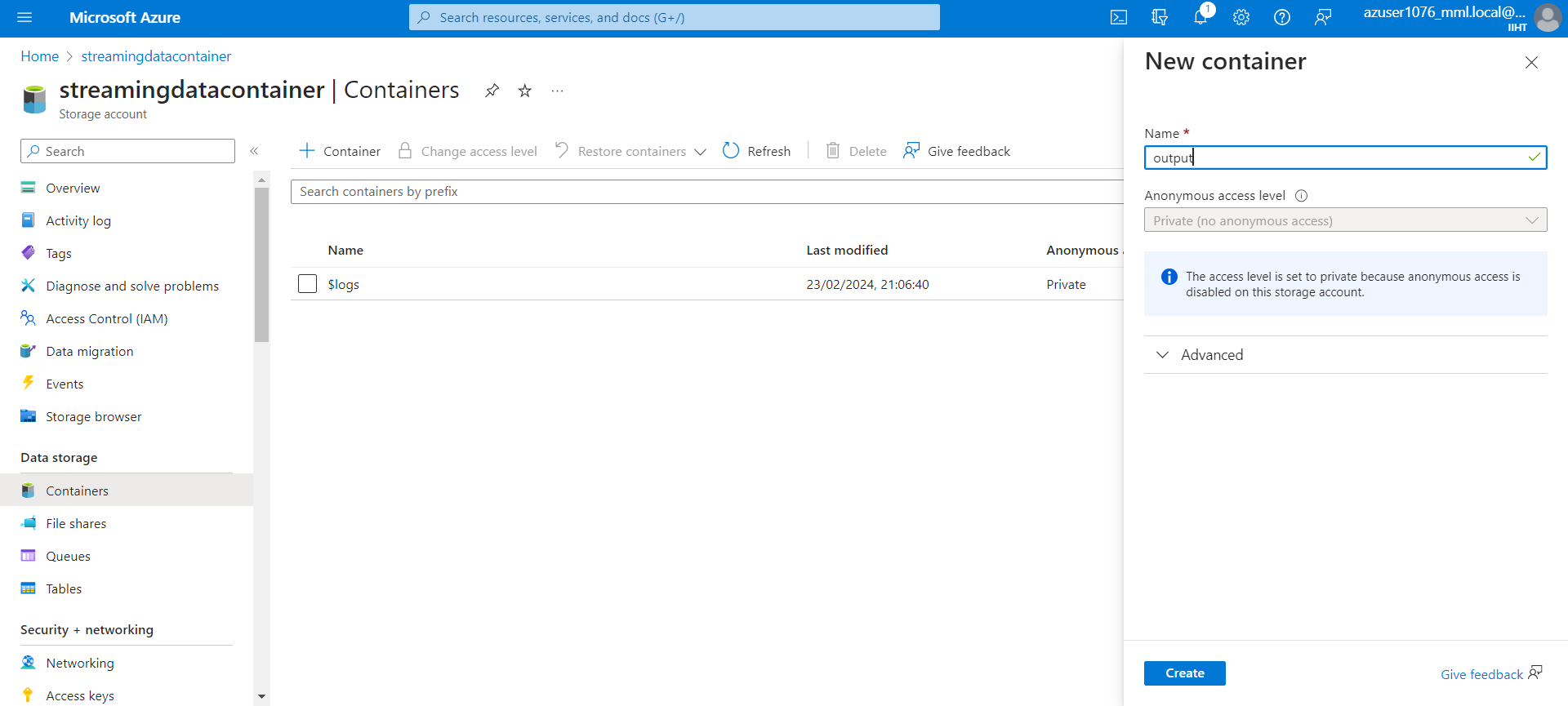
Enter the Event Hub Namespace name and other details in the input field



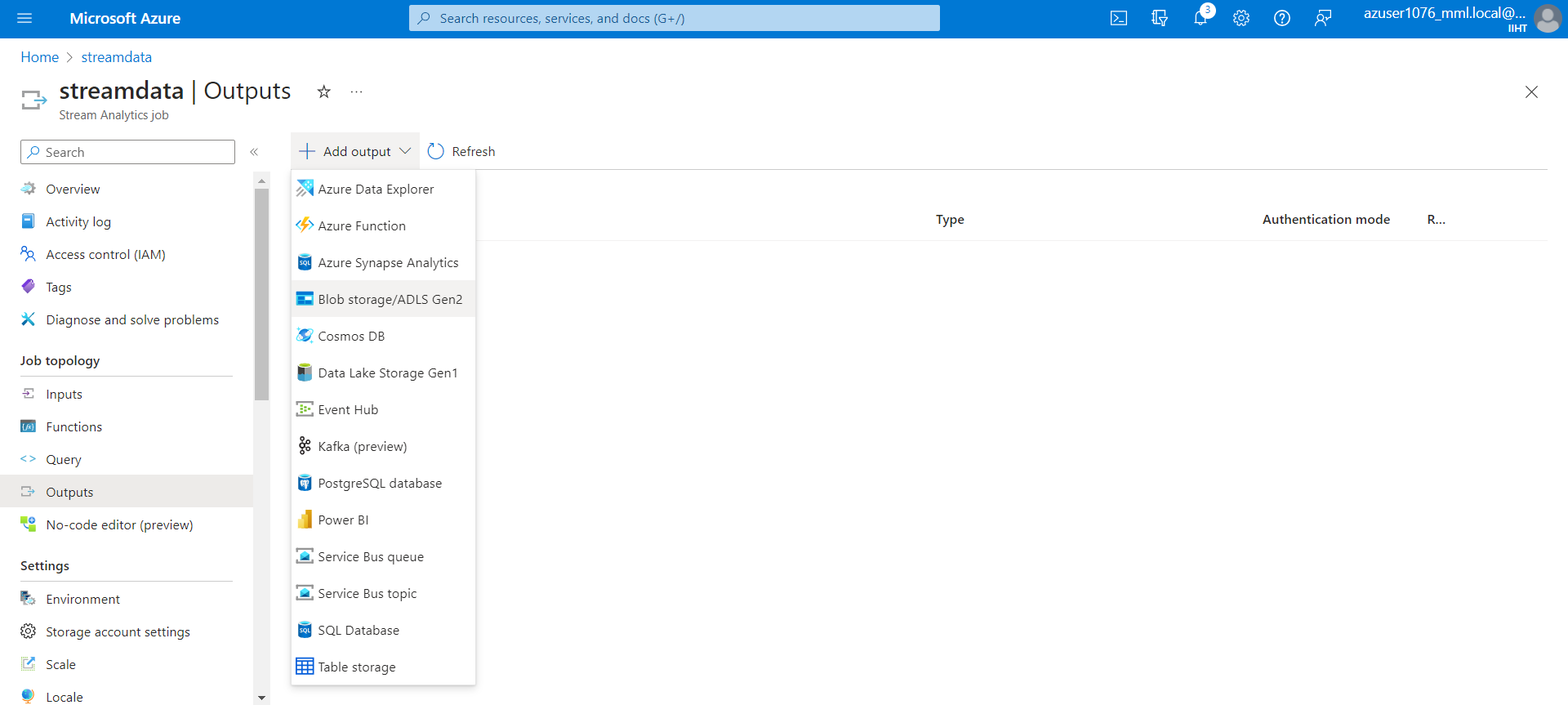
Creating an Azure Data Lake Storage



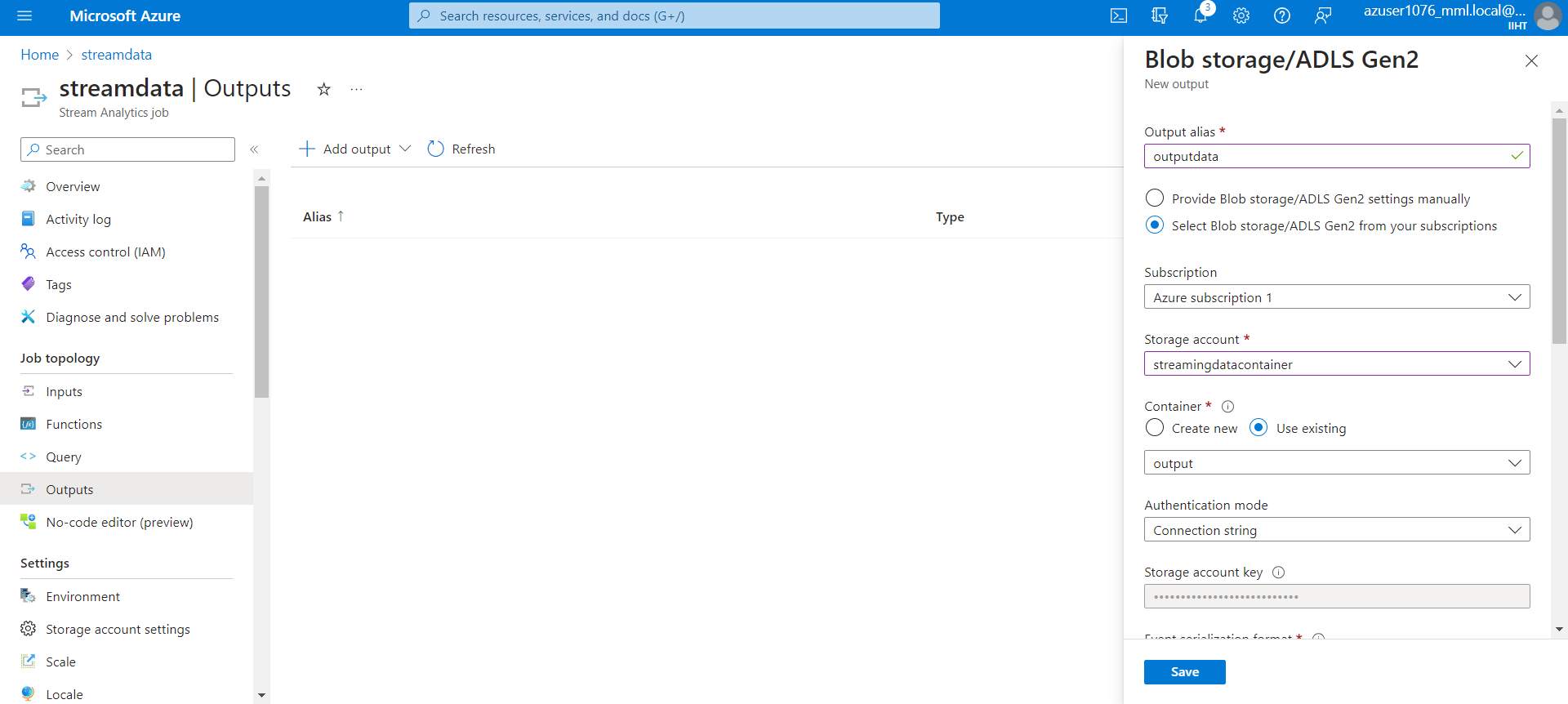
Creating an empty container to ingest the received streaming data



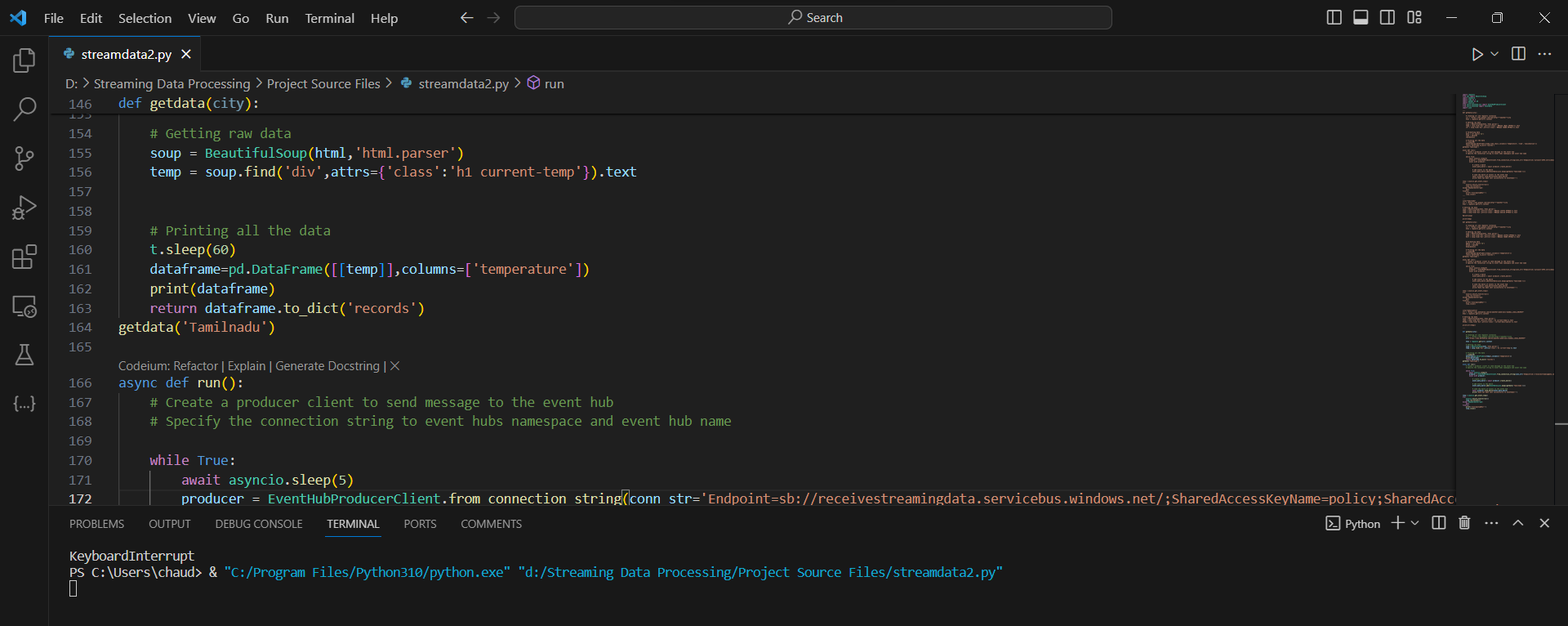
Now go to the output service in the Azure Stream Analytics Job and selecting output option as Blob Storage / ADLS Gen2



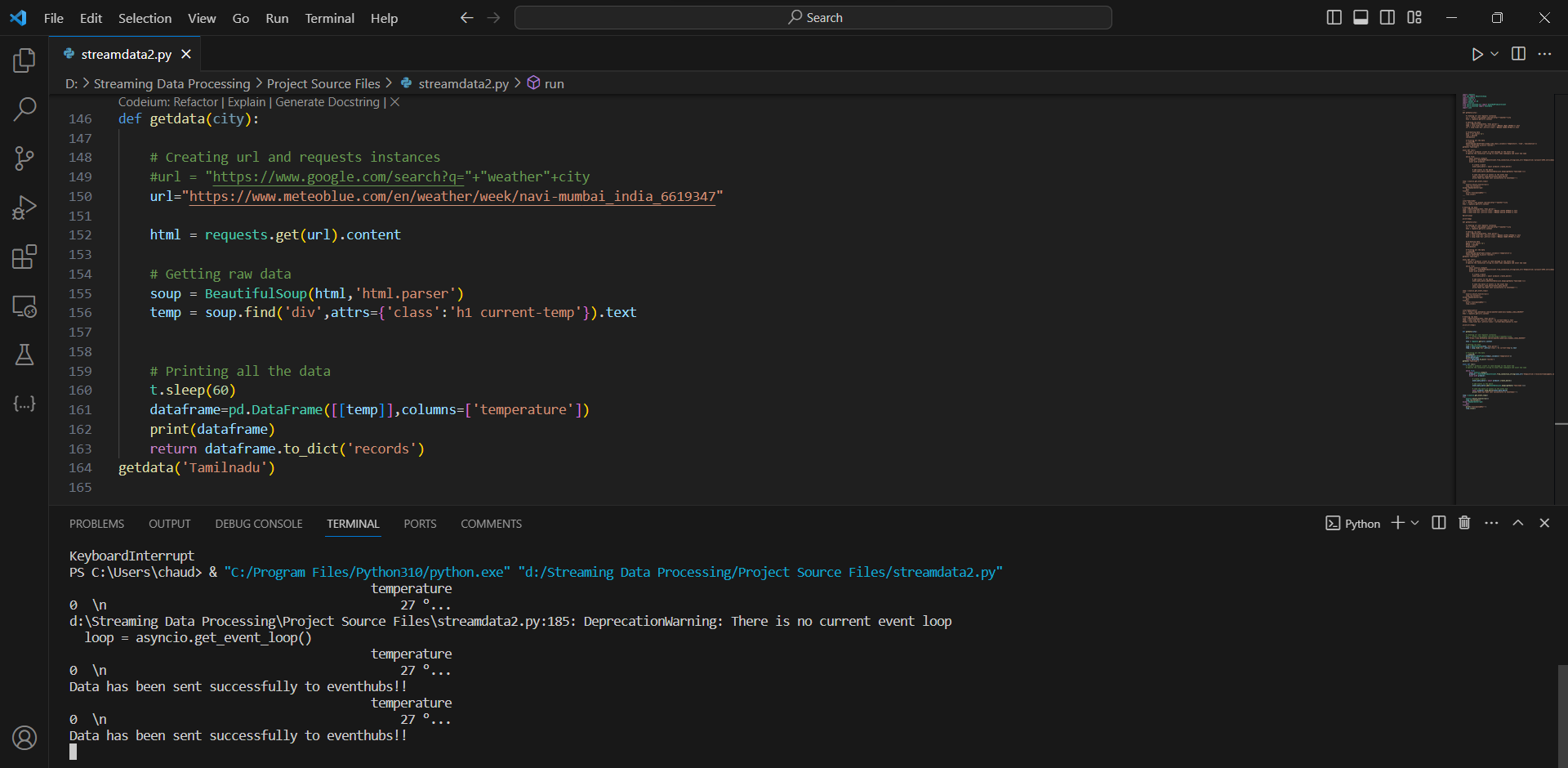
Entering the Azure Data Lake Storage name with container name and other details in the output field



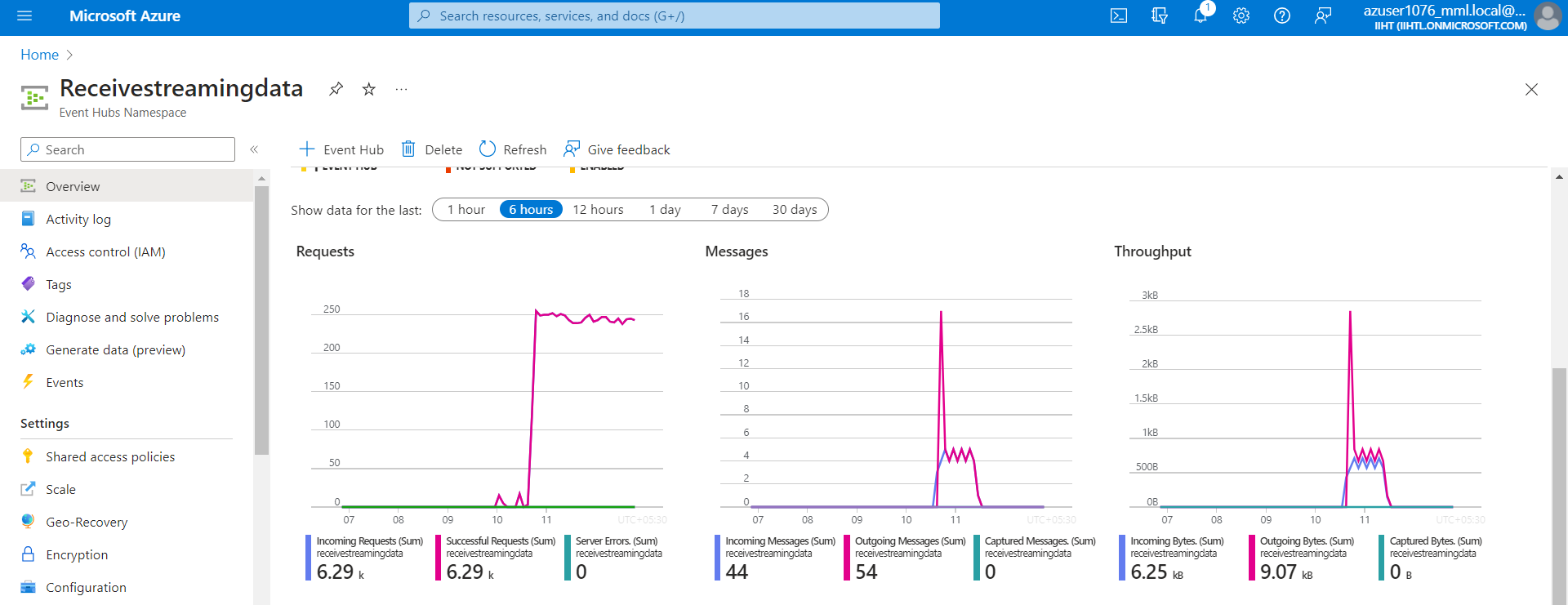
Now, running the python application start the process



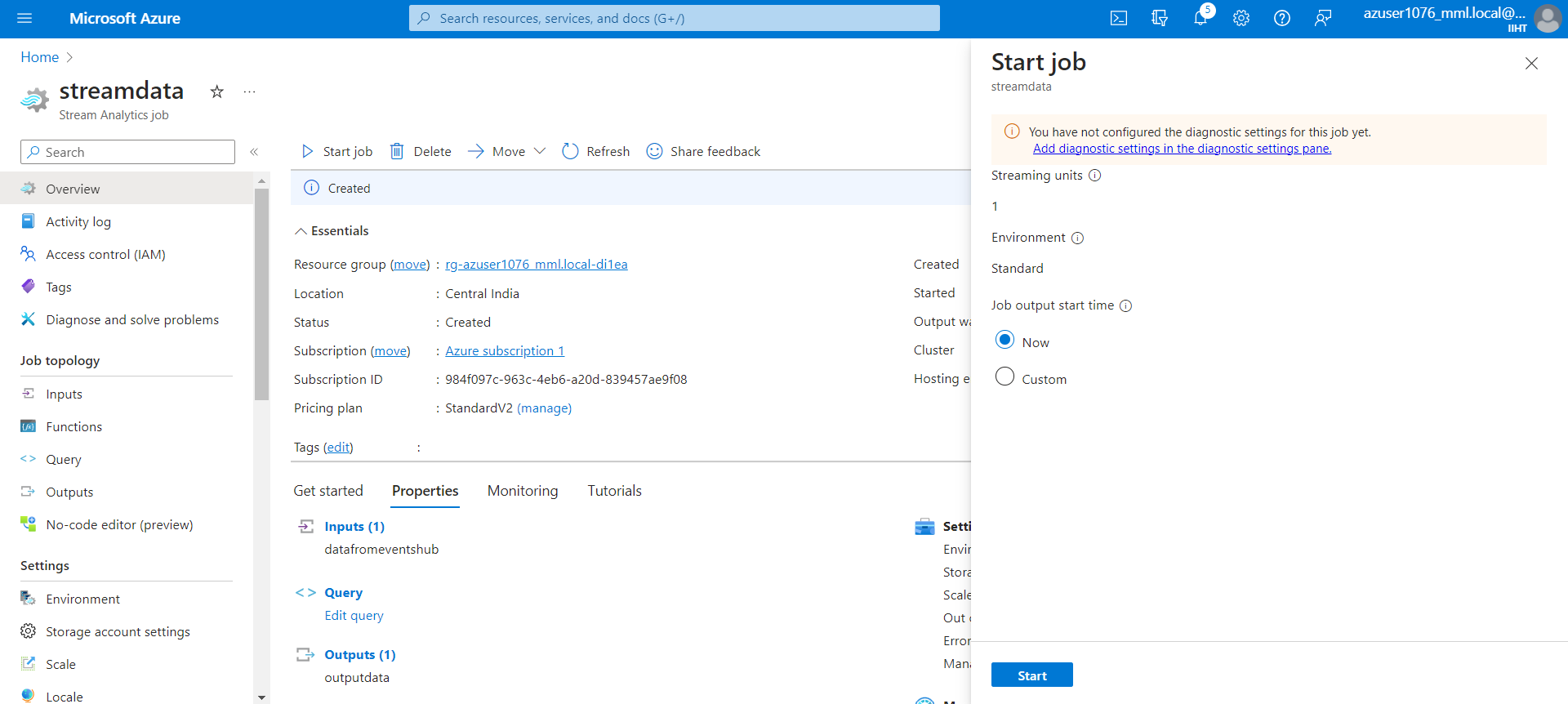
Once the application started running successfully, we can able to see the output data which is been sent and also **“Data has been sent successfully to eventhubs ”**



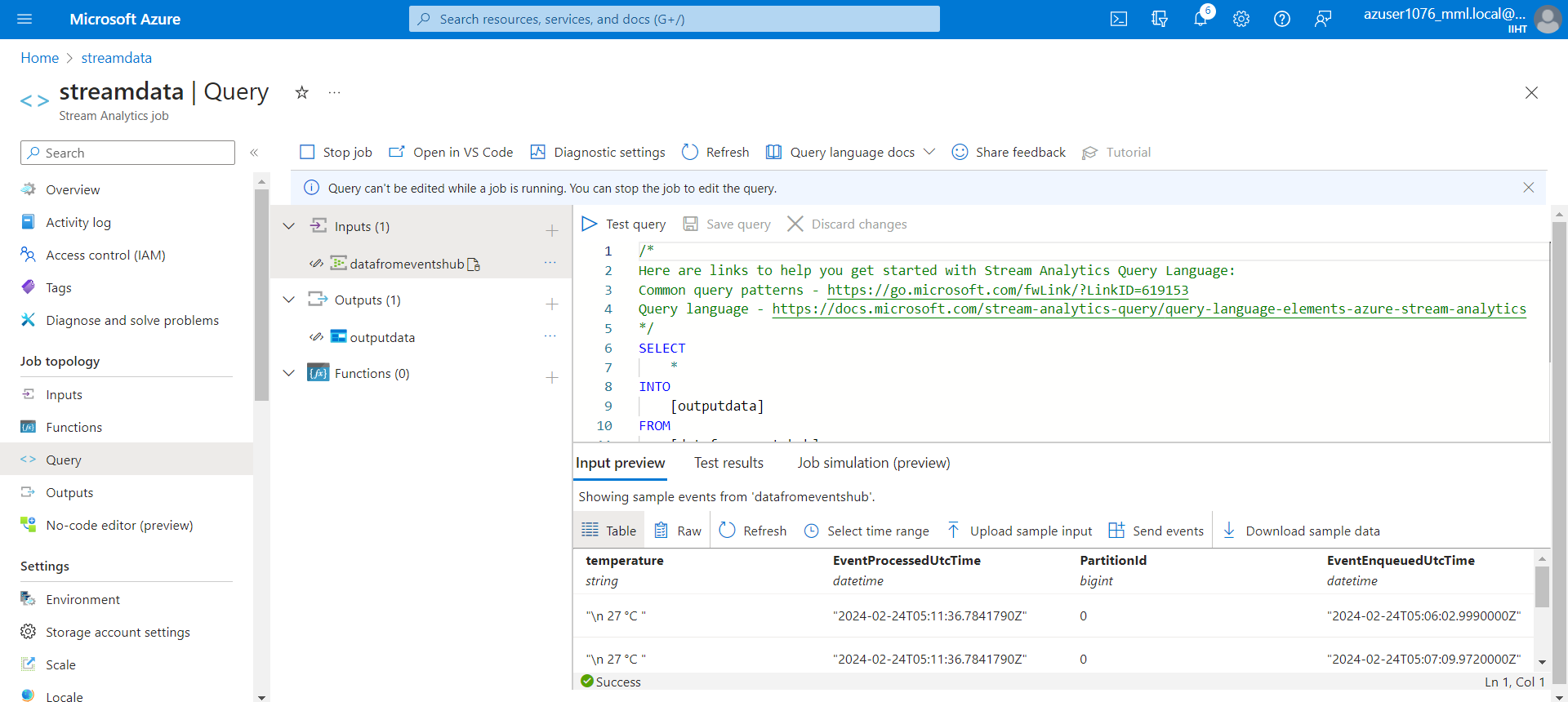
This is the Event hub Metric where data is getting received:



Now, the Azure Stream Analytics Job and start the job

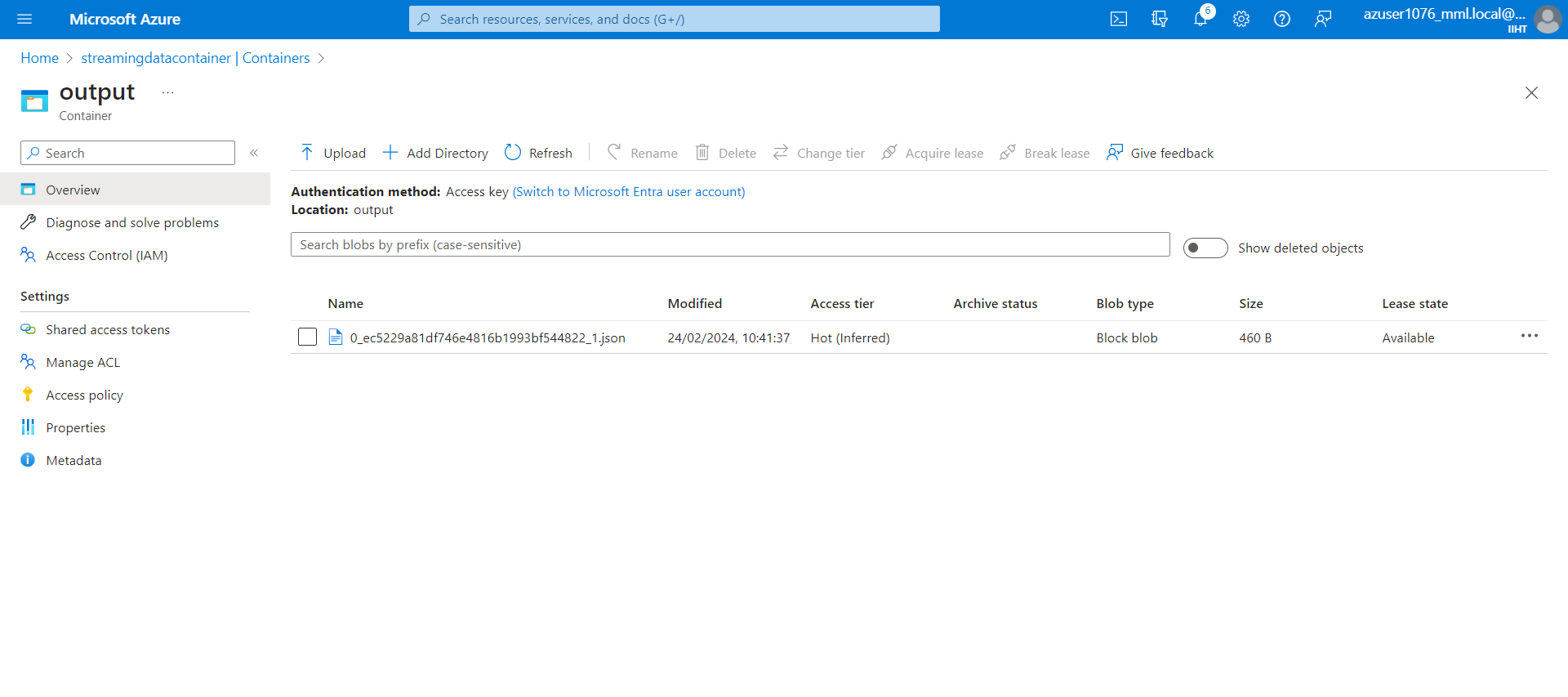


After starting the job, refreshing the query service, now we can able to see the streaming data coming from the google webpage

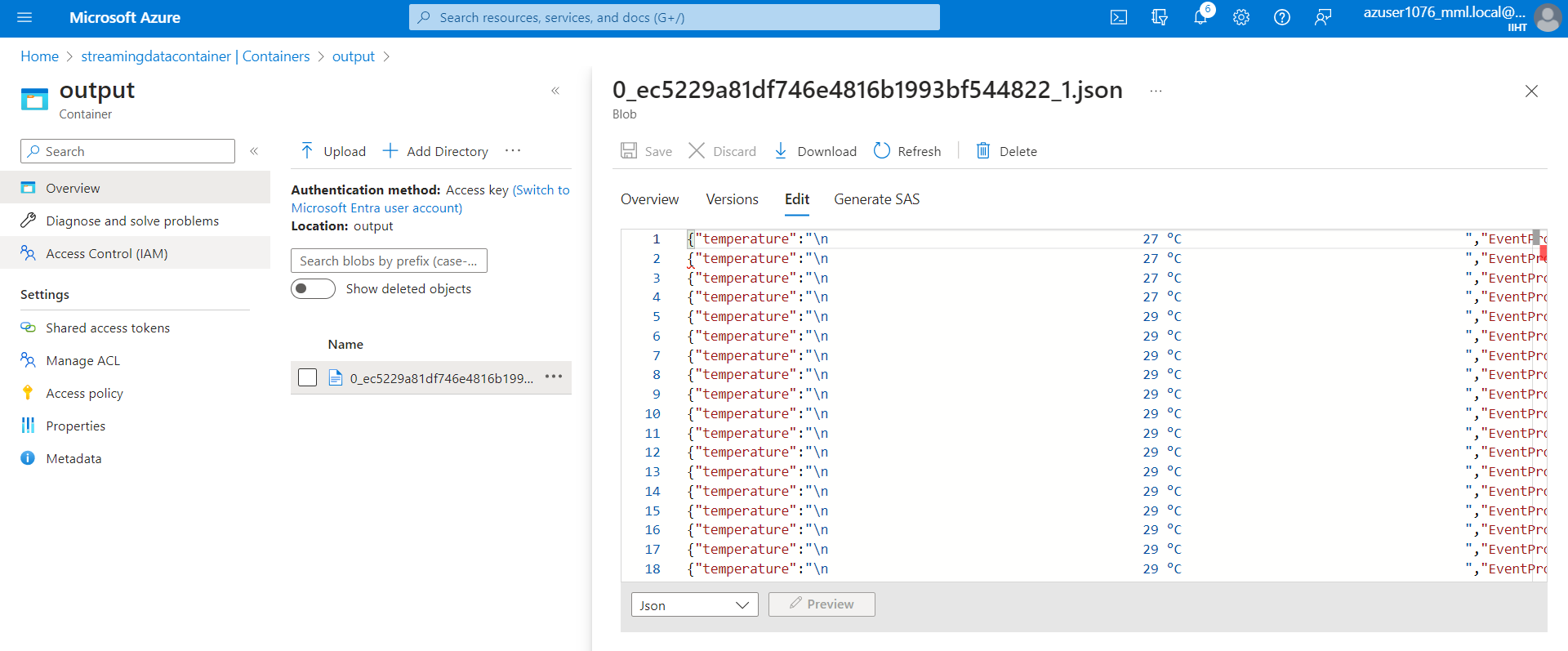


6.4) Azure Data Lake Storage

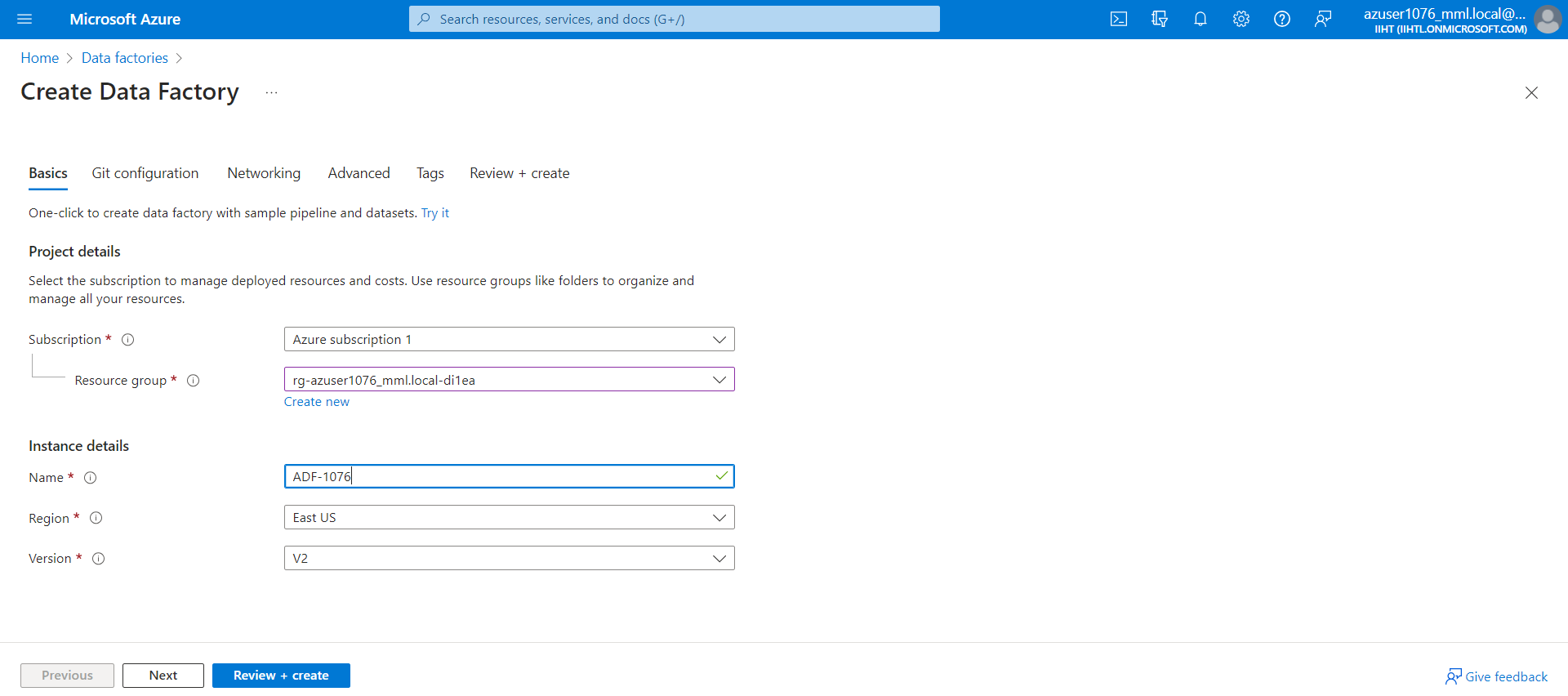
Go to the output container where we will be able to see the **JSON** formatted files present in the container



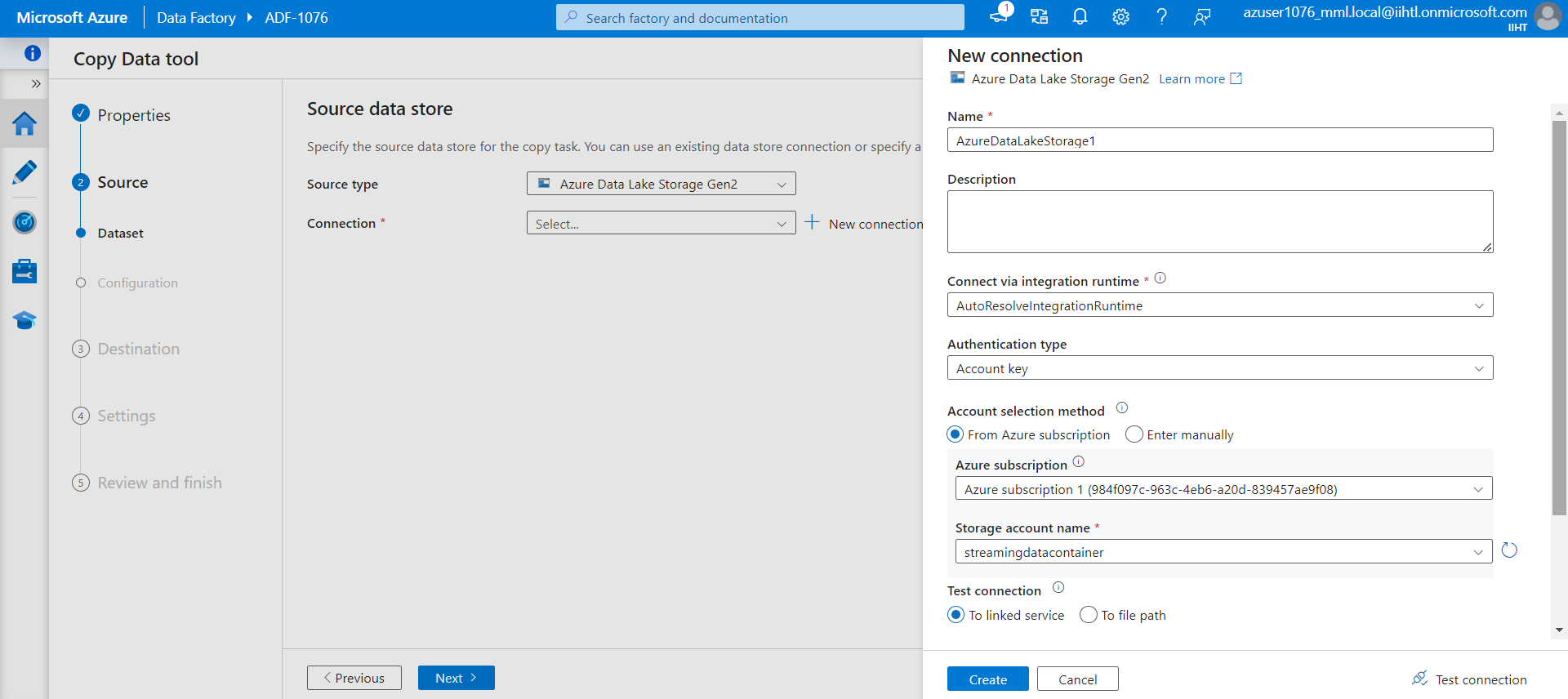
Data is present in the JSON file



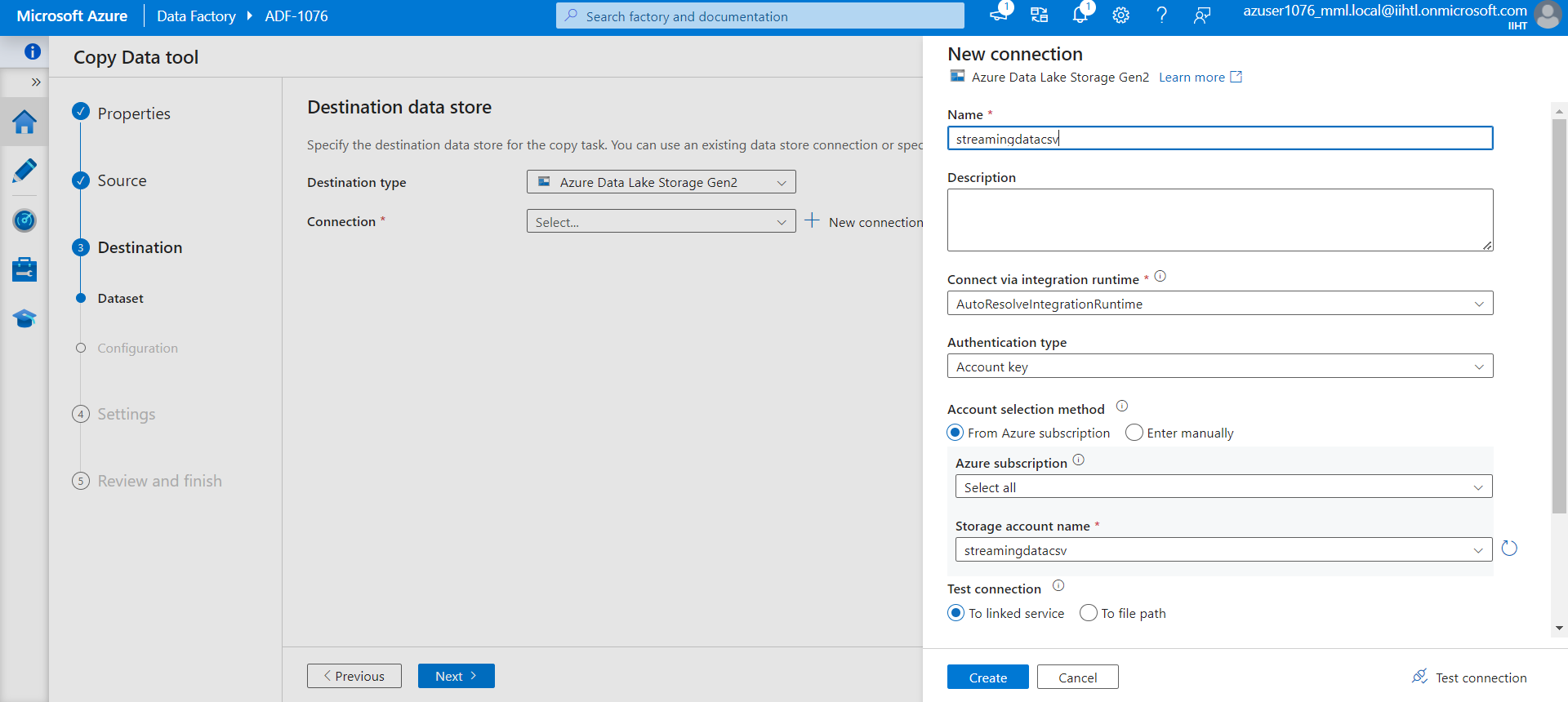
Creating Data Factory:



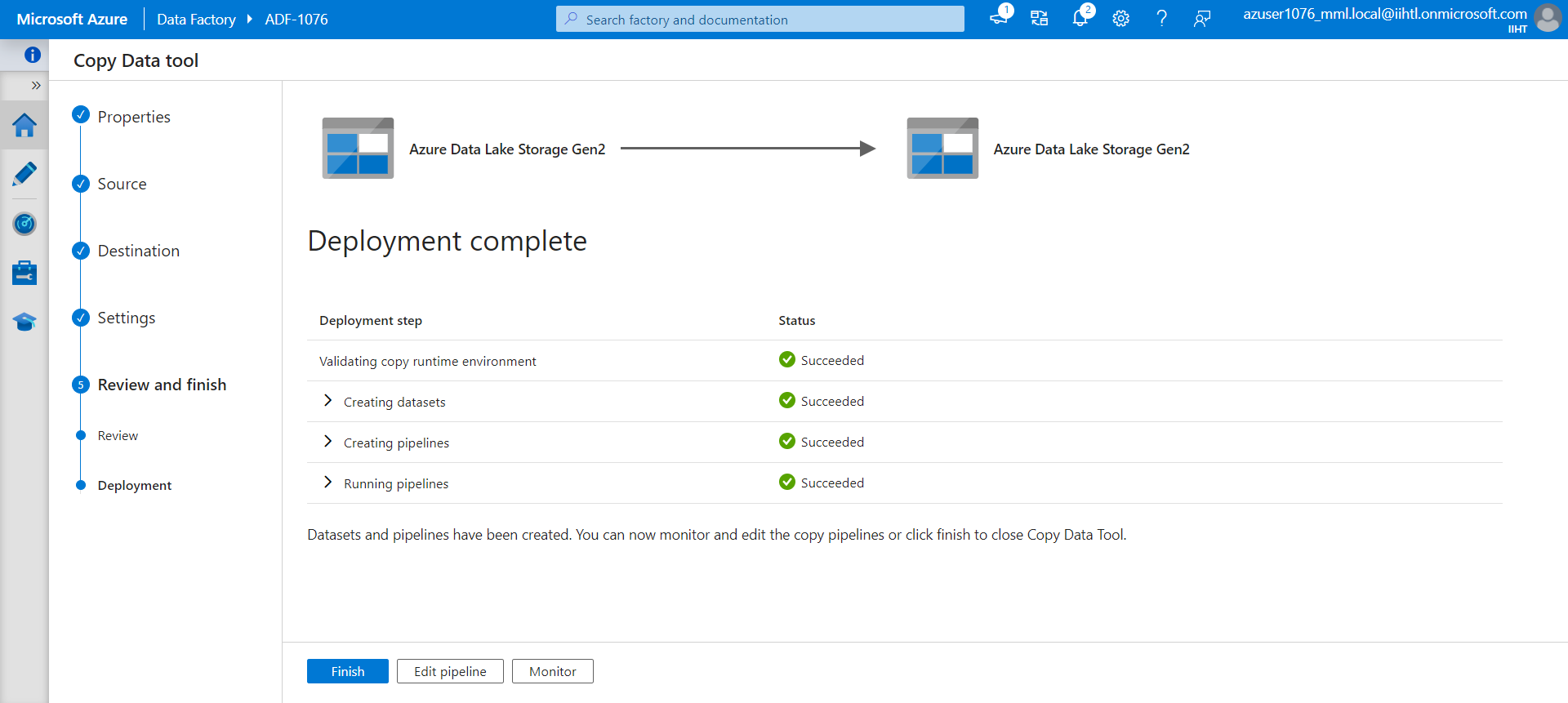
Selecting Source Data Storage:



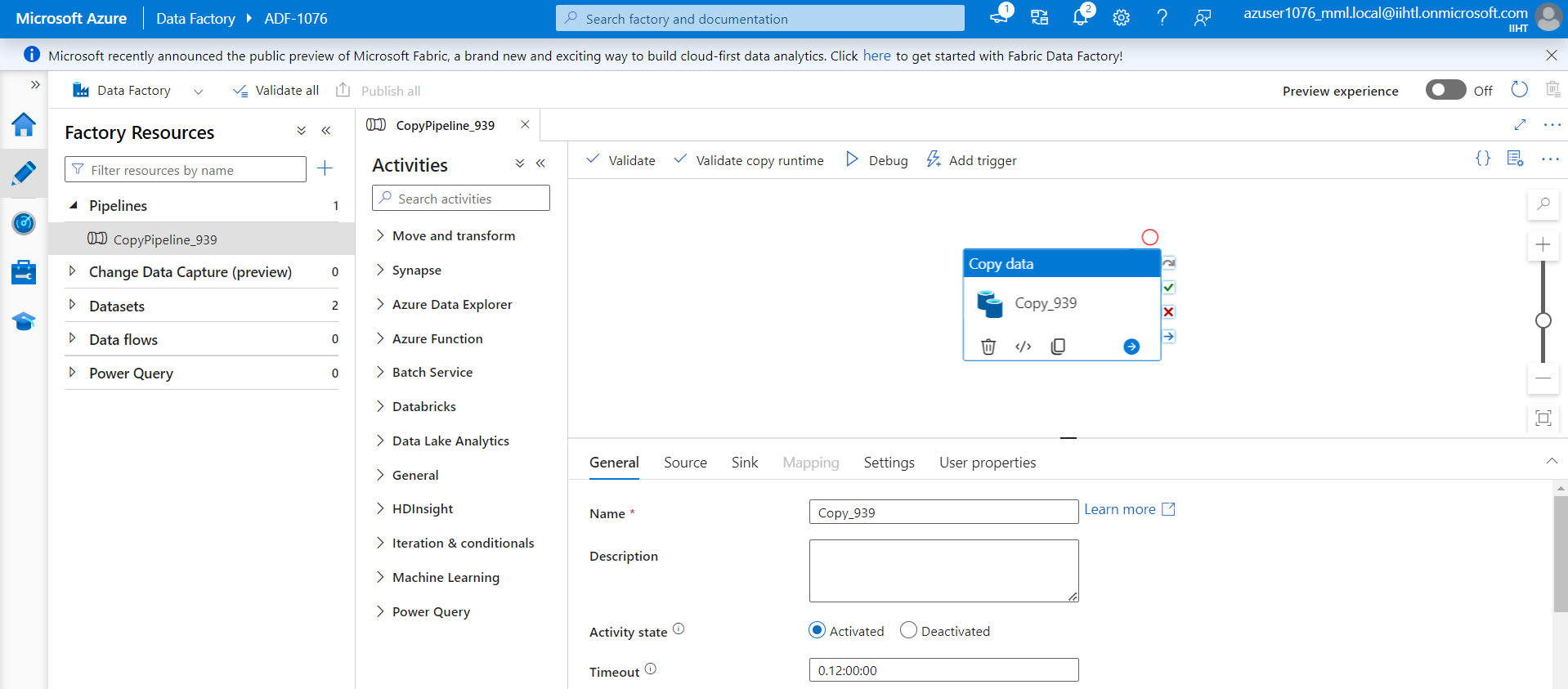
Selecting Destination Data Storage:



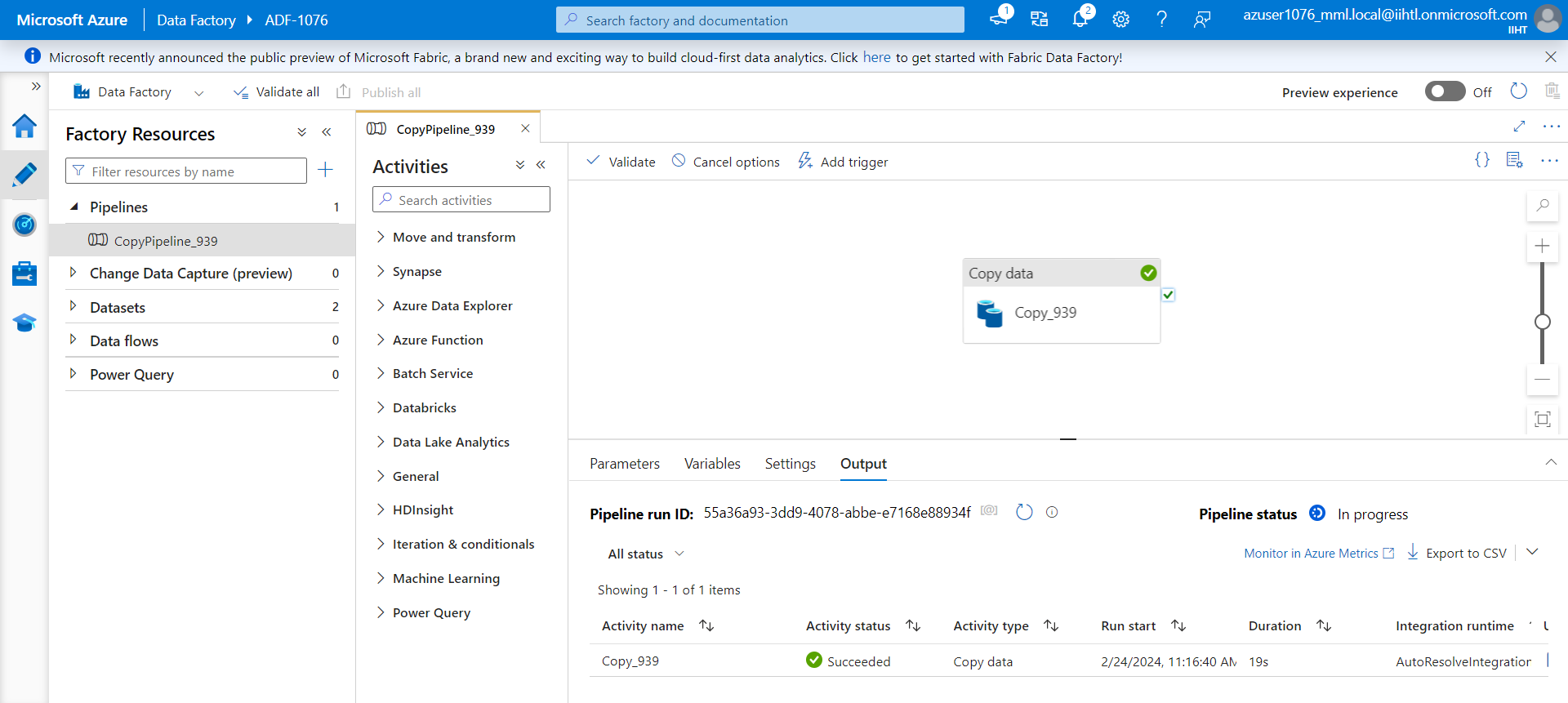
Creating Pipeline:



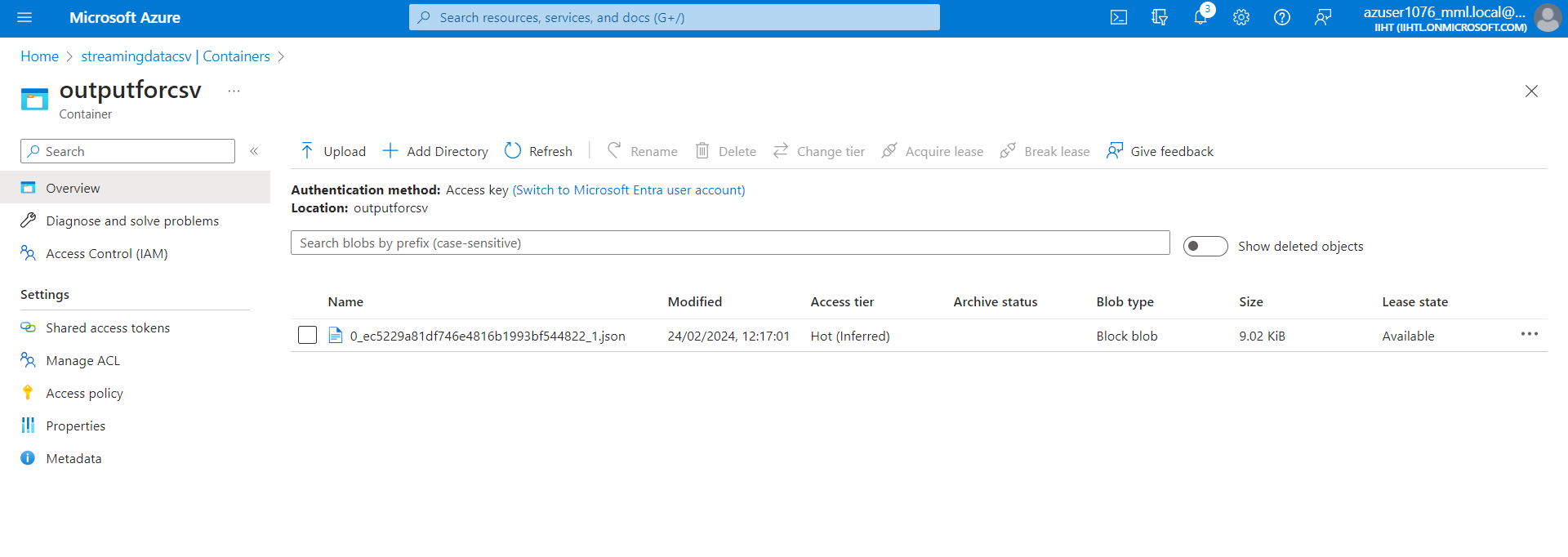
6.5) creating a pipeline to initiate the copy activity



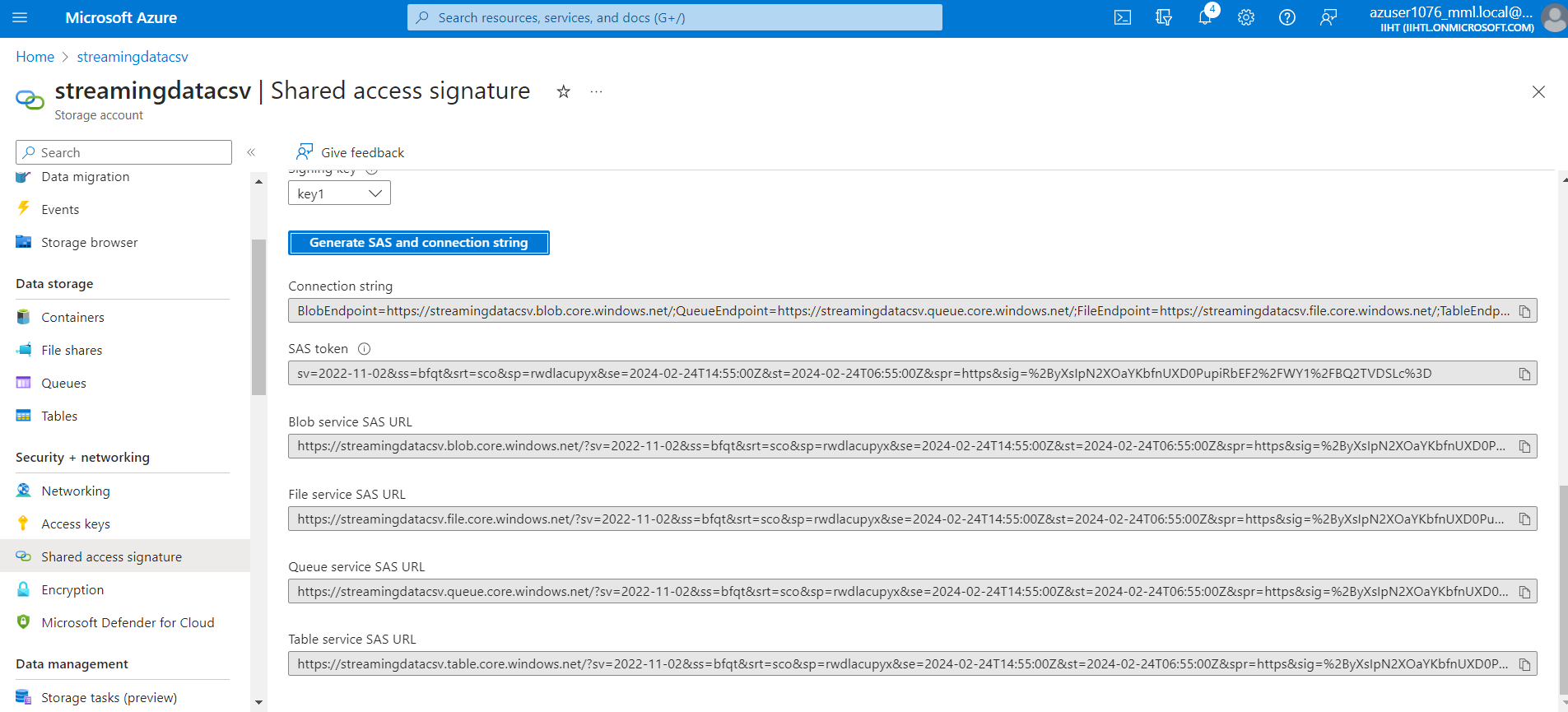
select debug option, after the successfully completion of the activity we can able to see the activity status as succeeded



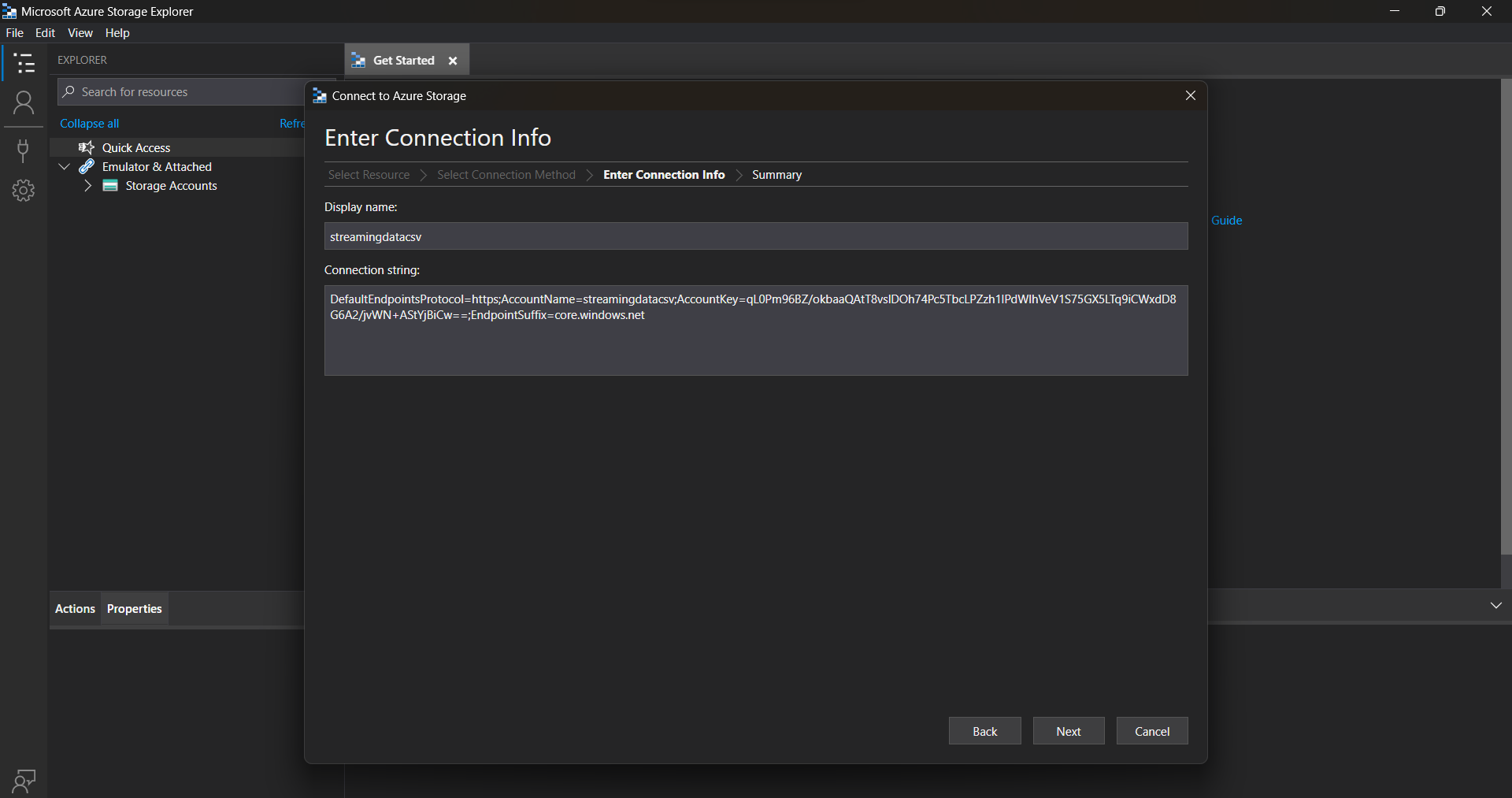
Data is been updated in Sink dataset



6.6)Generating SAS & connection string



Establishing the connection with explorer:

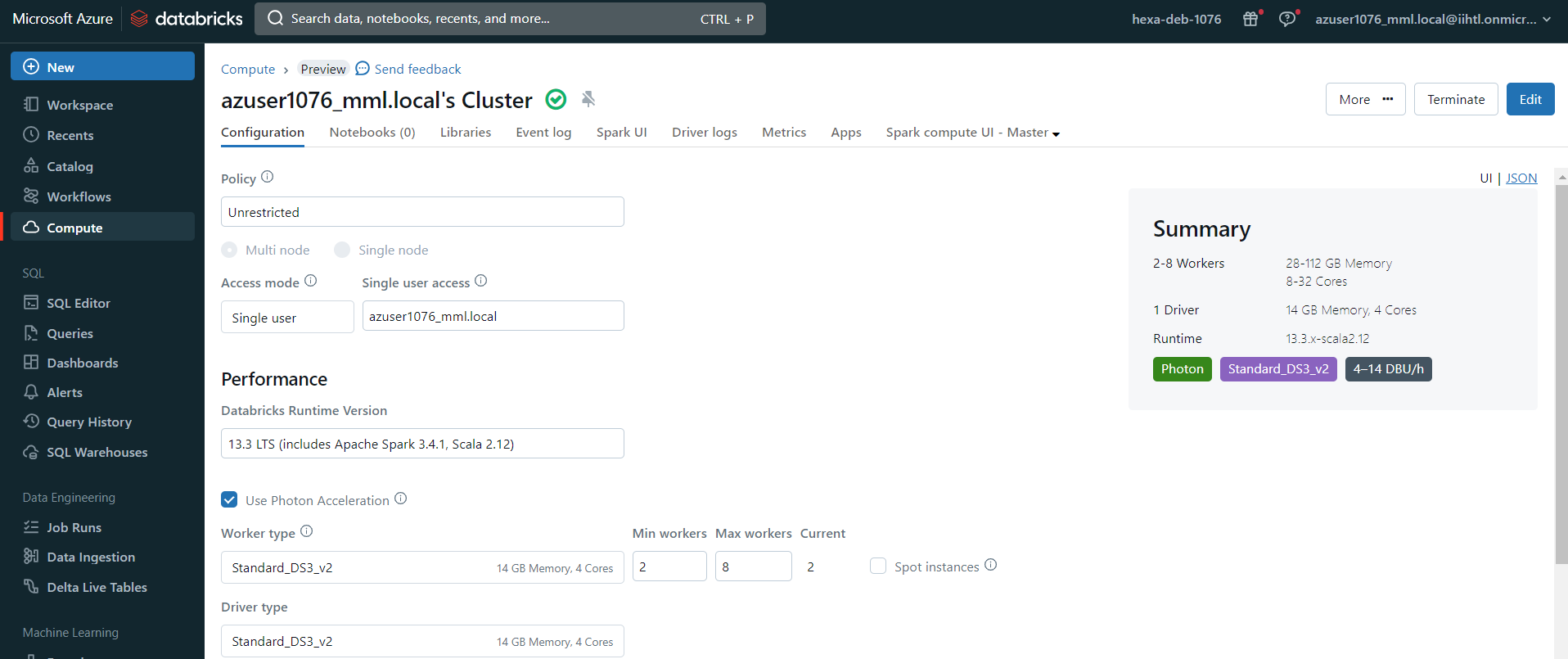


Connectiion was established successfully:

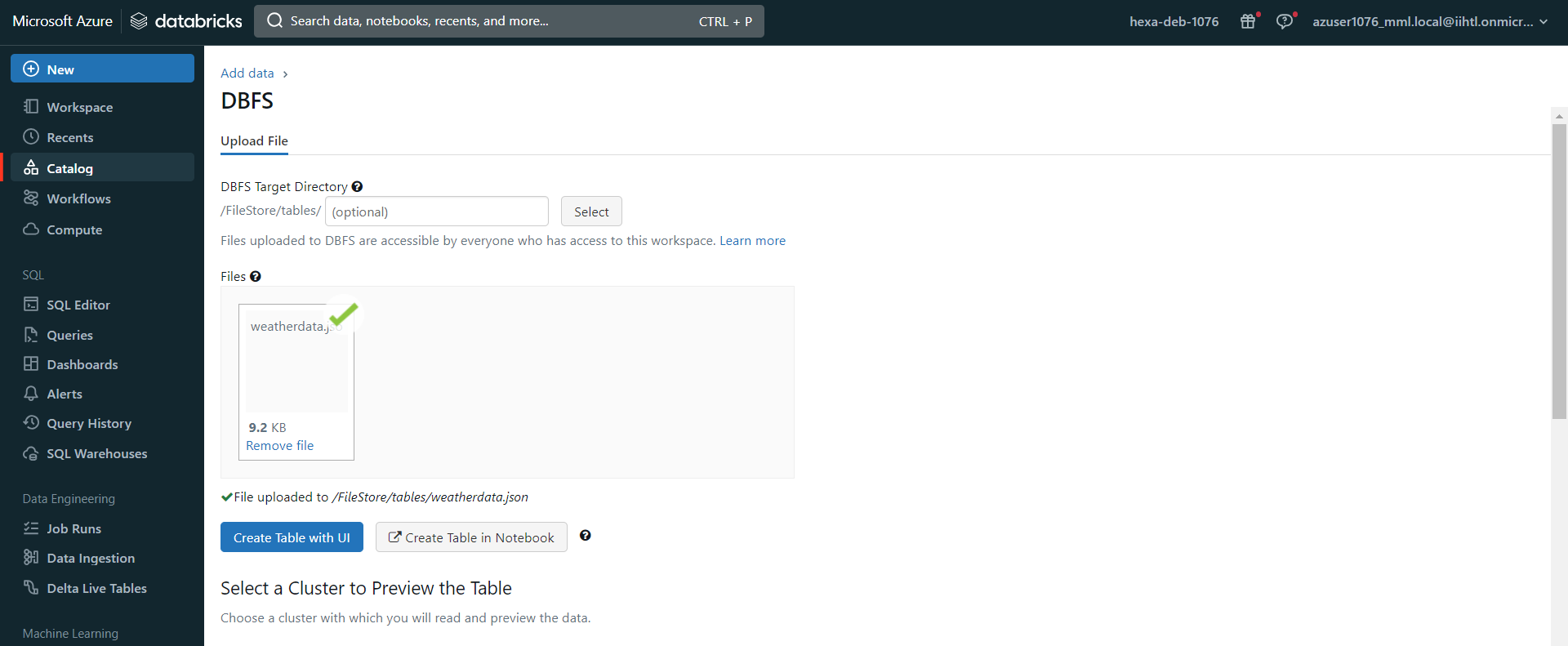


Azure Databricks

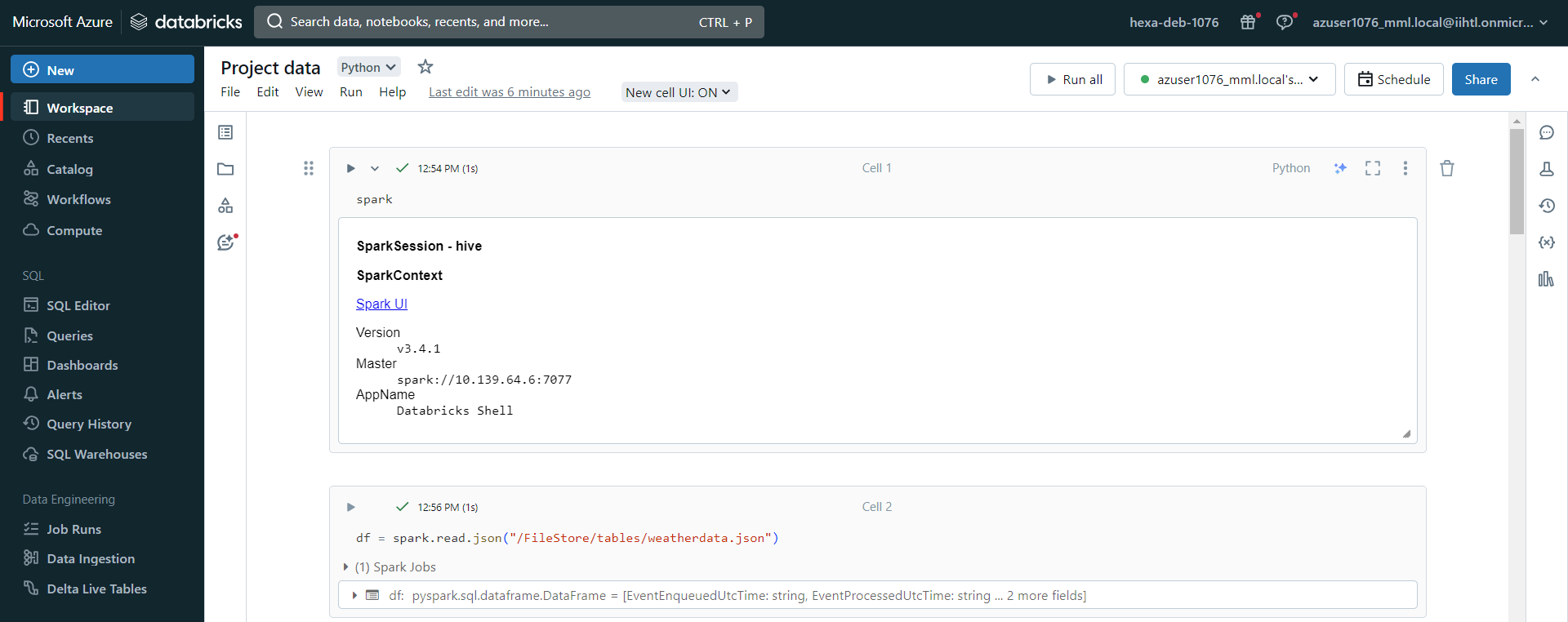
Creating a cluster in the Azure Databricks workspace



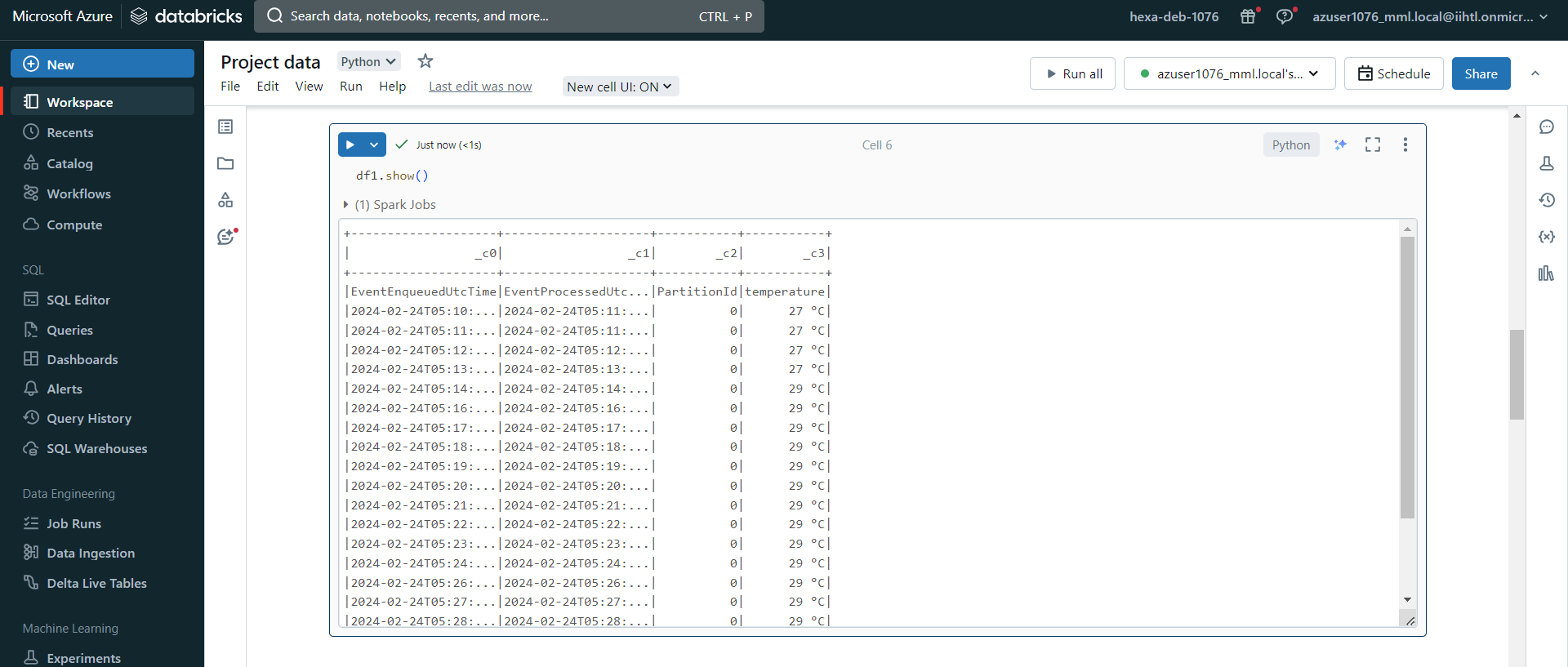
Then uploading the file by using create a new table option and select Create Table in Notebook

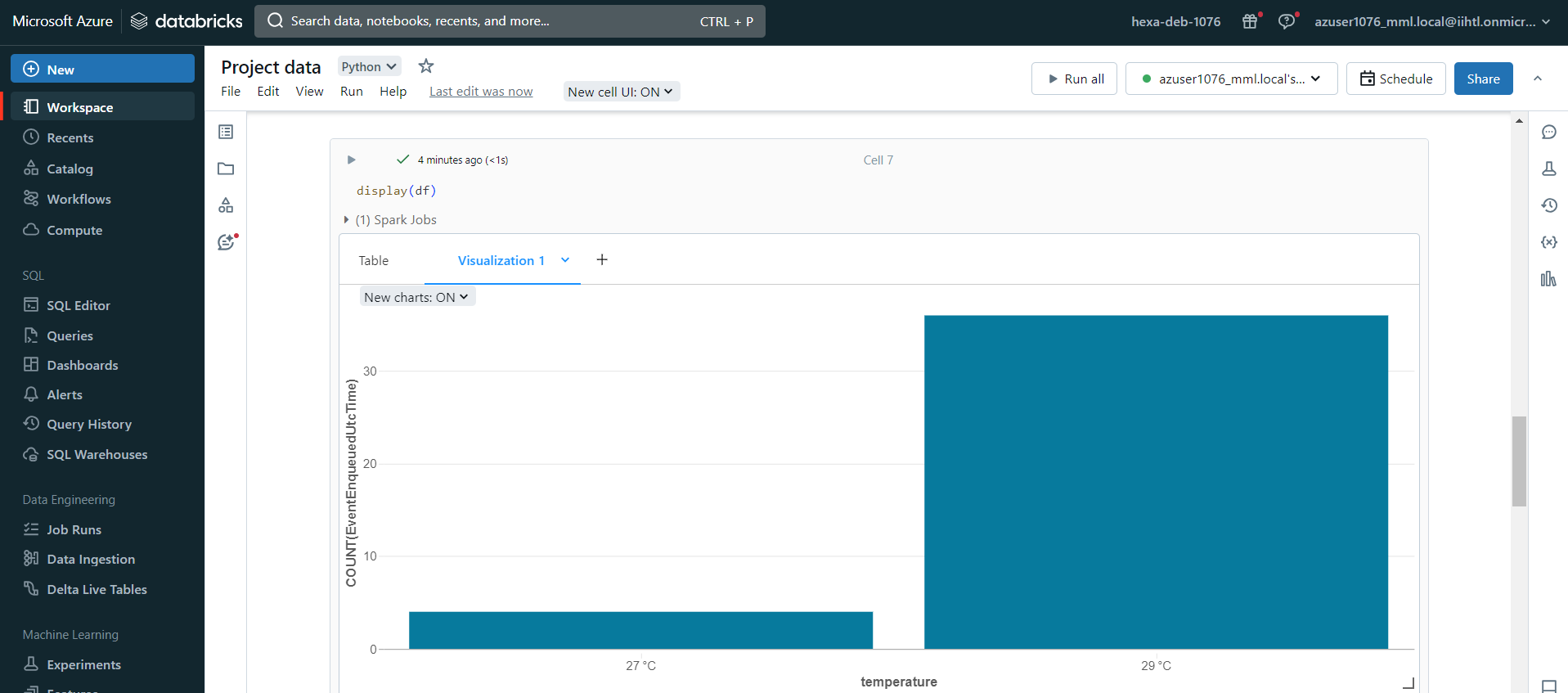


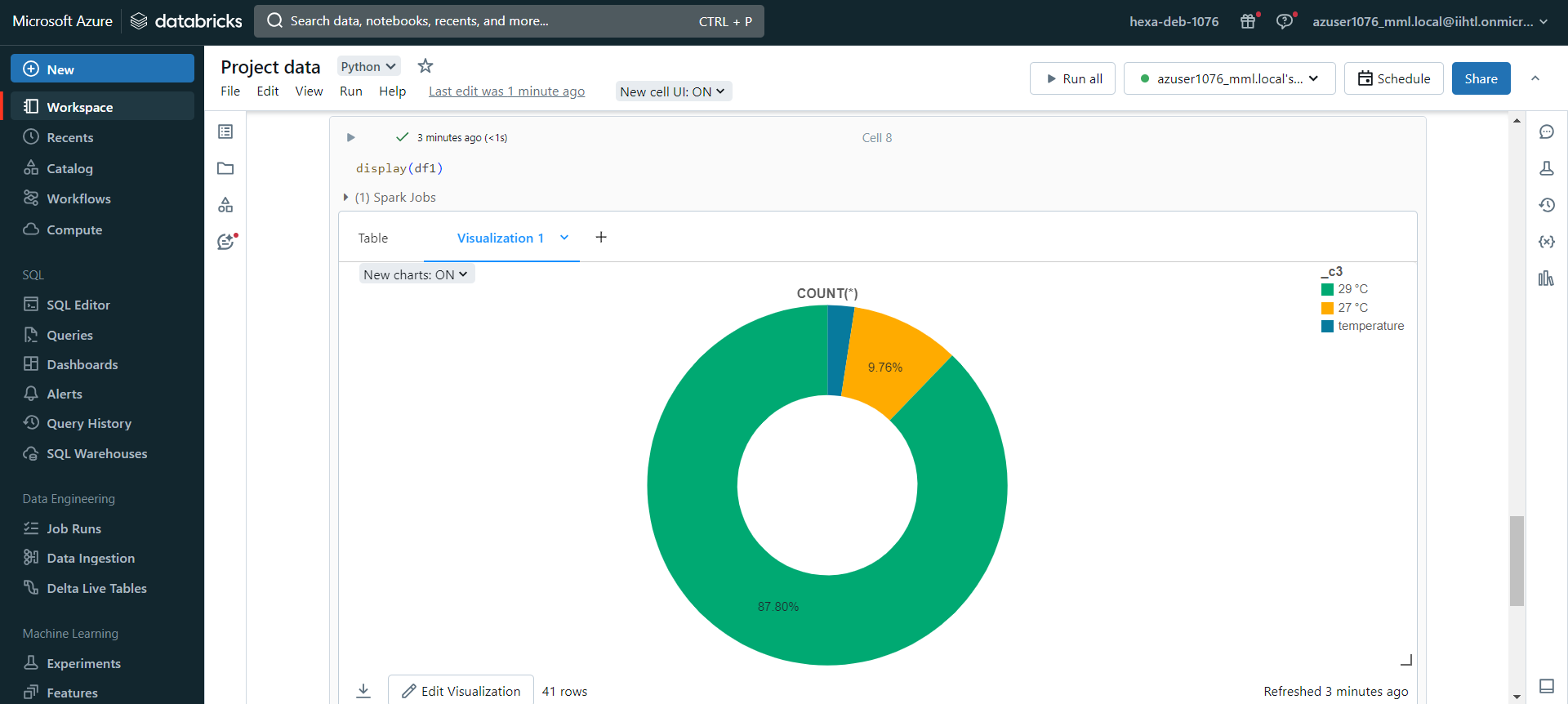
**Analysing the uploaded data**











1. **Conclusion**:

Setting up an end-to-end pipeline with Azure Data Factory for ingesting streaming data and Azure Databricks for real-time processing and analysis is a powerful approach for handling streaming data workflows. This setup allows you to efficiently ingest, process, and analyze data in real time, enabling you to derive valuable insights and make timely decisions. The flexibility and scalability of these Azure services enable you to adapt to changing data requirements and scale your pipeline as your data volume grows. Overall, this combination of Azure Data Factory and Azure Databricks provides a reliable and efficient solution for handling streaming data processing tasks.