uOttawa

ELG5166 Cloud Analytics

Assignment 3

Instructor: Dr. Benjamin Eze

Group: 4

Student Name: Ali El-Sherif ID: 300327246

Student Name: Basma Abd-Elwahab ID: 300327209

Student name: Abdelrhman Rezkallah ID: 300327290

Student Name: Abdulrahman Ahmed ID: 300327218

Table of Contents

Personal Ethics & Academic Integrity Statement

By typing in my name and student ID on this form and submitting it electronically, I am attesting to the fact that I have reviewed not only my work but the work of my team member, in its entirety.

I attest to the fact that my work in this project adheres to the fraud policies as outlined in the Academic Regulations in the University's Graduate Studies Calendar. I further attest that I have knowledge of and have respected the "Beware of Plagiarism" brochure for the university. To the best of my knowledge, I also believe that each of my group colleagues has also met the aforementioned requirements and regulations. I understand that if my group assignment is submitted without a completed copy of this Personal Work Statement from each group member, it will be interpreted by the school that the missing student(s) name is confirmation of non-participation of the aforementioned student(s) in the required work. We, by typing in our names and student IDs on this form and submitting it electronically,

• warrant that the work submitted herein is our own group members' work and not the work of others.

• acknowledge that we have read and understood the University Regulations on Academic Misconduct.

• acknowledge that it is a breach of University Regulations to give or receive unauthorized and/or unacknowledged assistance on a graded piece of work.

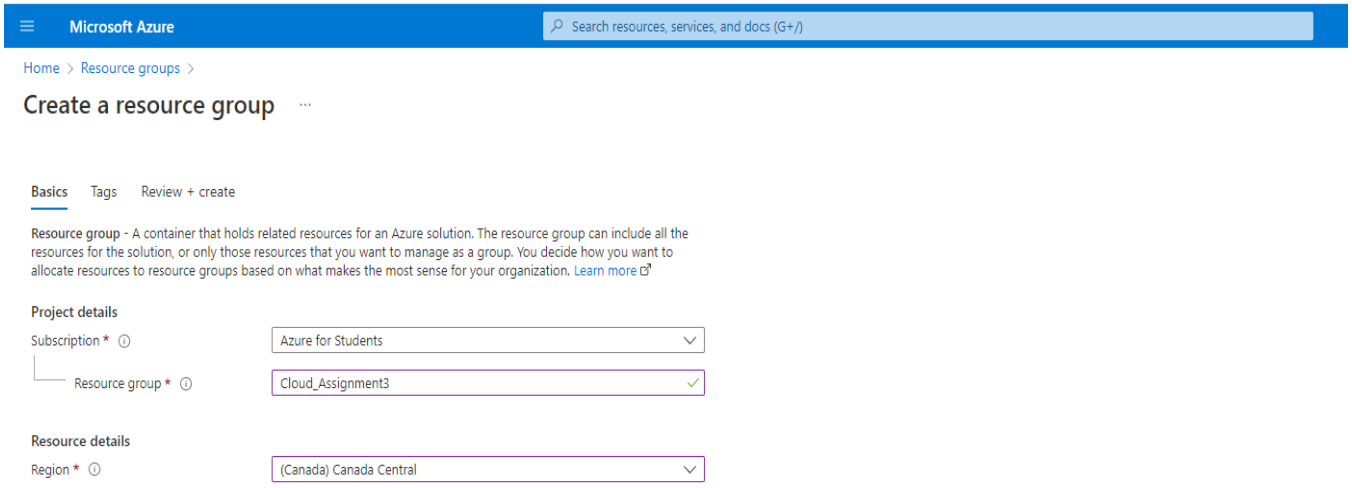
Part 1: Event Hub Analytics

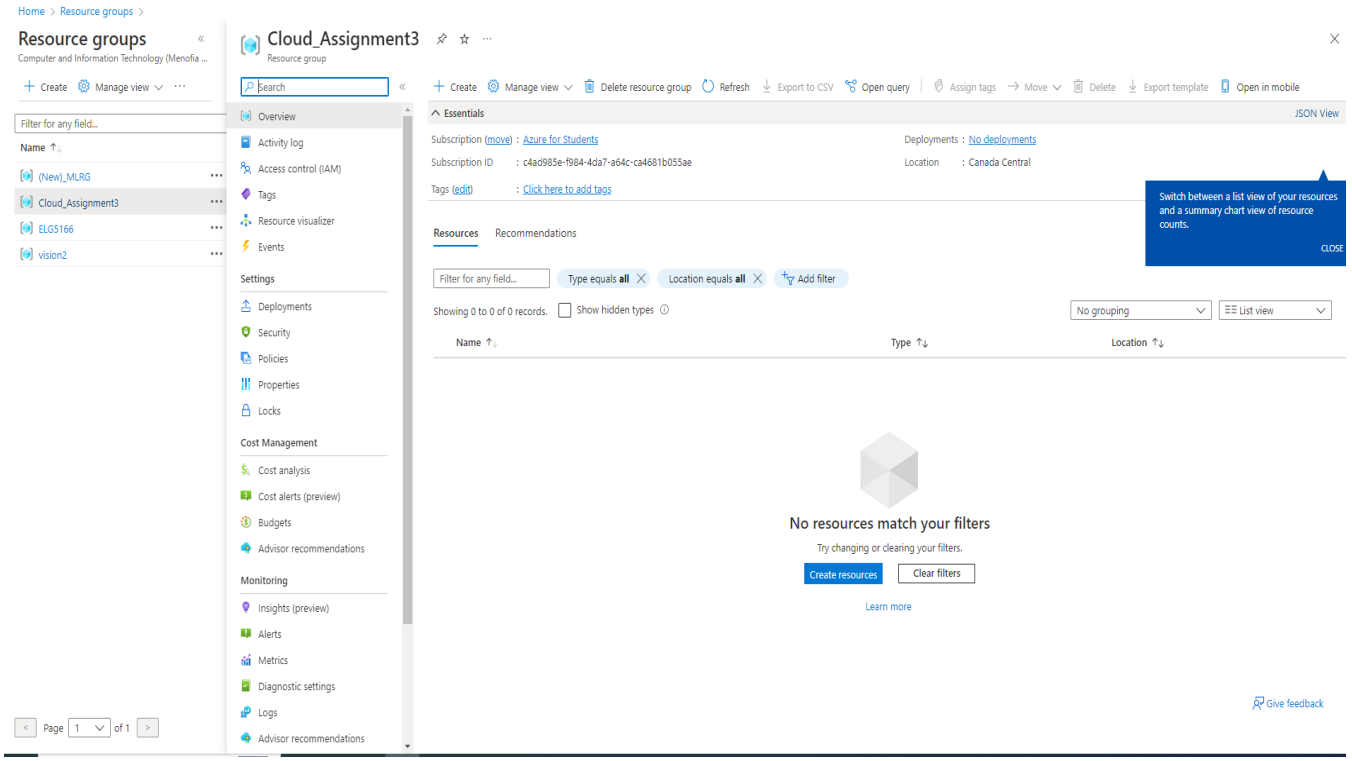
Setting up the environment:

1- Signup into Azure as a student and claim the credit.
2- Install visual studio community edition.
3- Install visual studio code dependencies.

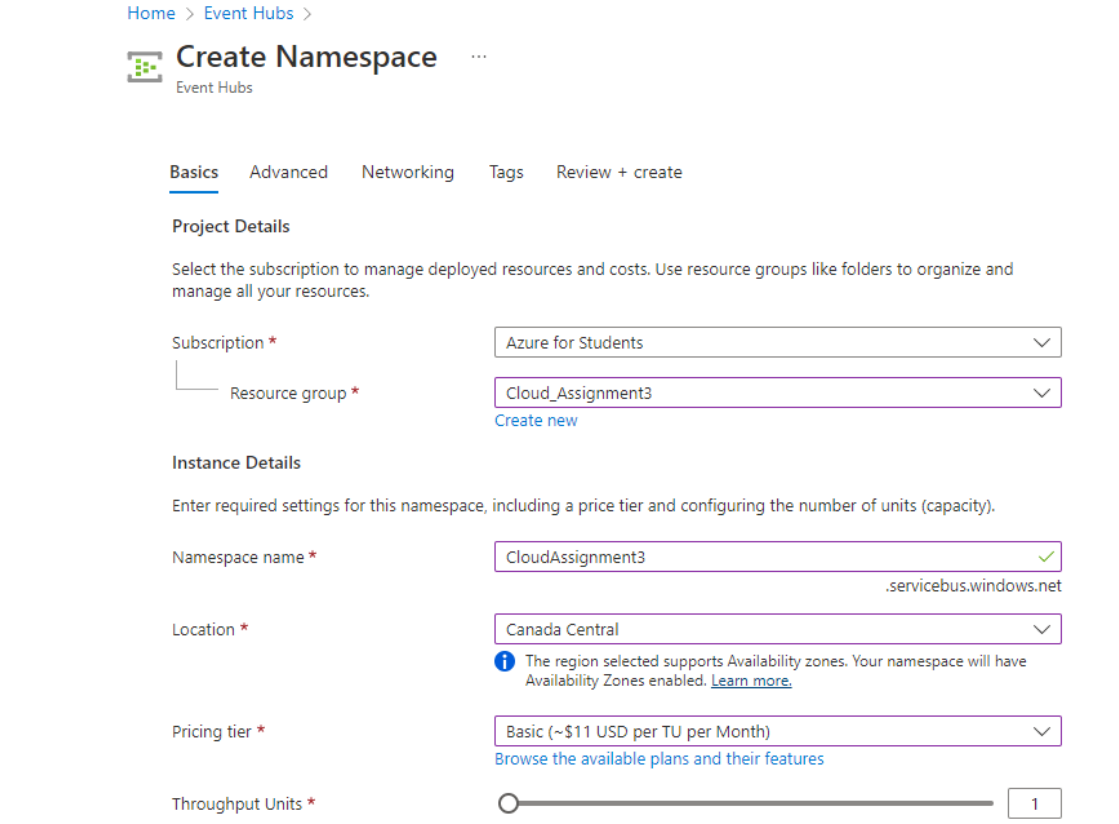
After setting up the environment:

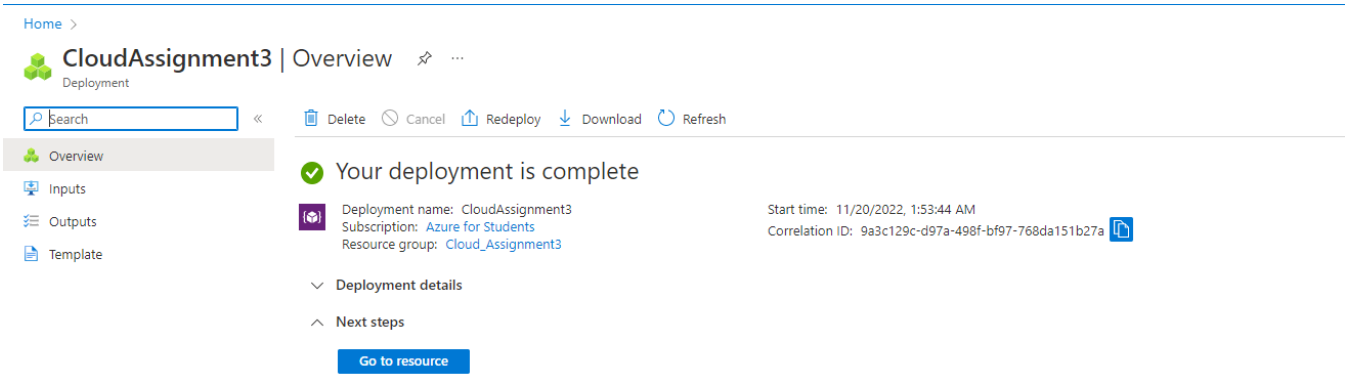
1- Create a resource group:



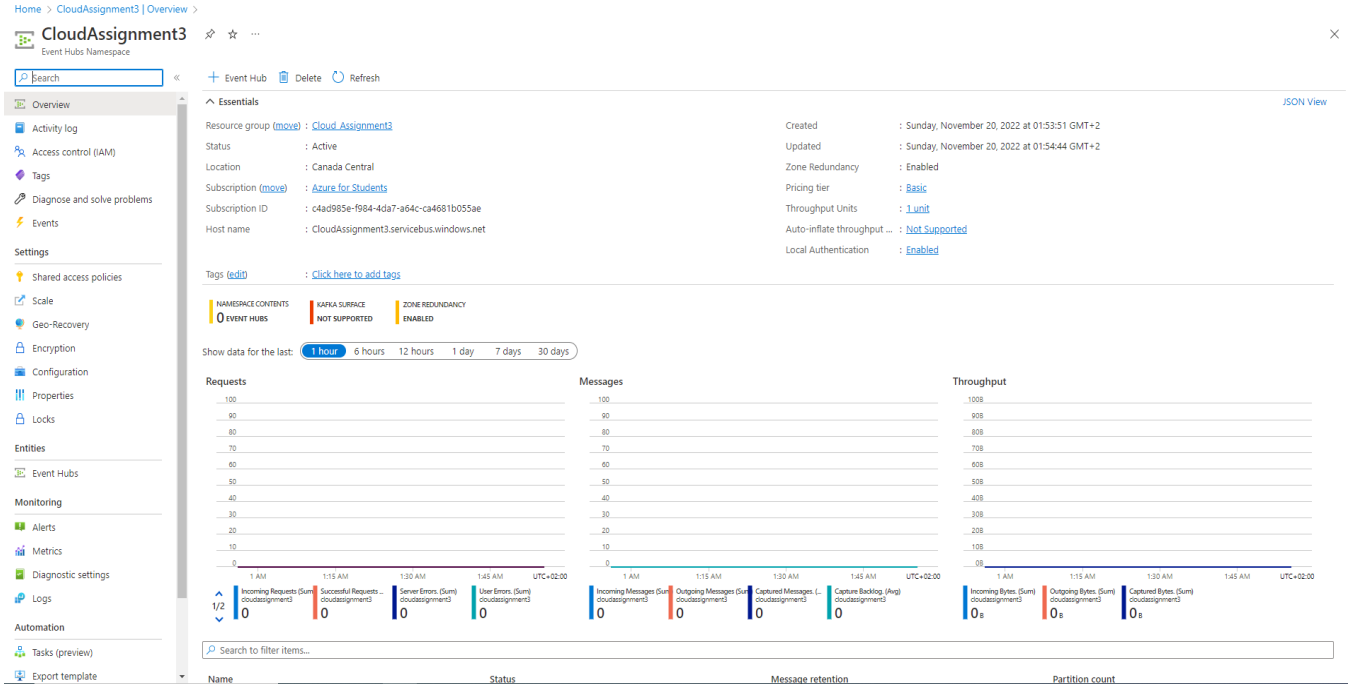


2- Create Event Hub Namespace:

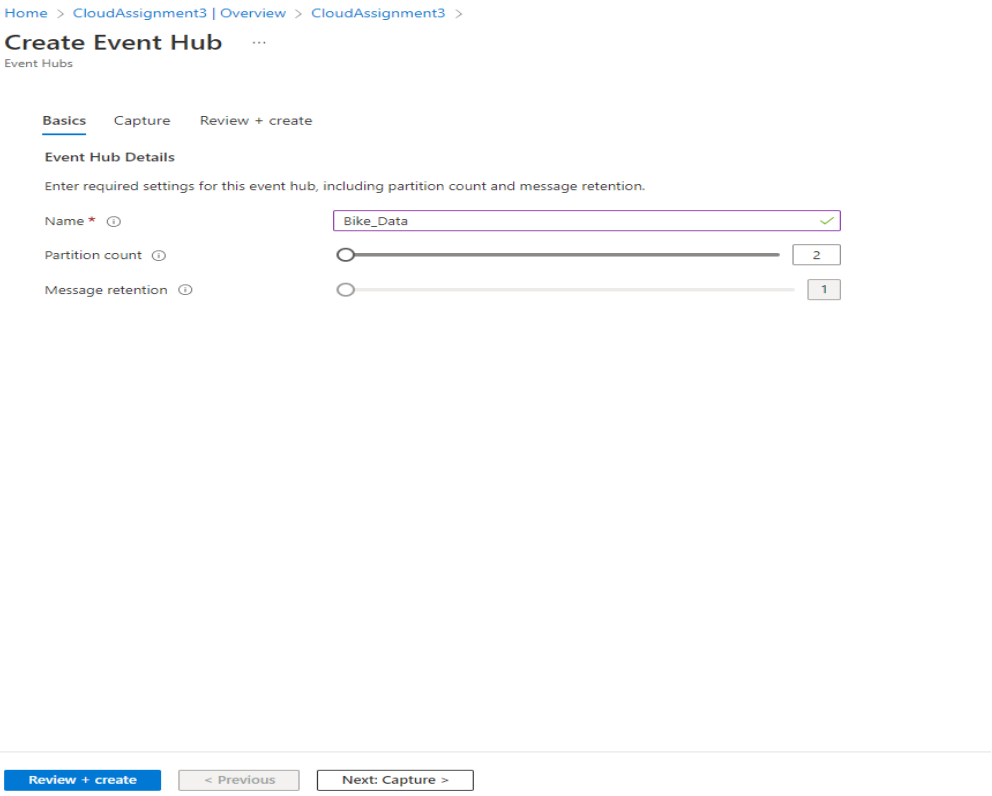


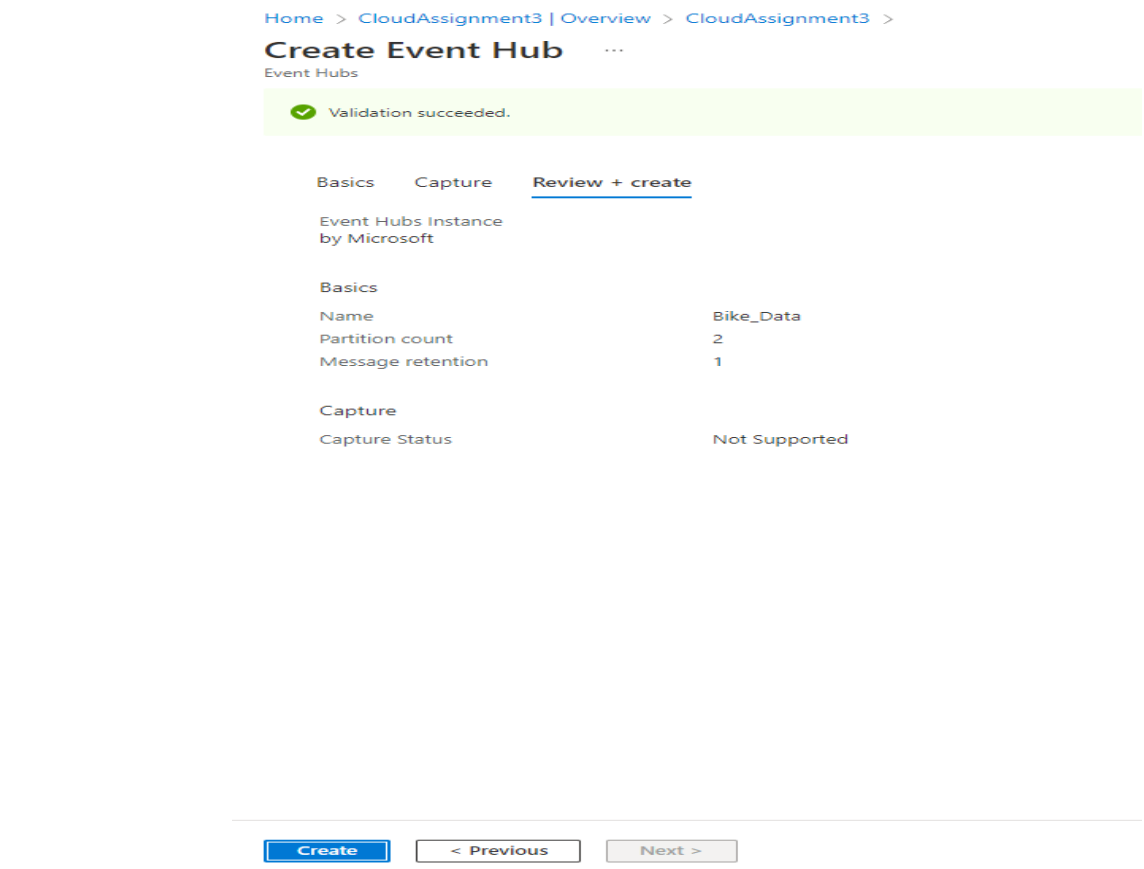


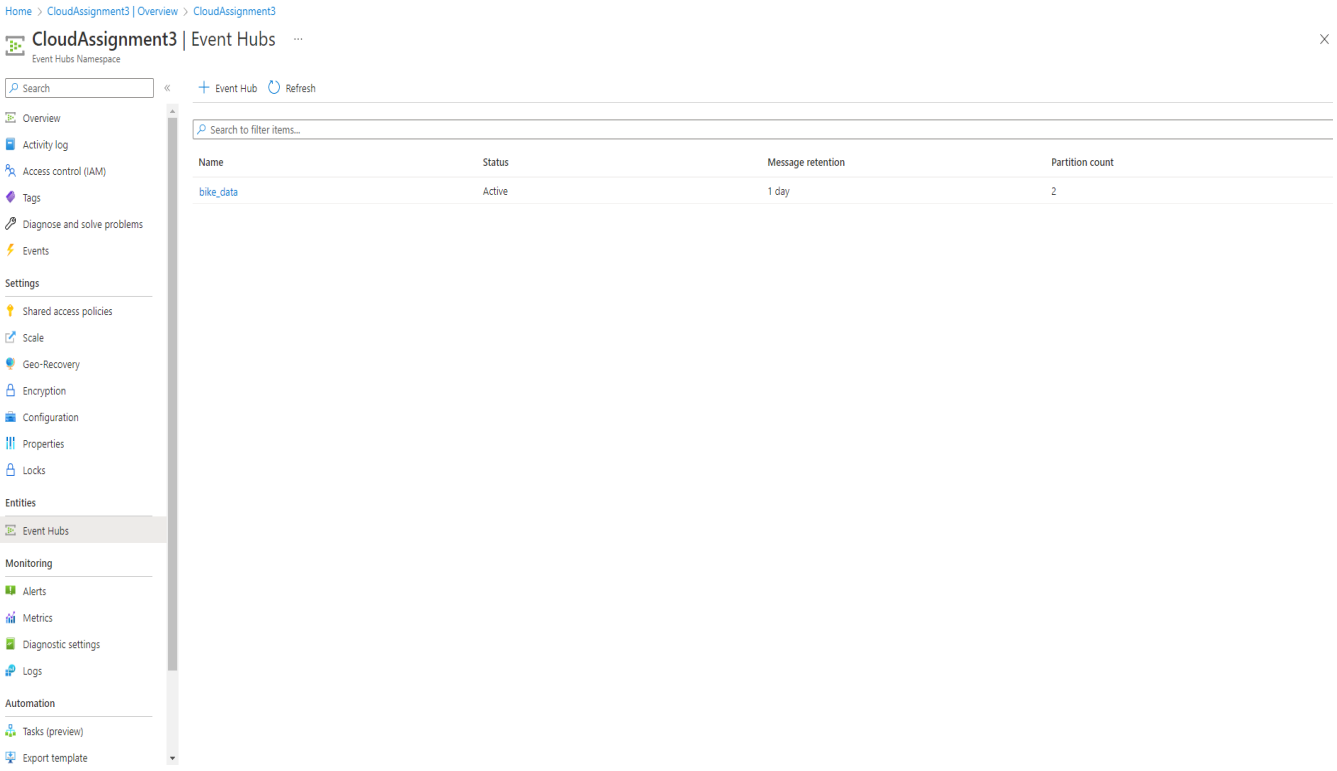
The Namespace is empty, because no streams done yet.



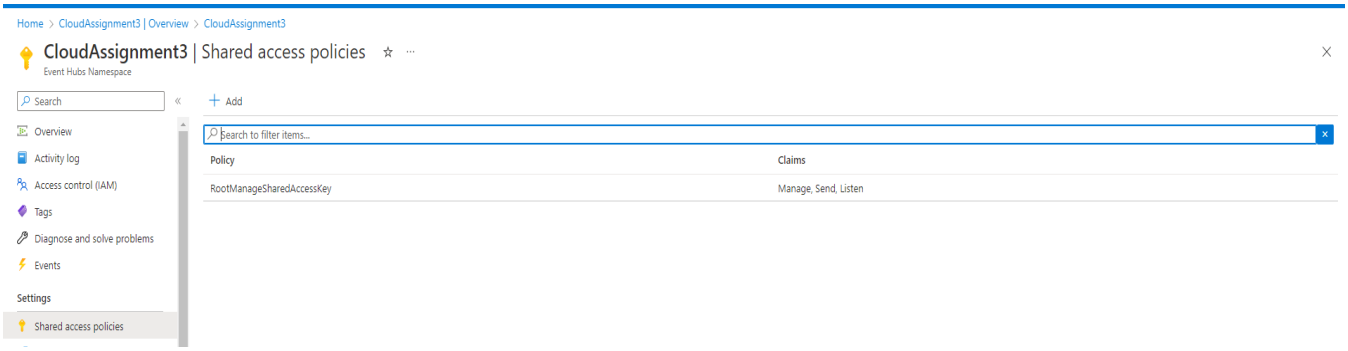
3- Create Event Hub for bike data:



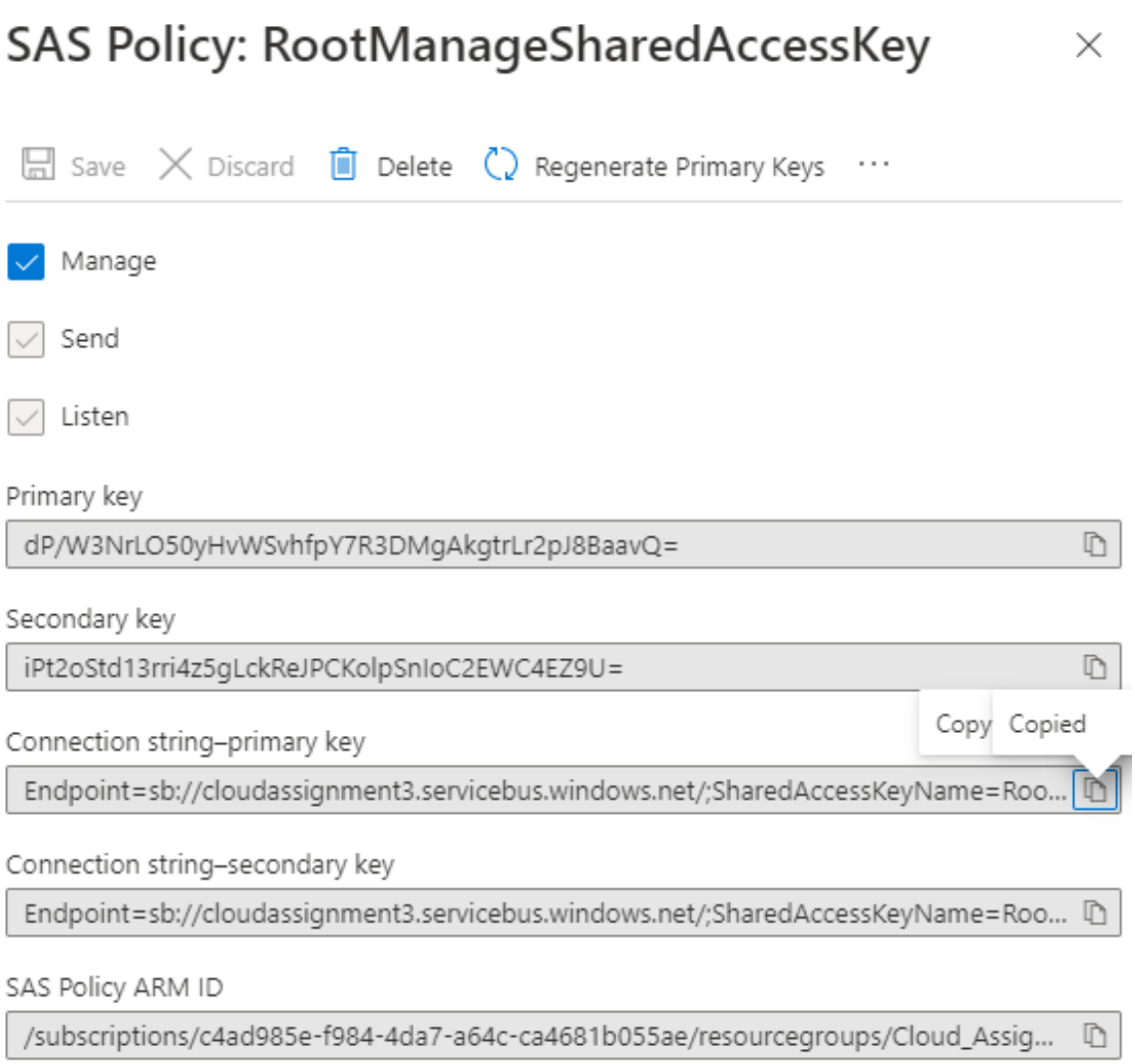

Home > CloudAssignment3 | Overview > CloudAssignment3 >
Create Event Hub
Event Hubs
Validation succeeded.
Basics
Capture
Review + create
Event Hubs Instance
by Microsoft
Basics
Name
Bike_Data
Partition count
2
Message retention
1
Capture
Capture Status
Not Supported
Create
< Previous
Next >


Home > CloudAssignment3 | Overview > CloudAssignment3
CloudAssignment3 | Event Hubs
Event Hubs Namespace
Search
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Events
Settings
Shared access policies
Scale
Geo-Recovery
Encryption
Configuration
Properties
Locks
Entities
Event Hubs
Monitoring
Alerts
Metrics
Diagnostic settings
Logs
Automation
Tasks (preview)
Export template
Event Hub
Refresh
Search to filter items...
Name
Status
Message retention
Partition count
bike_data
Active
1 day
2

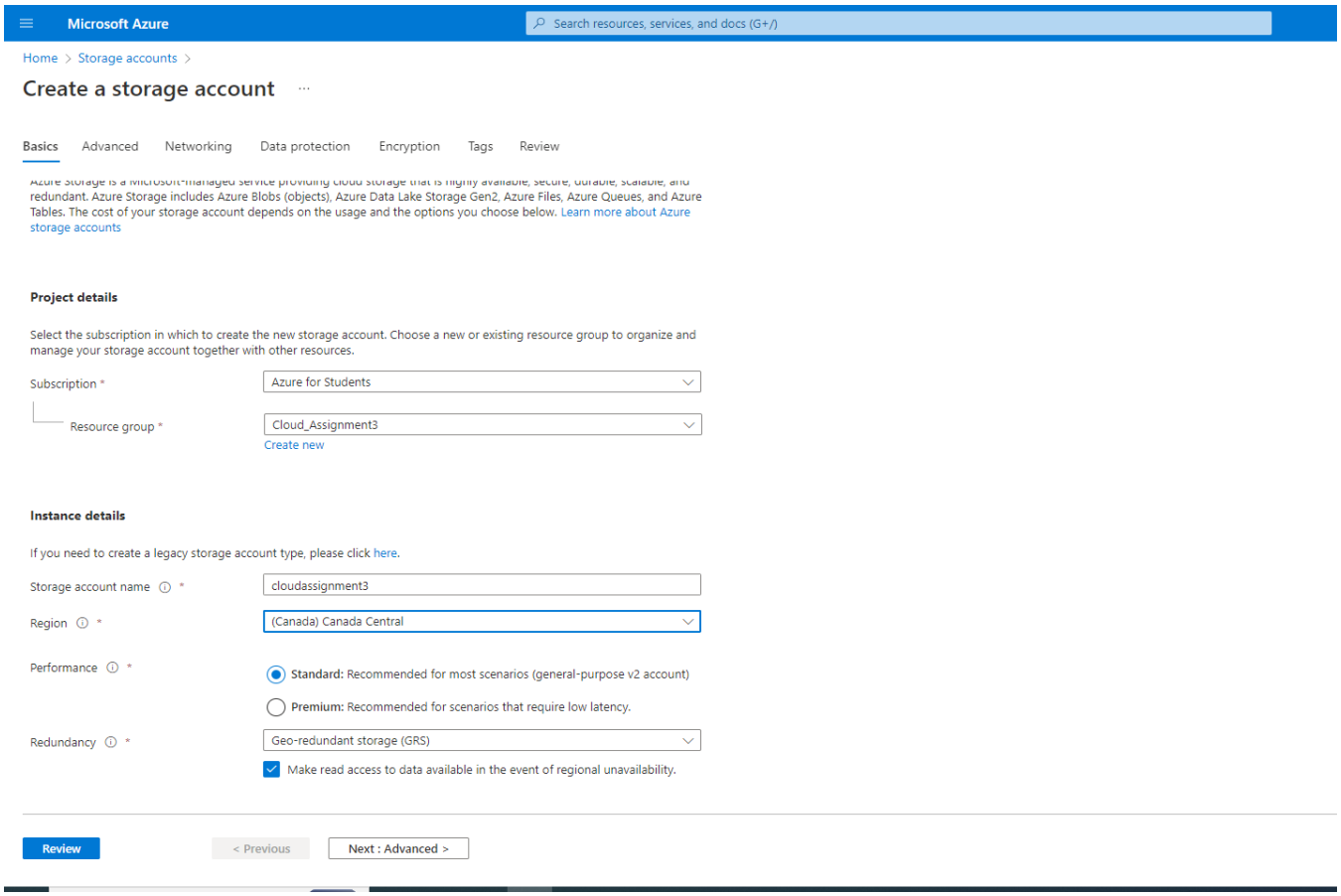
4- Go to the shared access policies:

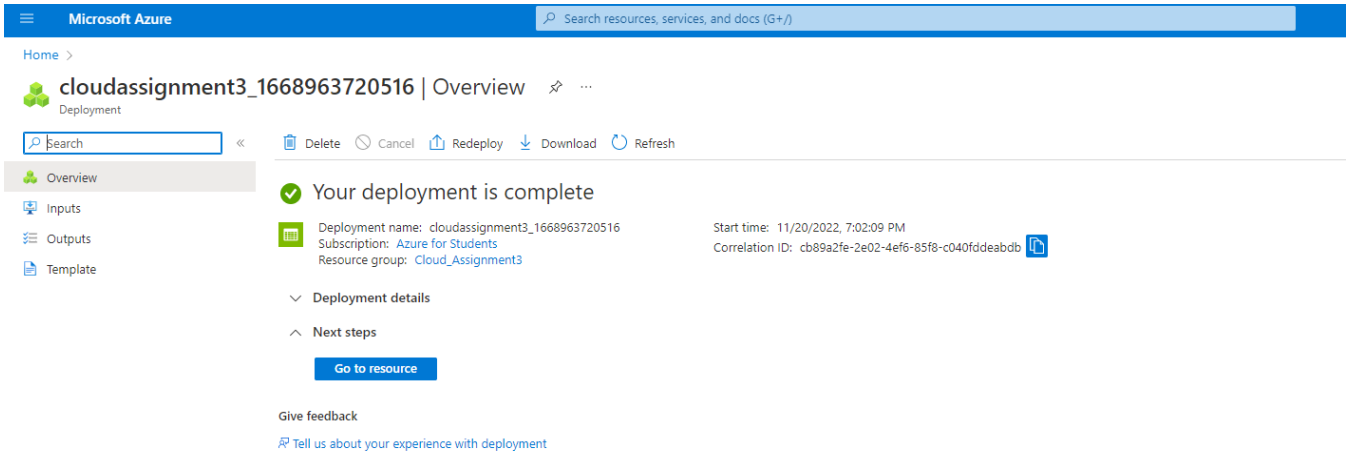


5- Copy the connection key primary string to use in the visual studio:

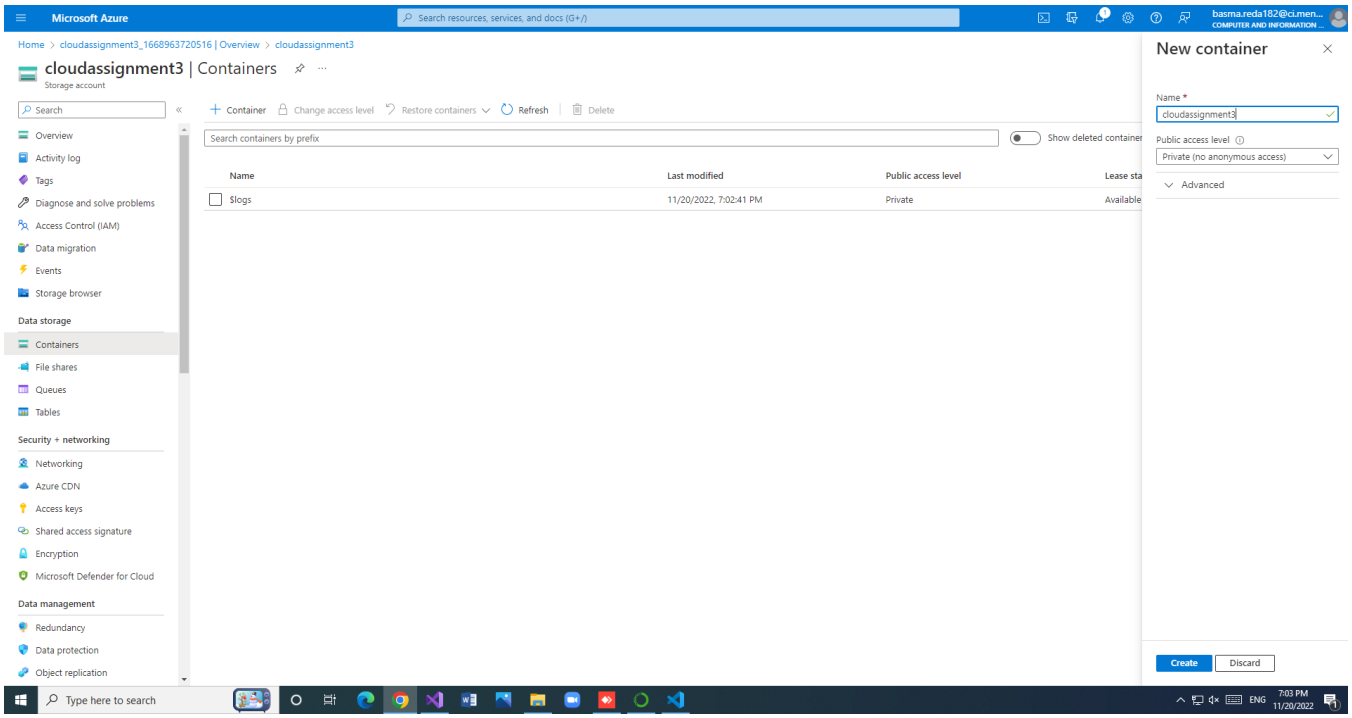


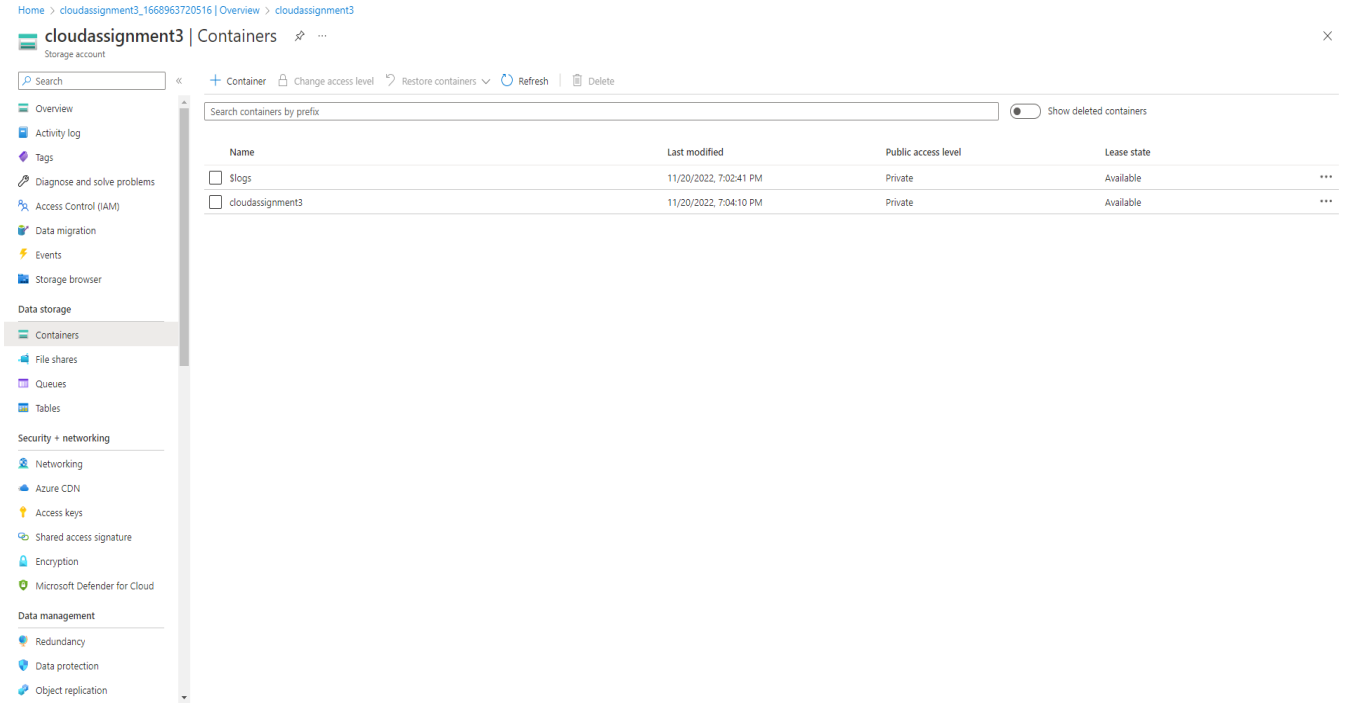
6- Create a storage account to store the bike data:



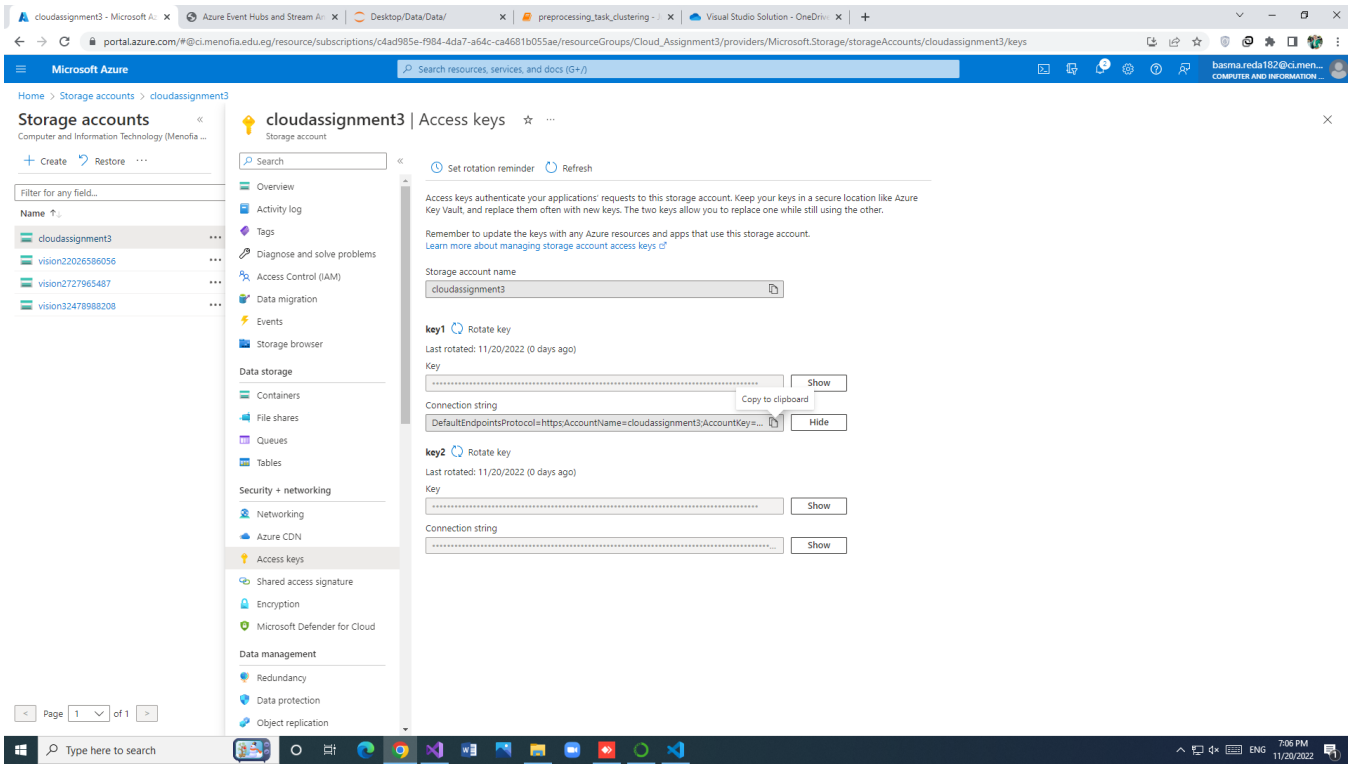


7- Create a container:

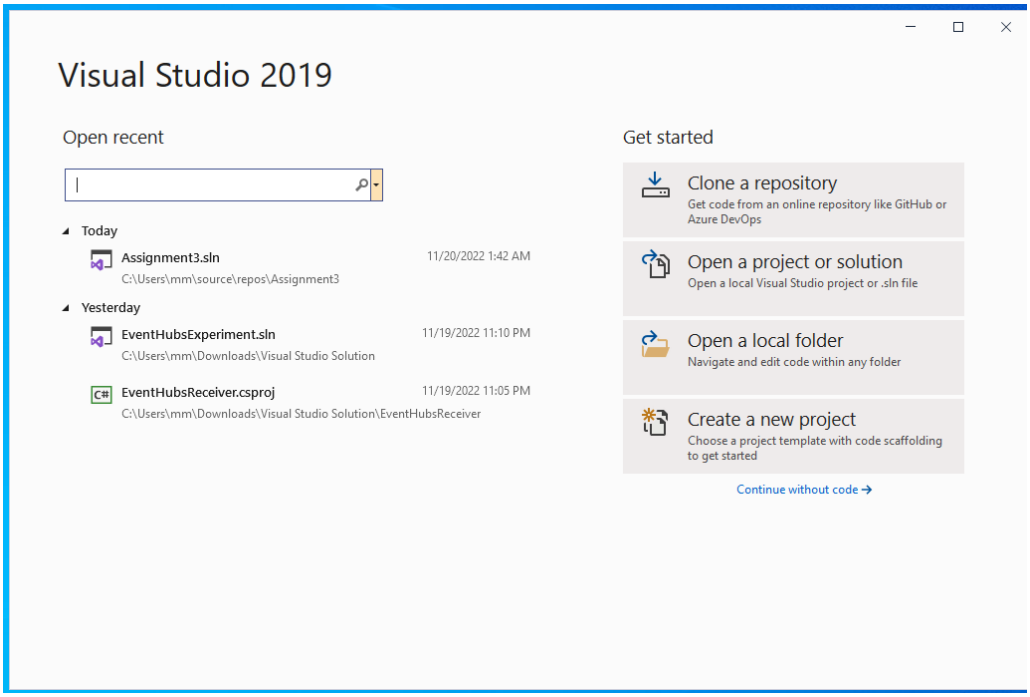




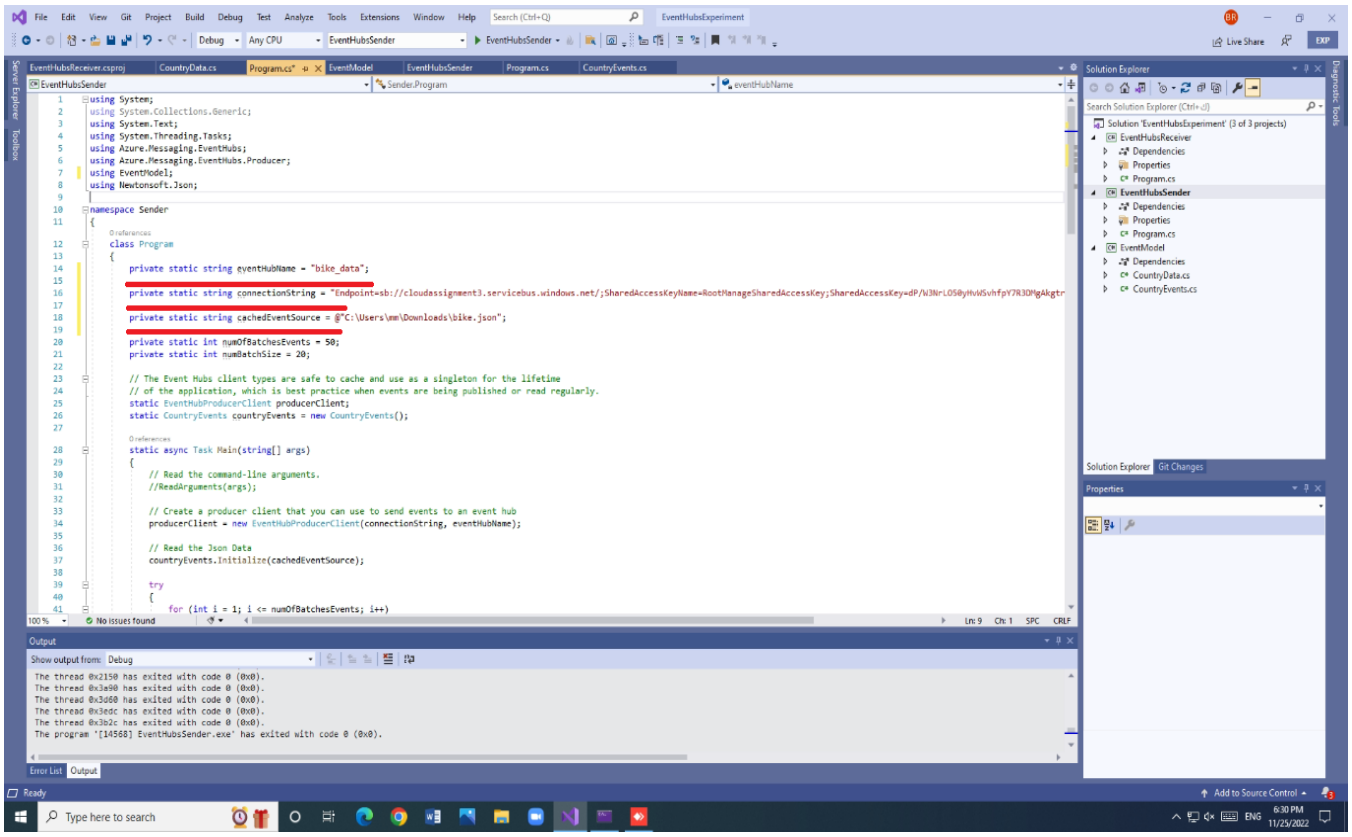
8- Get the connection string access key of the cloud storage and save this to use:



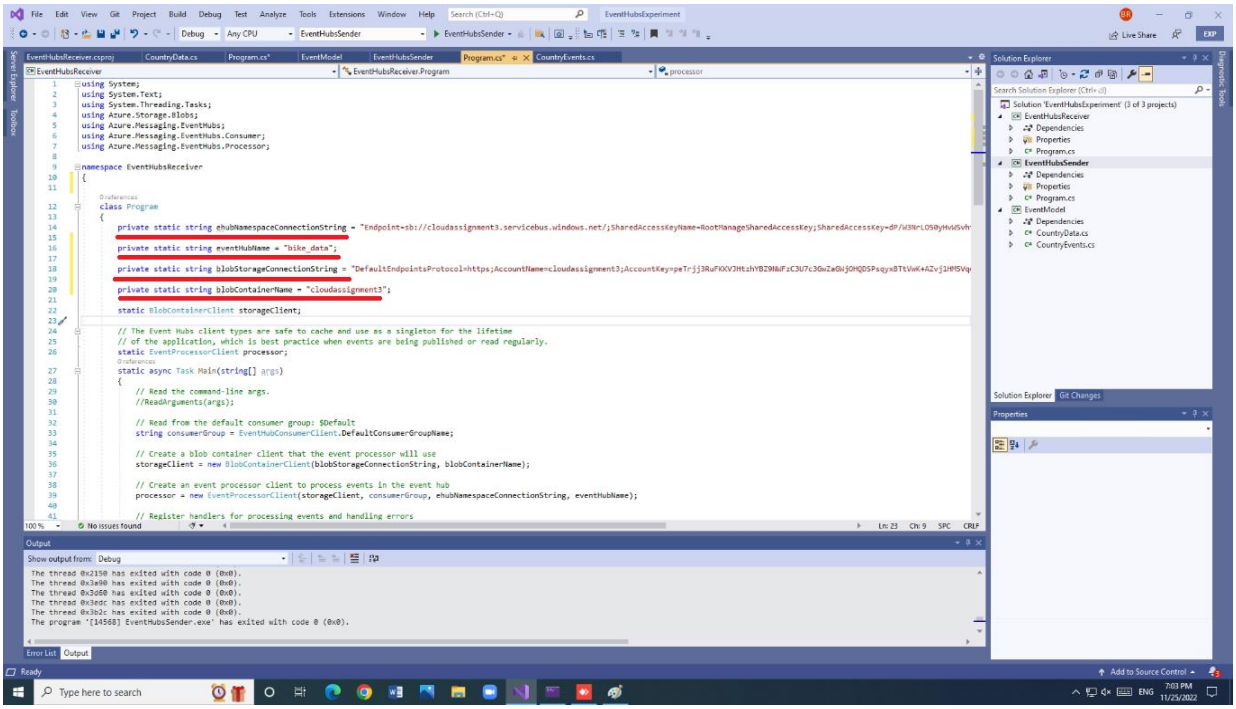
9- Open the Visual studio to open the project:



10- Use the solution created before in the lecture and modify the connection string that copied before, set the event hub namespace, and the cached event source:

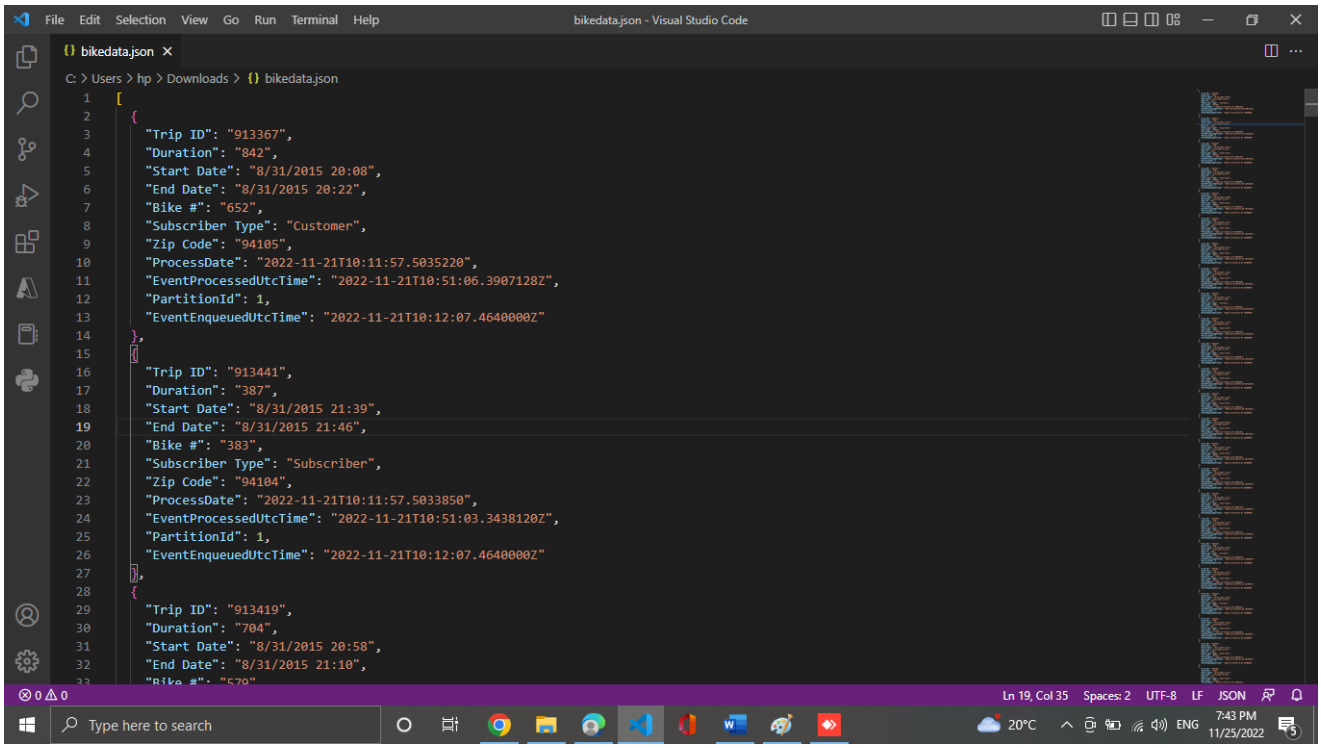


11- Set up the connection of the receiver to receive the data:

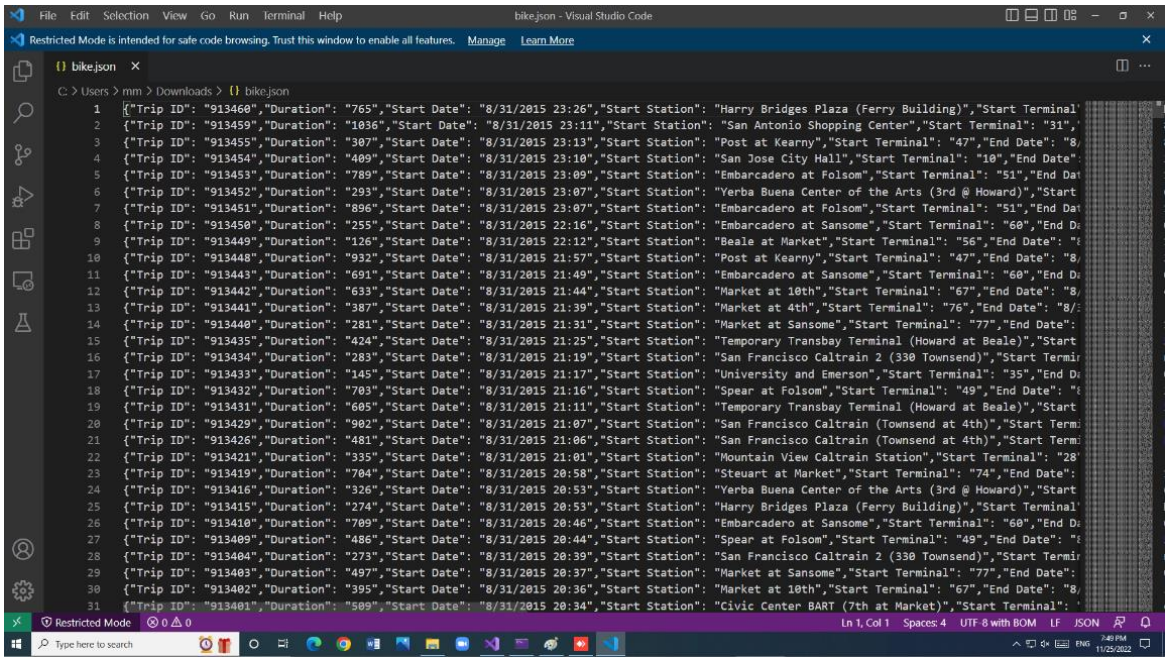


A) Broadcast bike rental events in batches of 20 trip entries, and publish at least 25 events at 10-second intervals per batch. Each event should include the following fields –Process Time, Trip ID, Start Date/Time, Duration, Bike #, Subscriber Type, and Zip Code.

1- Take a look at the Bike data JSON file:



2- Modify the schema of the Bike data JSON file to be in an inline format as we don't need to modify the code to read the data in the old format:

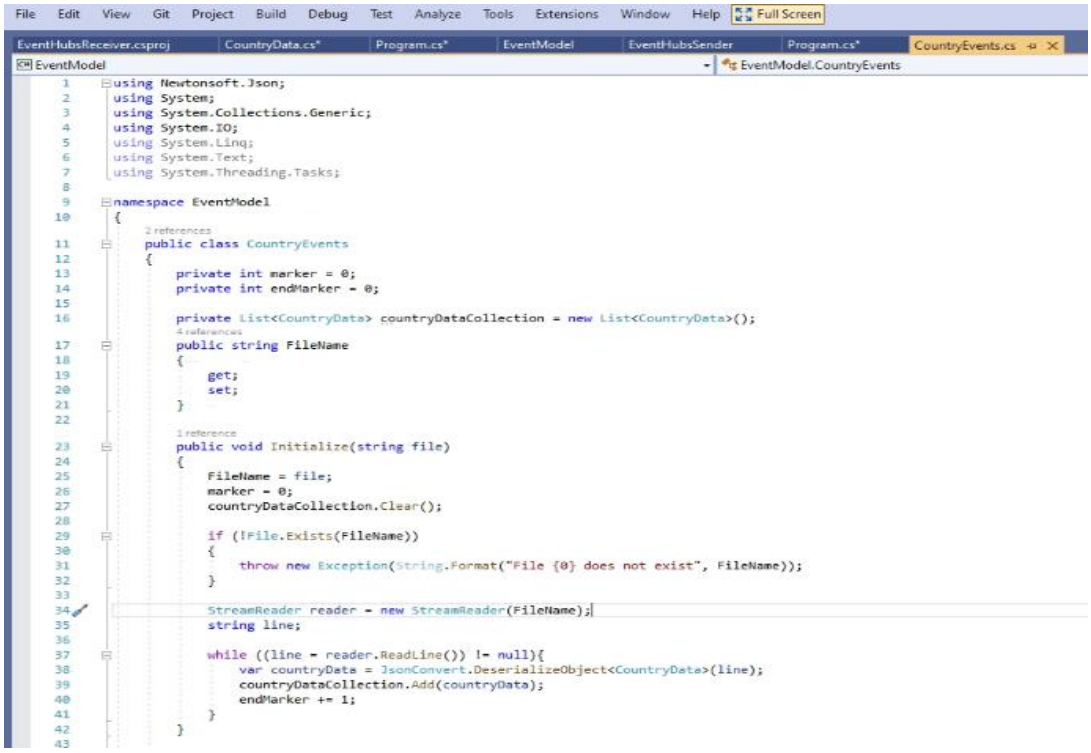


3- Change the class data to be able to send these variables:

Use JsonProperty to make the class able to read the variables with a space in between.

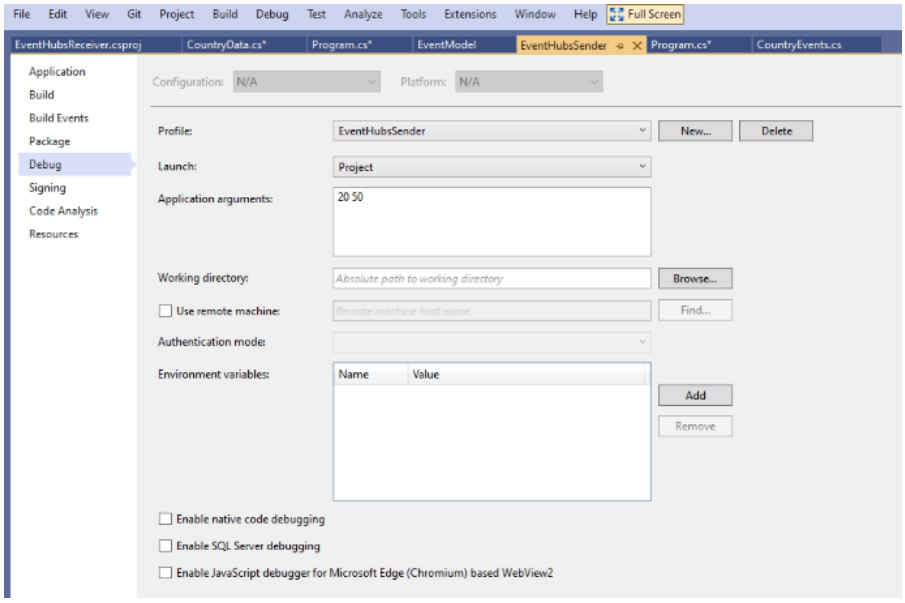
```
using System;
using Newtonsoft.Json;

namespace EventModel
{
    5 references
    public class CountryData
    {
        [JsonProperty("Trip ID")]
        0 references
        public string TripID { get; set; }

        0 references
        public string Duration { get; set; }
        [JsonProperty("Start Date")]

        0 references
        public string Start_Date { get; set; }
        [JsonProperty("End Date")]

        0 references
        public string EndDate { get; set; }
        [JsonProperty("Bike #")]

        0 references
        public string Bike { get; set; }
        [JsonProperty("Subscriber Type")]

        0 references
        public string SubscriberType { get; set; }
        [JsonProperty("Zip Code")]

        0 references
        public string ZipCode { get; set; }

        1 reference
        public DateTime ProcessDate { get; set; }
    }
}
```

4- Read the data line by line:

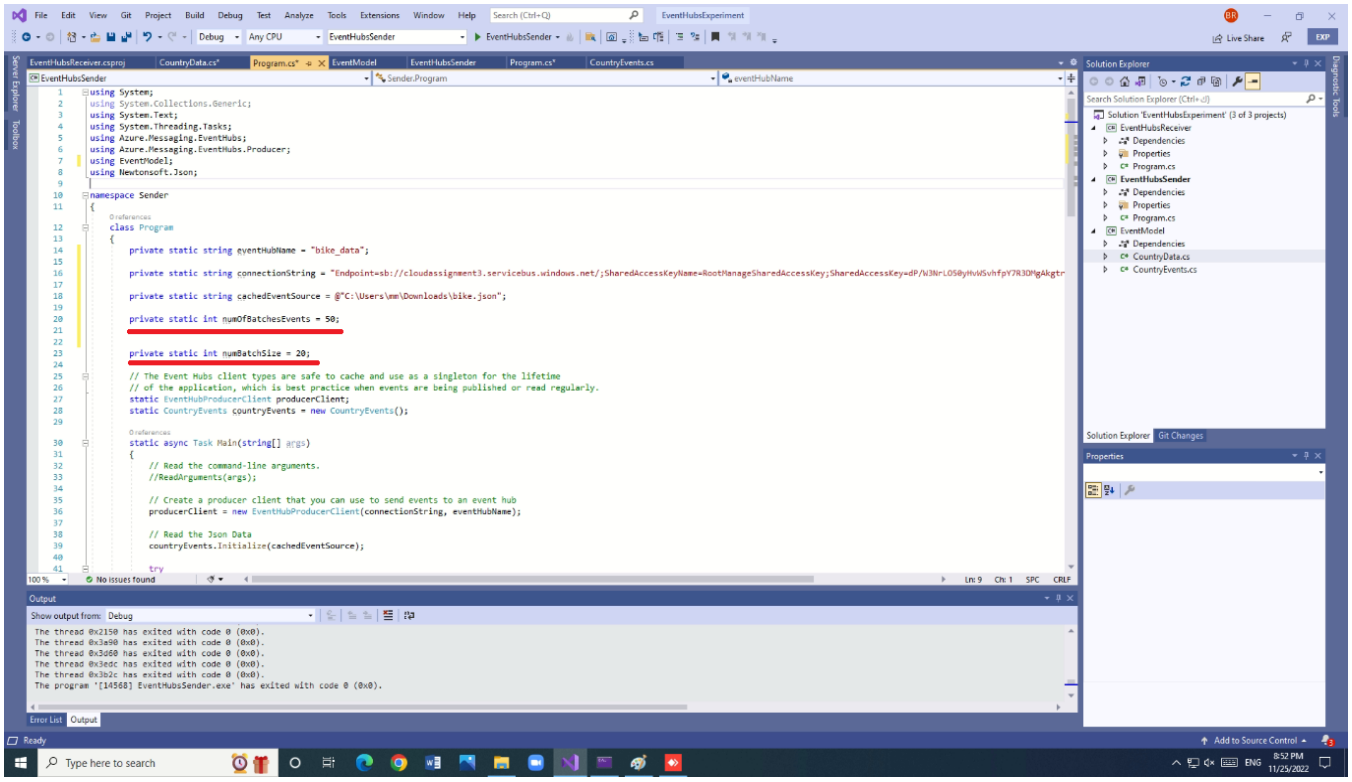


```
using Newtonsoft.Json;
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace EventModel
{
    2 references
    public class CountryEvents
    {
        private int marker = 0;
        private int endMarker = 0;

        private List<CountryData> countryDataCollection = new List<CountryData>();
        4 references
        public string FileName
        {
            get;
            set;
        }

        1 reference
        public void Initialize(string file)
        {
            FileName = file;
            marker = 0;
            countryDataCollection.Clear();

            if (!File.Exists(FileName))
            {
                throw new Exception(String.Format("File {0} does not exist", FileName));
            }

            StreamReader reader = new StreamReader(FileName);
            string line;

            while ((line = reader.ReadLine()) != null){
                var countryData = JsonConvert.DeserializeObject<CountryData>(line);
                countryDataCollection.Add(countryData);
                endMarker += 1;
            }
        }
```

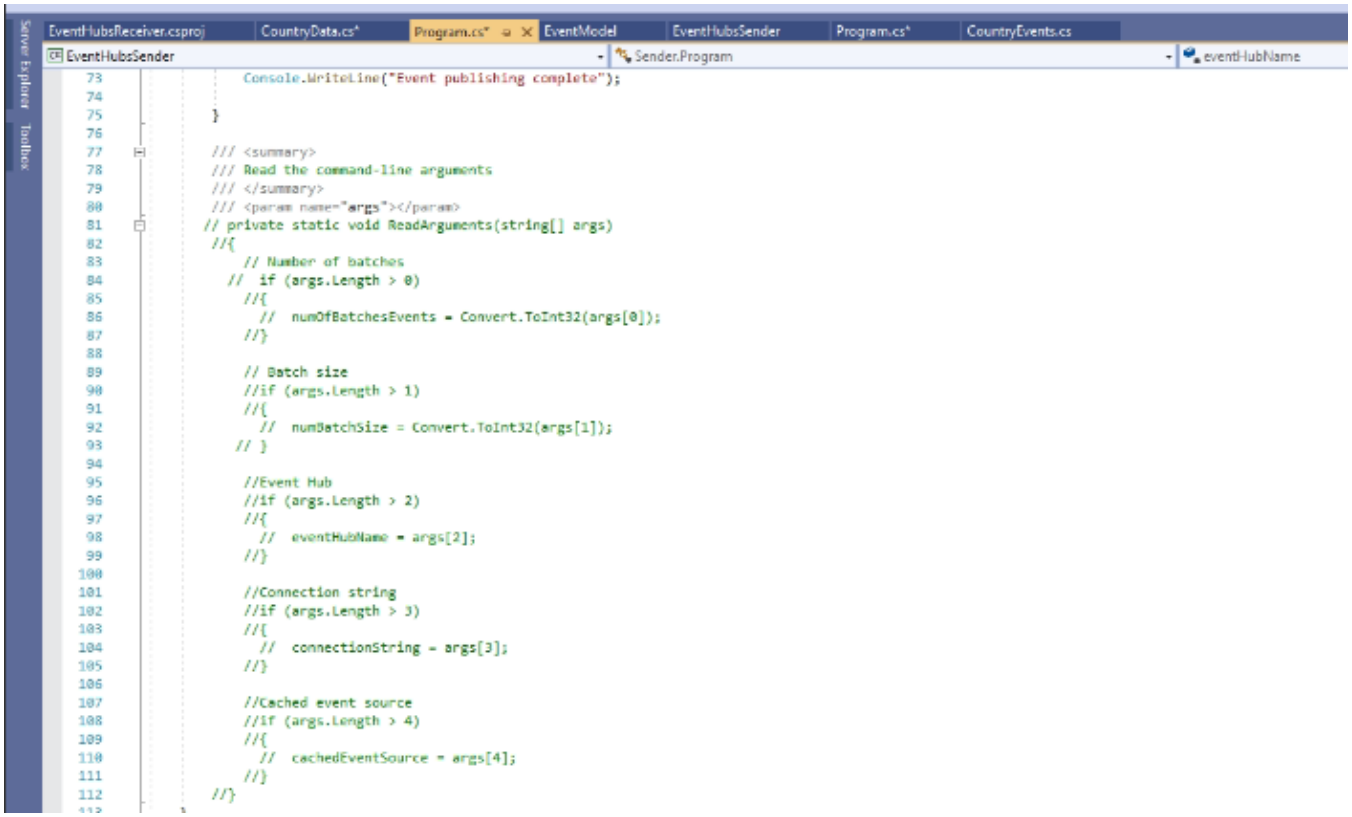
5- Modify the trip entries to 20 and number of events to 50 in the properties of the EventHubSender:



6- Set the number of BatchesEvents to 50, and numBatchSize to 20:



7- Hashing the ReadArguments function in sender and reciever, as we don't have to use it:

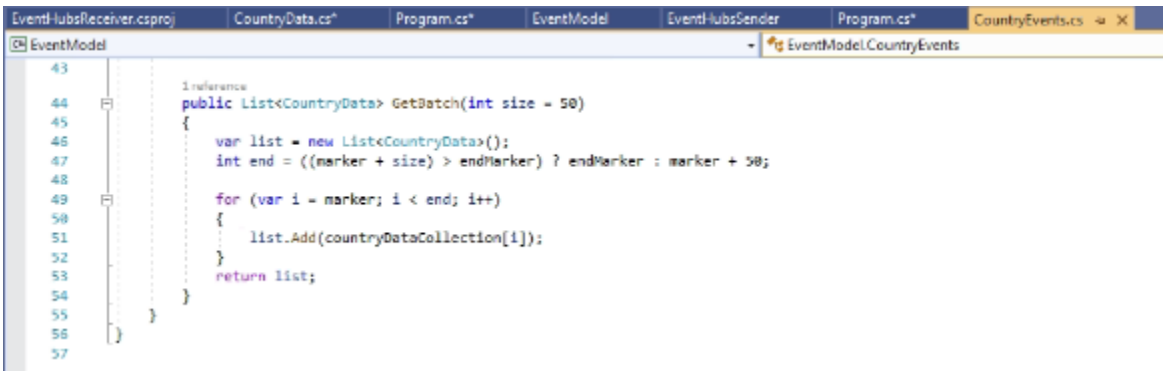


```
            Console.WriteLine("Event publishing complete");

        }

        /// <summary>
        /// Read the command-line arguments
        /// </summary>
        /// <param name="args"></param>
        // private static void ReadArguments(string[] args)
        //{
            // Number of batches
          //  if (args.Length > 0)
            //{
              //  numOfBatchesEvents = Convert.ToInt32(args[0]);
            //}

            // Batch size
            //if (args.Length > 1)
            //{
              //  numBatchSize = Convert.ToInt32(args[1]);
           // }

            //Event Hub
            //if (args.Length > 2)
            //{
              //  eventHubName = args[2];
            //}

            //Connection string
            //if (args.Length > 3)
            //{
              //  connectionString = args[3];
            //}

            //Cached event source
            //if (args.Length > 4)
            //{
              //  cachedEventSource = args[4];
            //}
        //}
```

8- Set the number of seconds to 10:

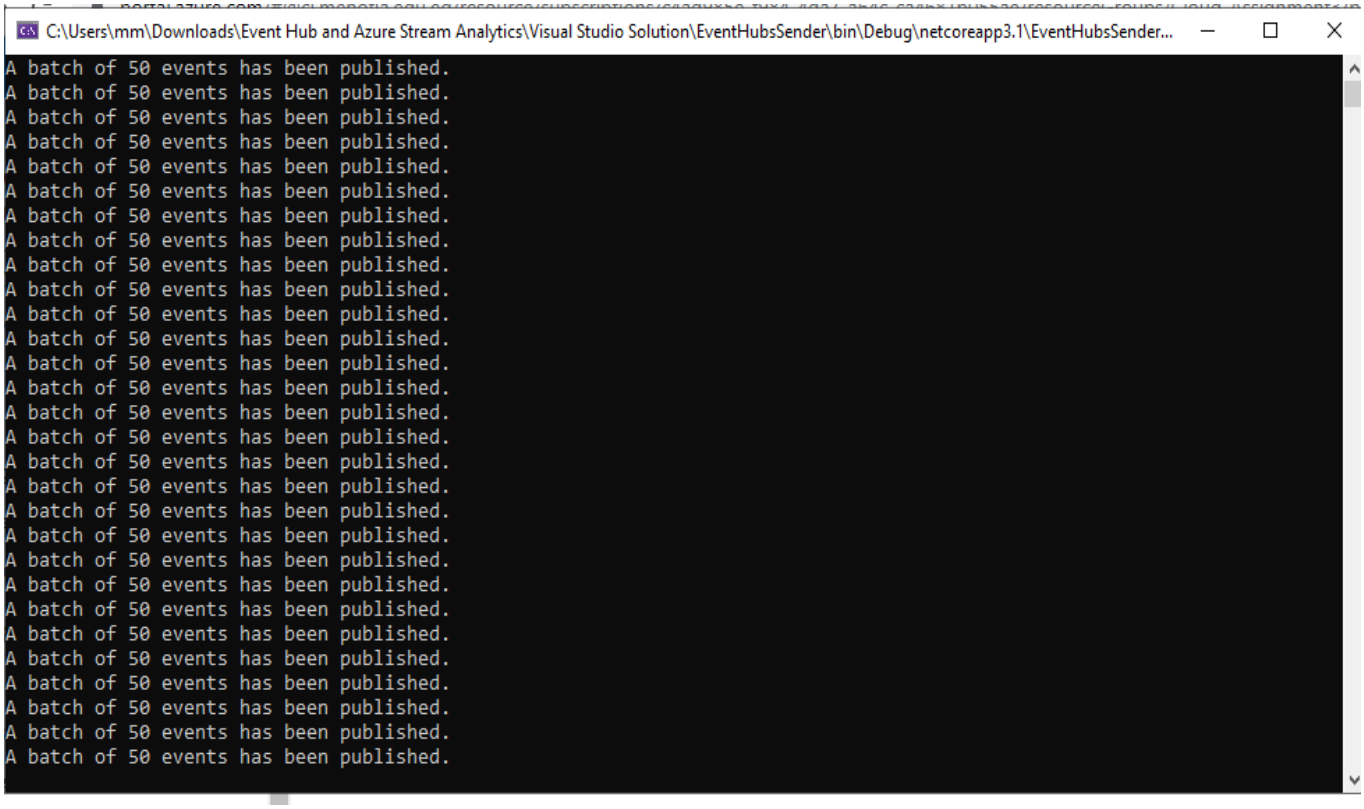
```
    // Add artificial 10 seconds delay to the events
    System.Threading.Thread.Sleep(10 * 1000);

    // Use the producer client to send the batch of events to the event hub
    await producerClient.SendAsync(eventBatch);
    Console.WriteLine("A batch of {0} events has been published.", eventBatch.Count);

}
```

9- At the Event.Cs modify the batch size to be 50:

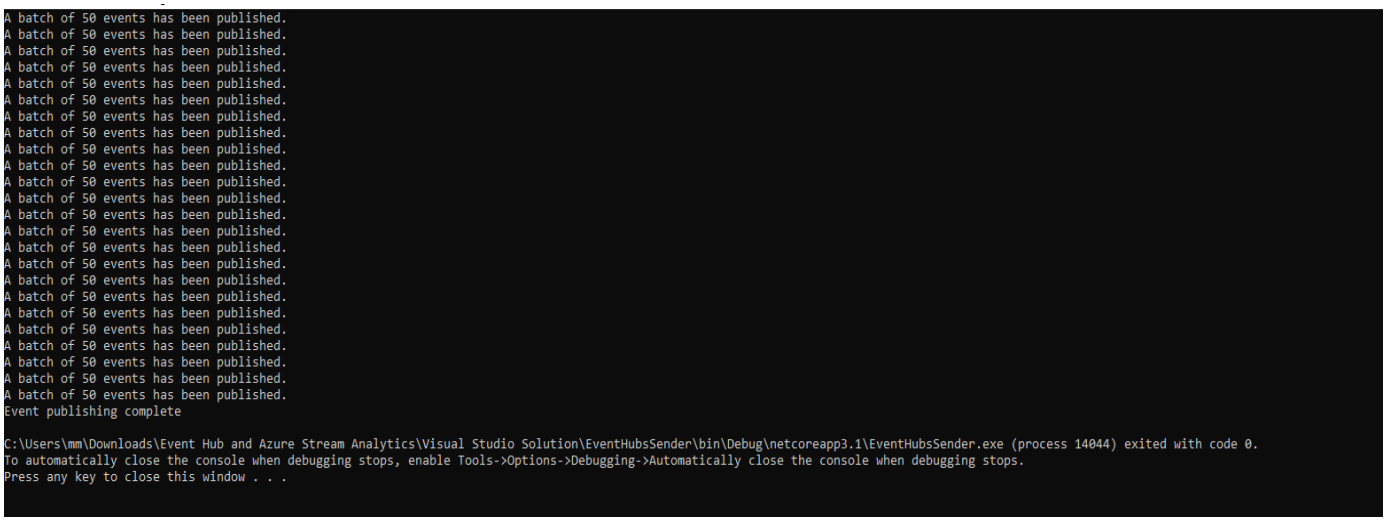


```
        1 reference
        public List<CountryData> GetBatch(int size = 50)
        {
            var list = new List<CountryData>();
            int end = ((marker + size) > endMarker) ? endMarker : marker + 50;

            for (var i = marker; i < end; i++)
            {
                list.Add(countryDataCollection[i]);
            }
            return list;
        }
    }
}
```

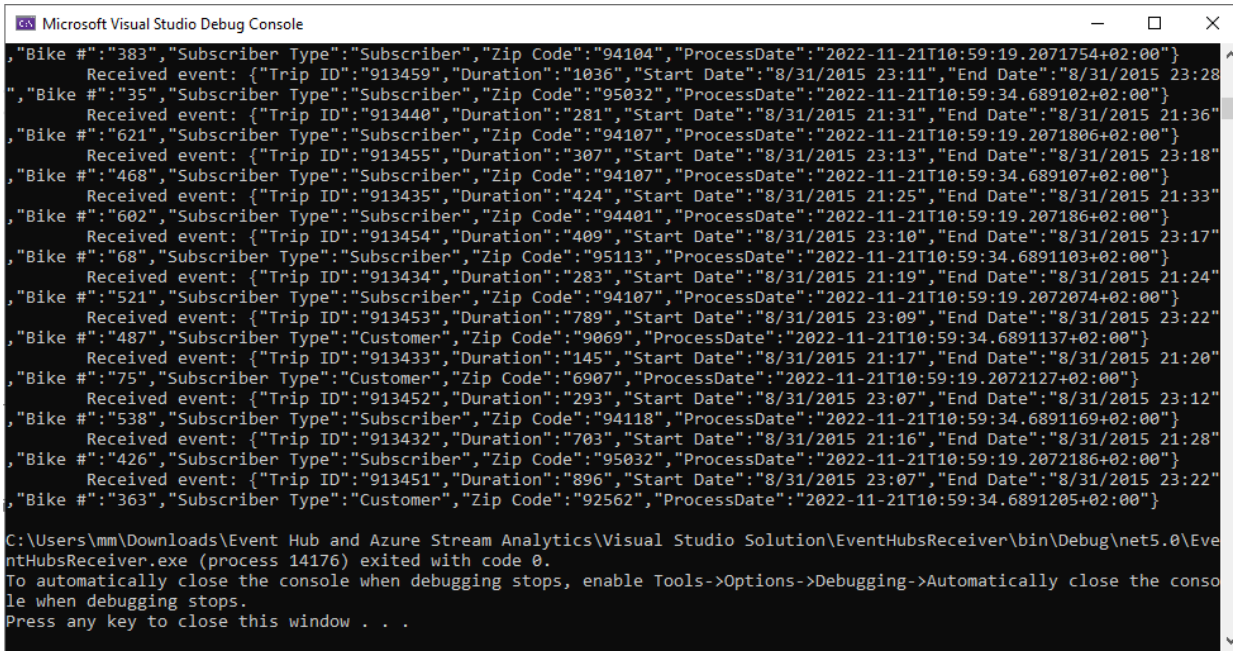
10- Finally, we can send the batches to Azure:



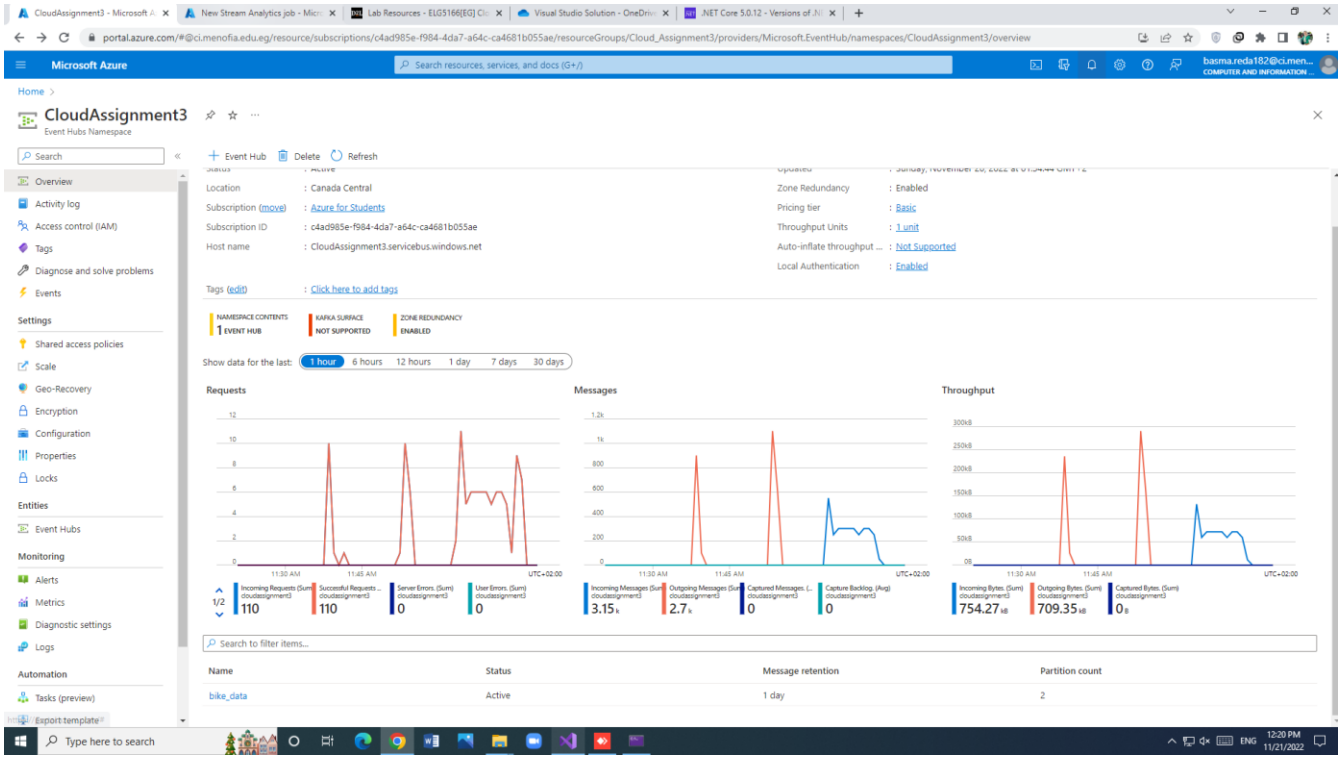
The batches published successfully:

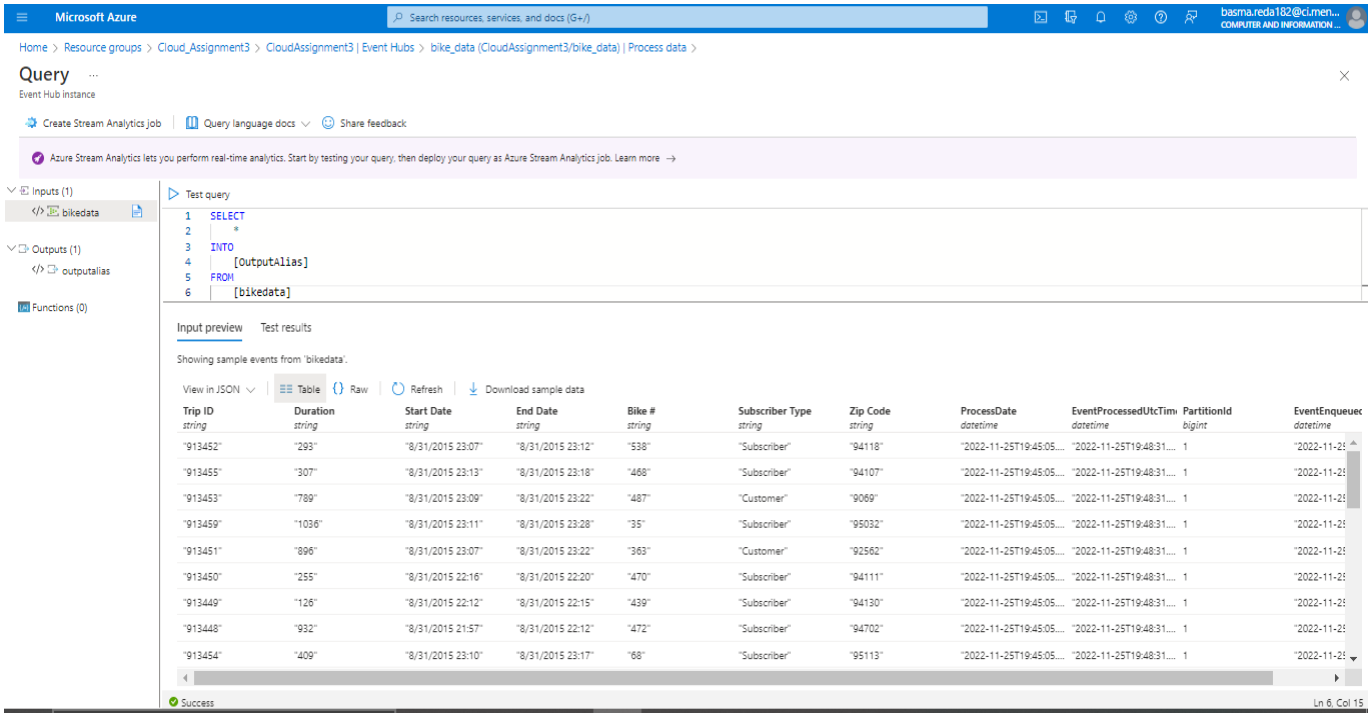


Start the receiver to receive the data:

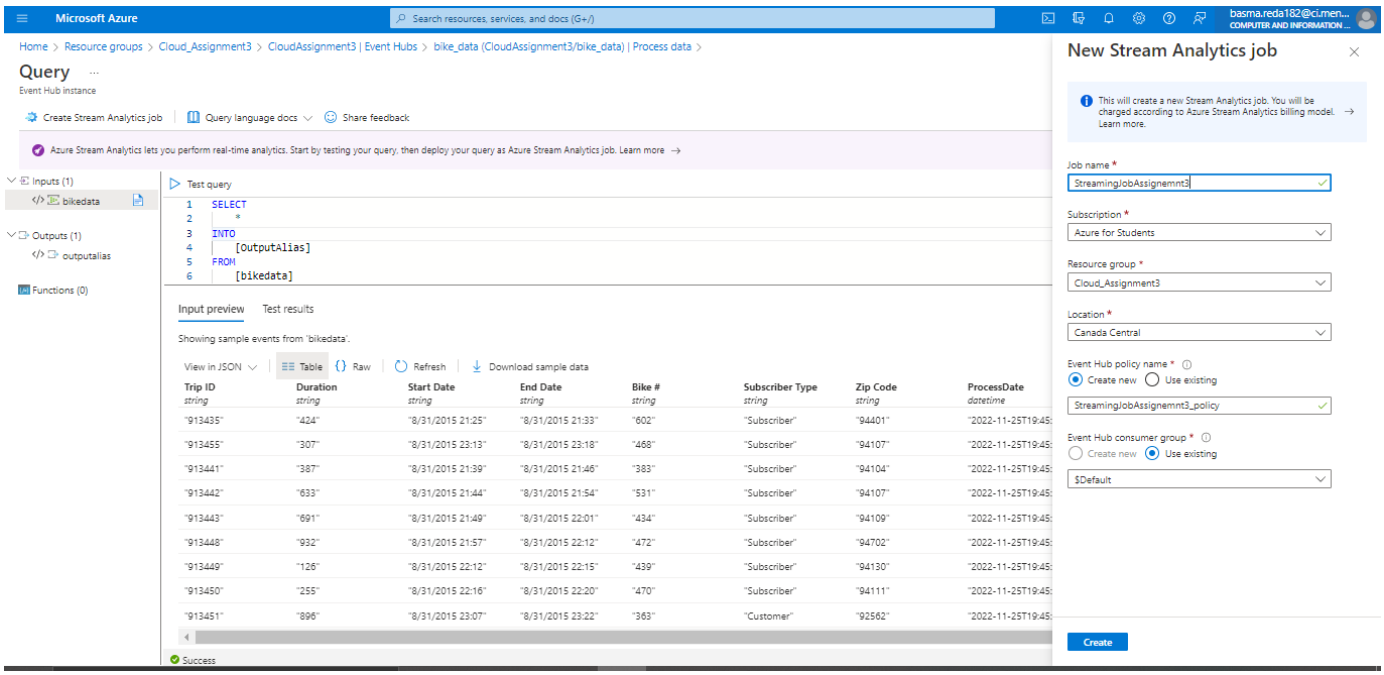


B) Go to your event hub's namespace and show that the entire message set was received. Provide one or more screenshots.

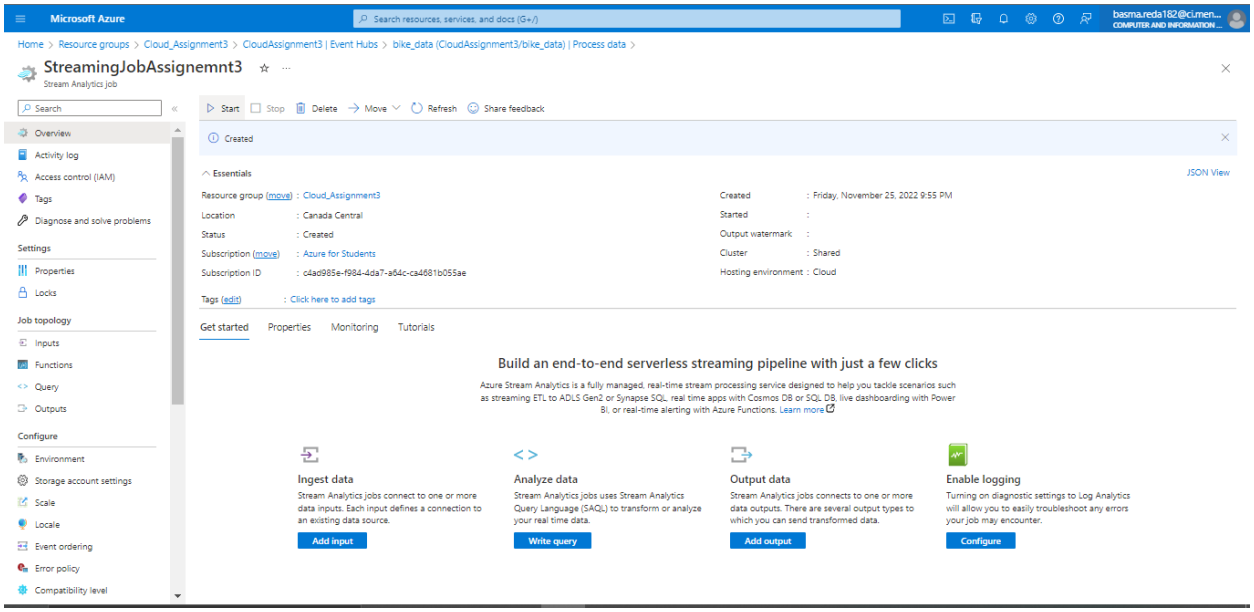


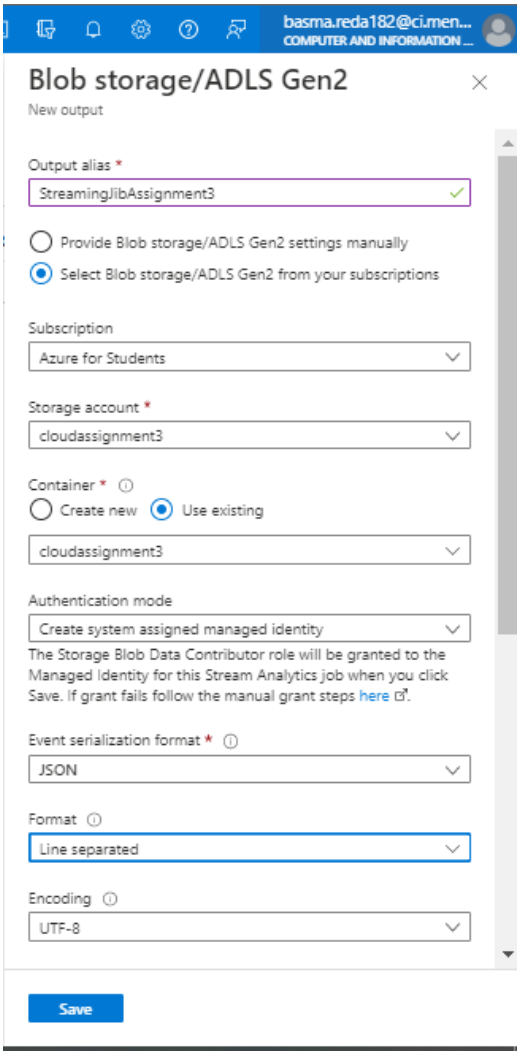


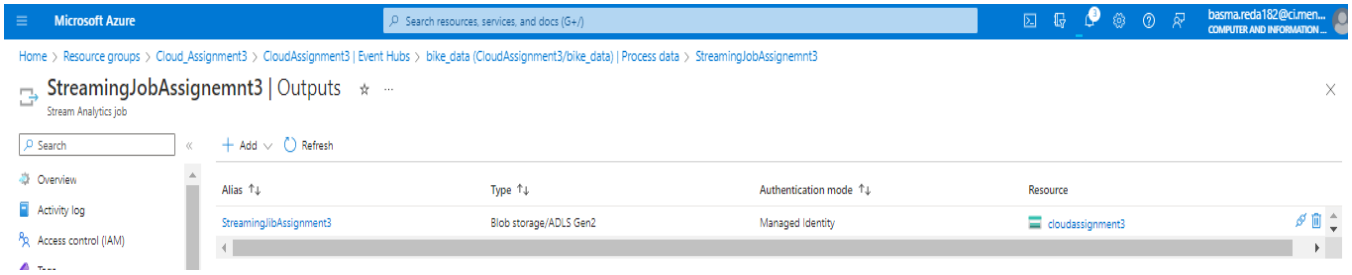
C) *Create an Azure Stream Analytics job that uses either a Storage account to store the summary of the event received. Your summary should include the following fields – WindowEnd, Total Bikes, Total Duration for each Batch received. Download this dataset and include both a screenshot and the data as part of your submission.*



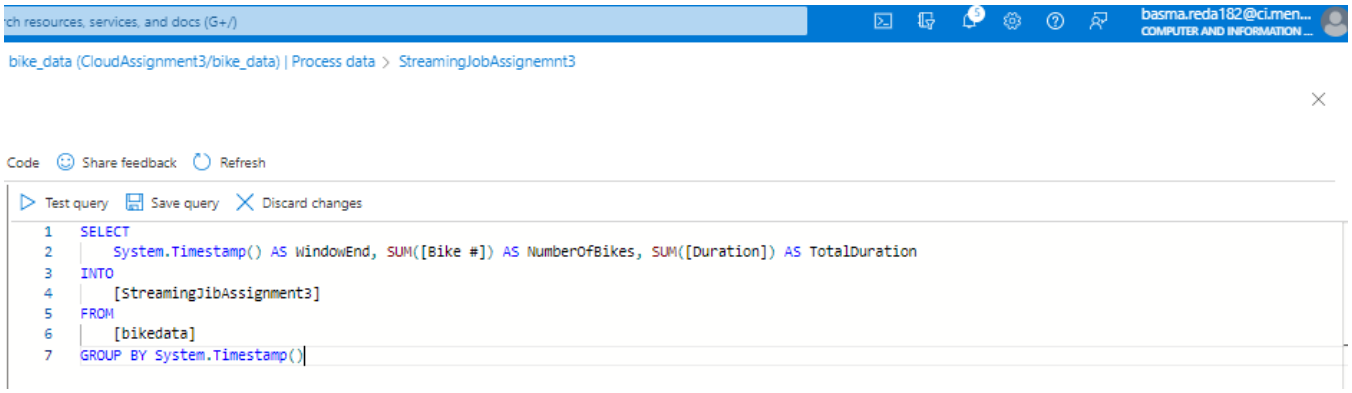
To add an output:



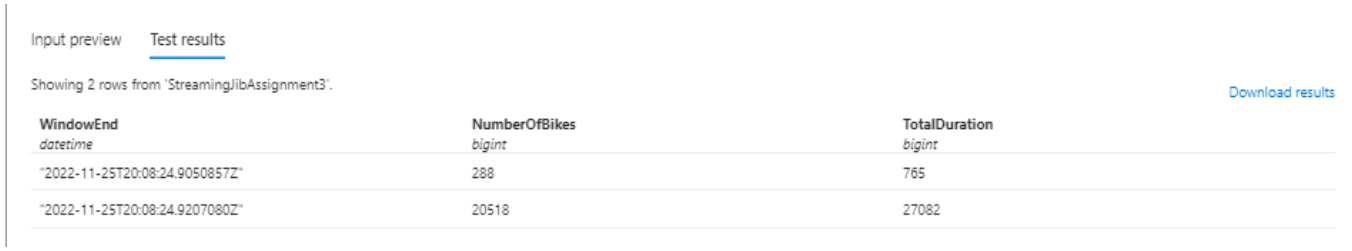




The query is:



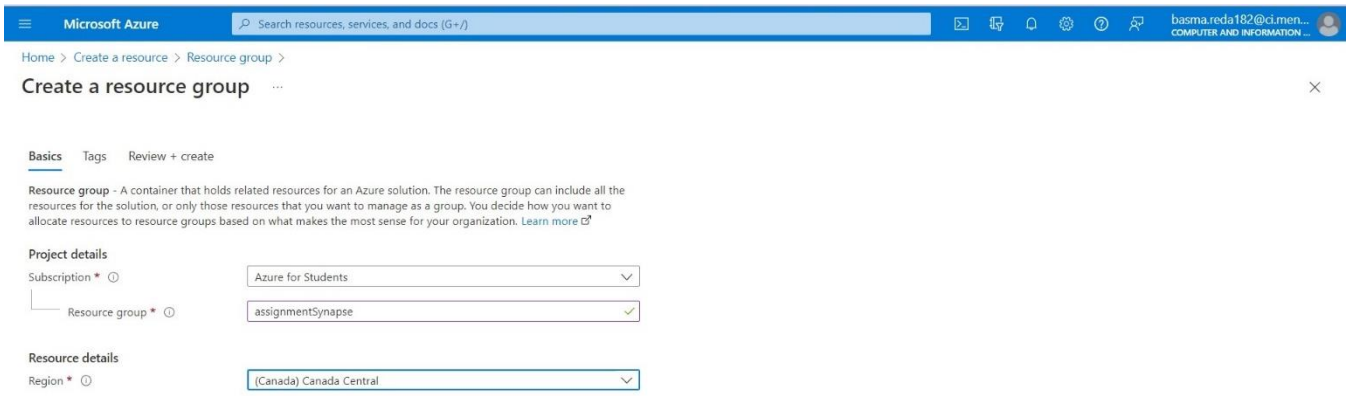
The output:

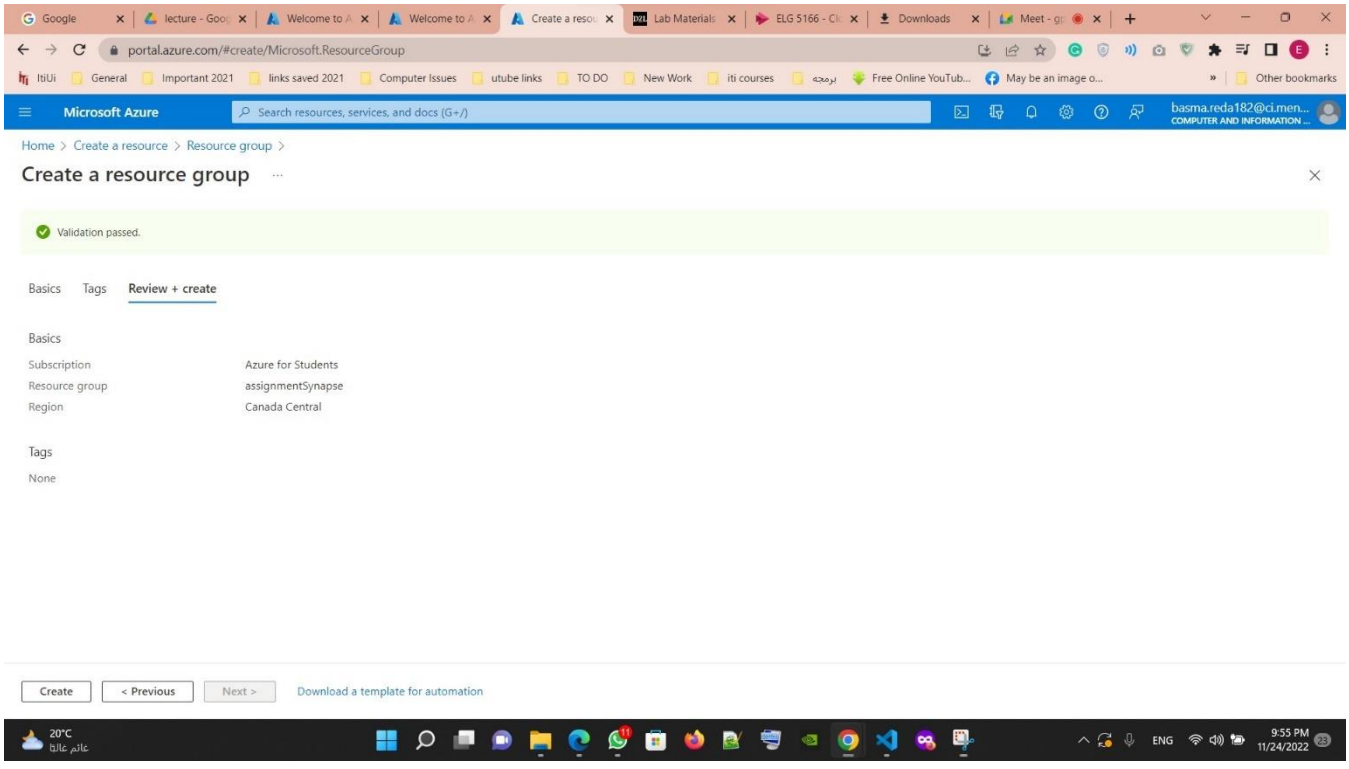


Save the query and download the results.

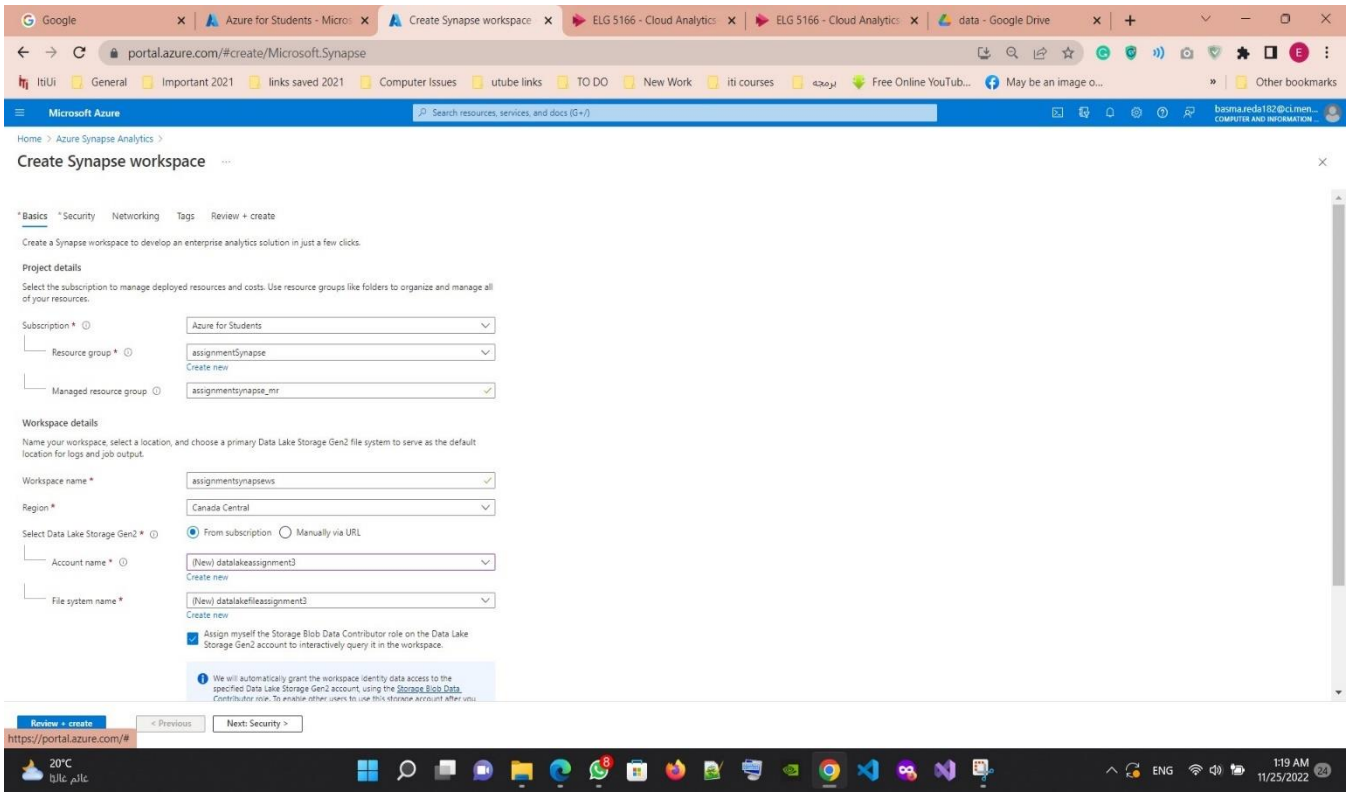
Part 2: Azure Synapse Analytics:

1- Create a resource group:

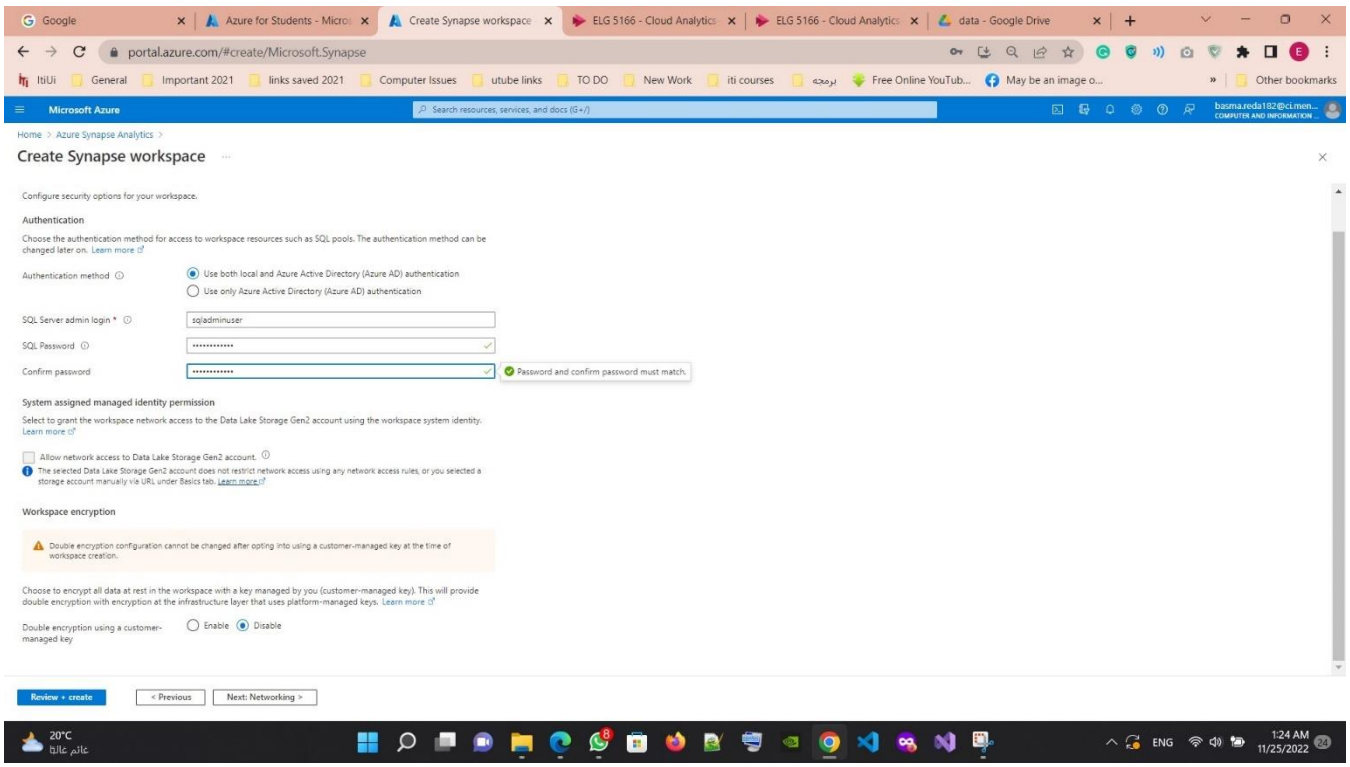


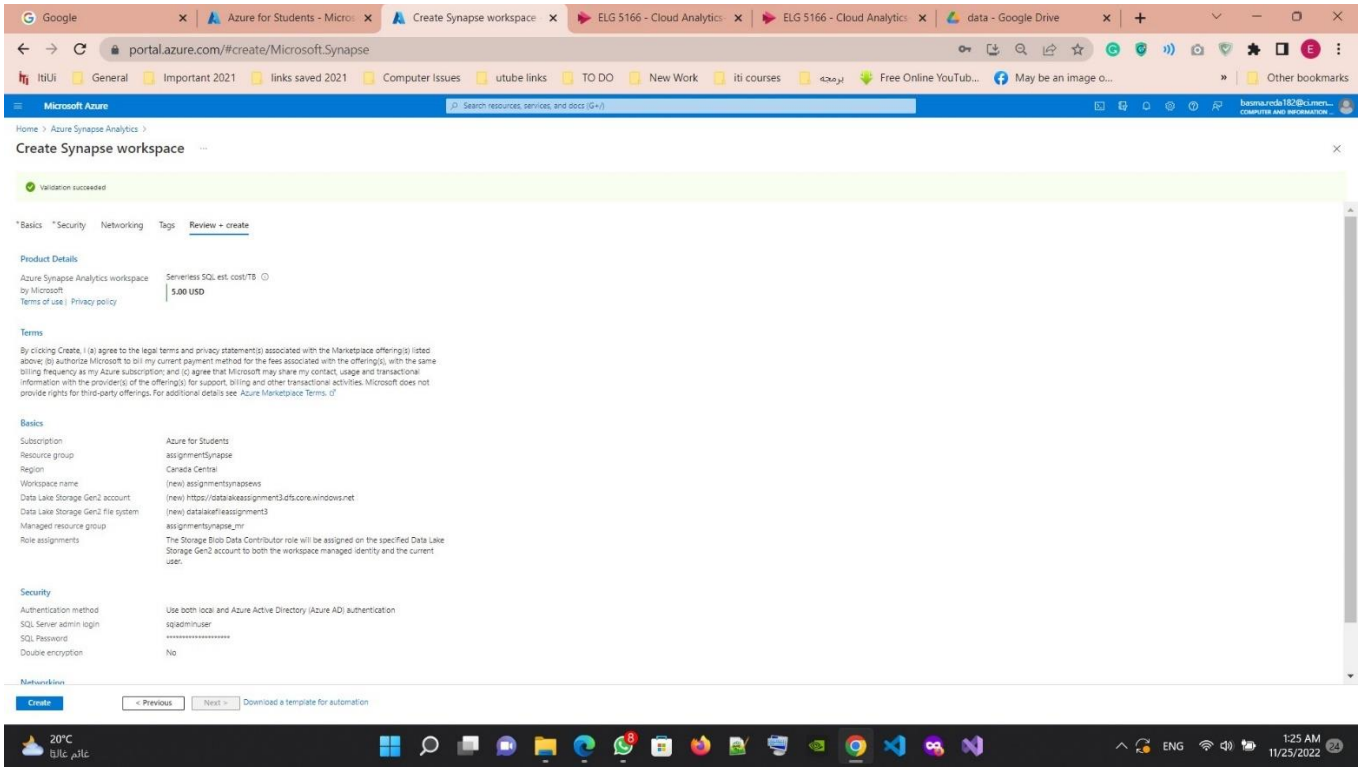


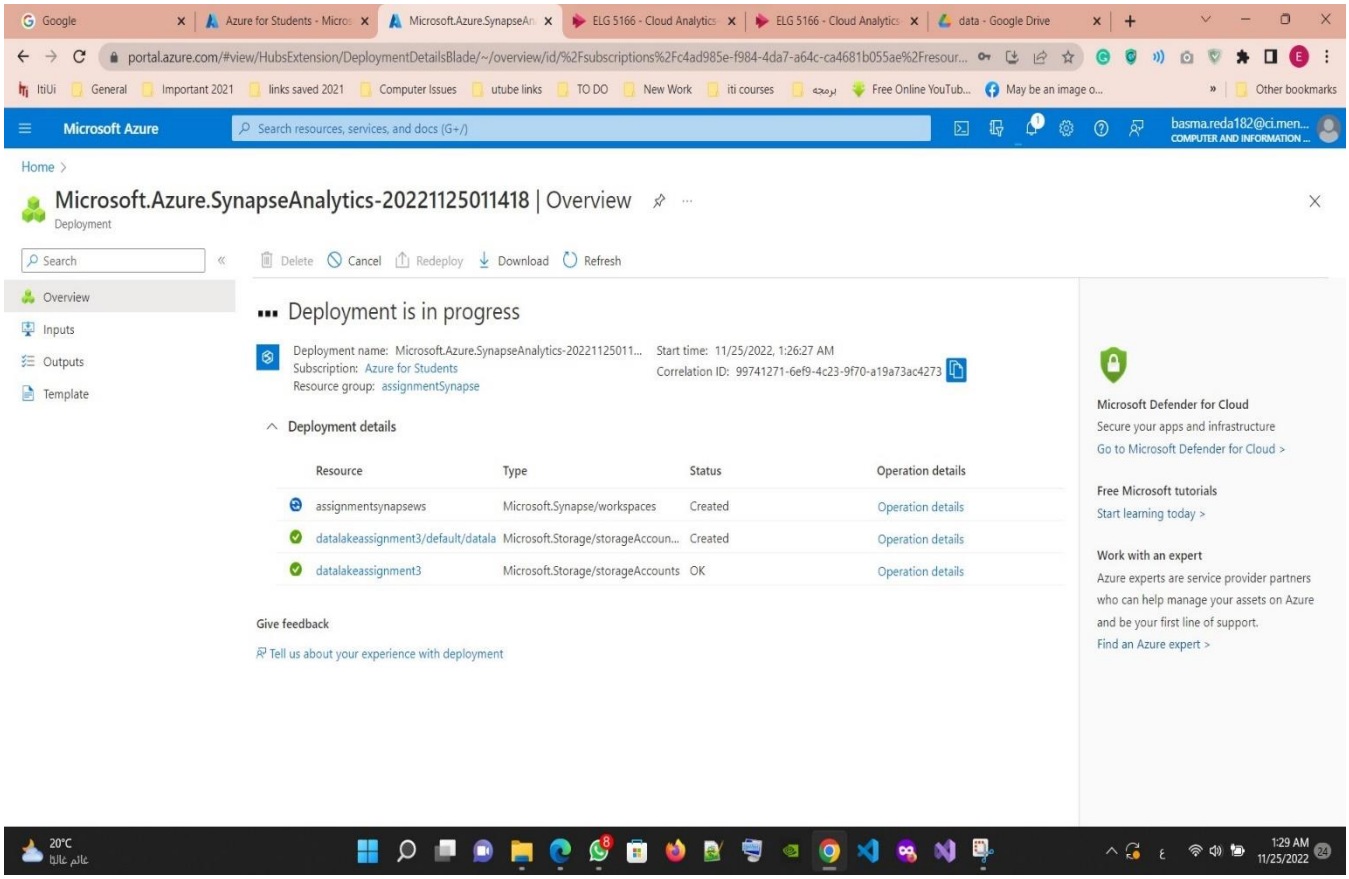
2- Create a Synapse Workspace:

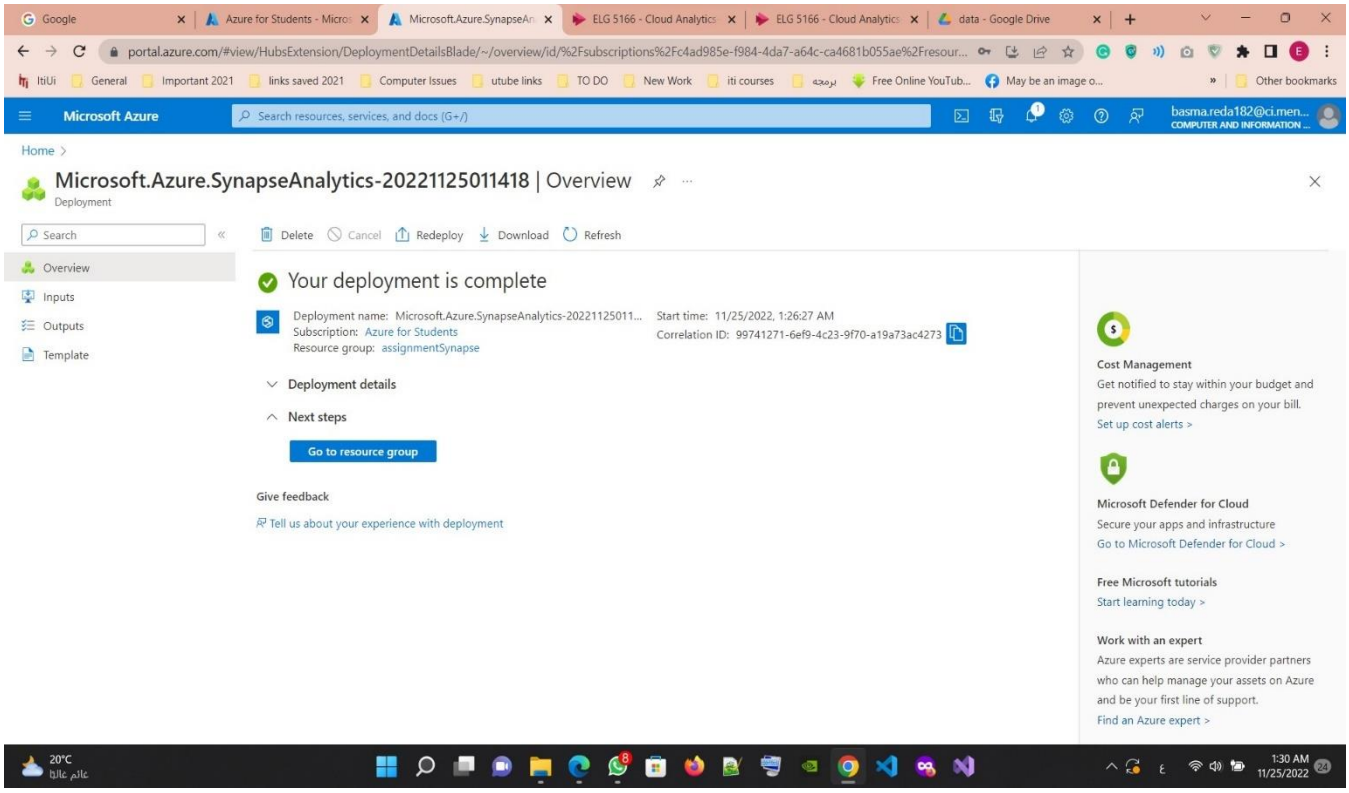


3- Review the security of the workspace:

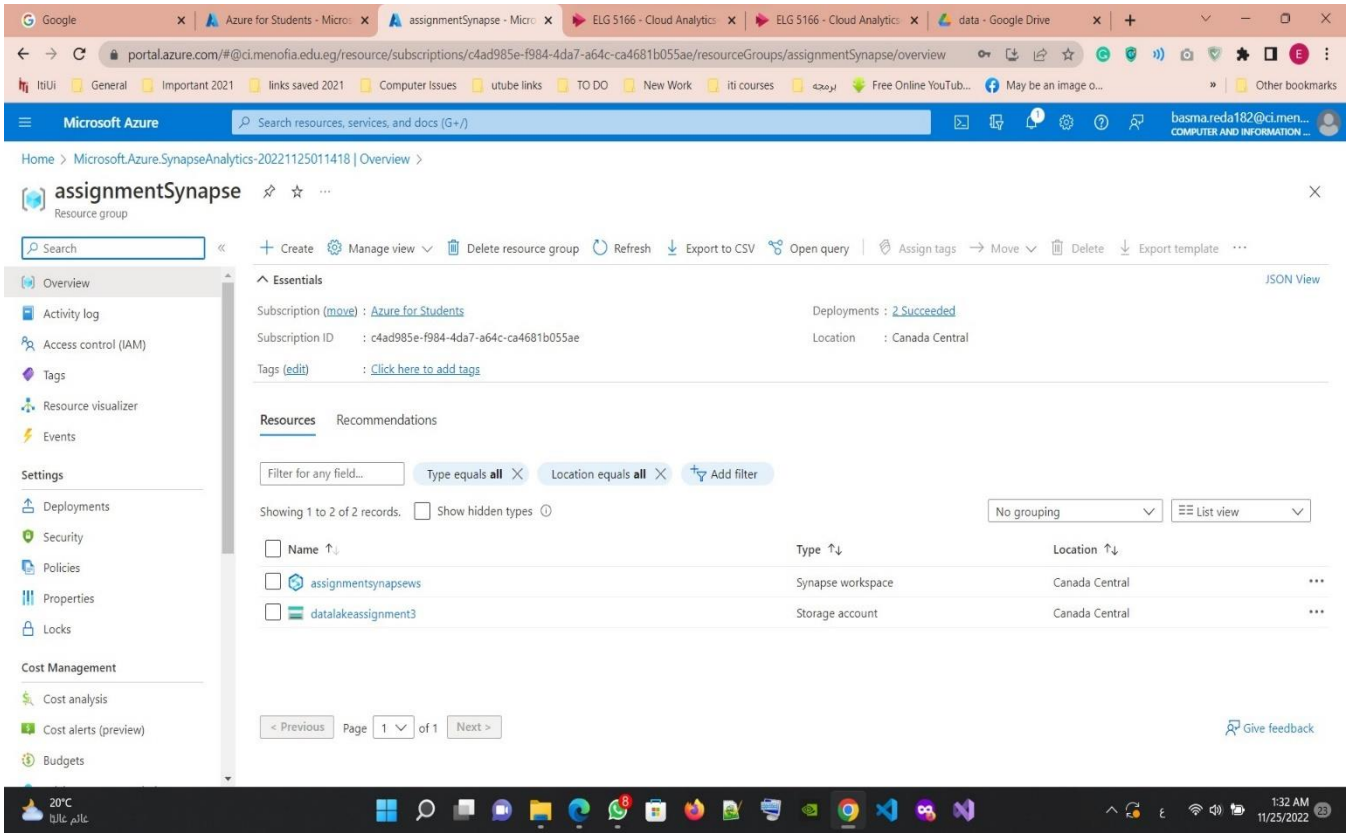


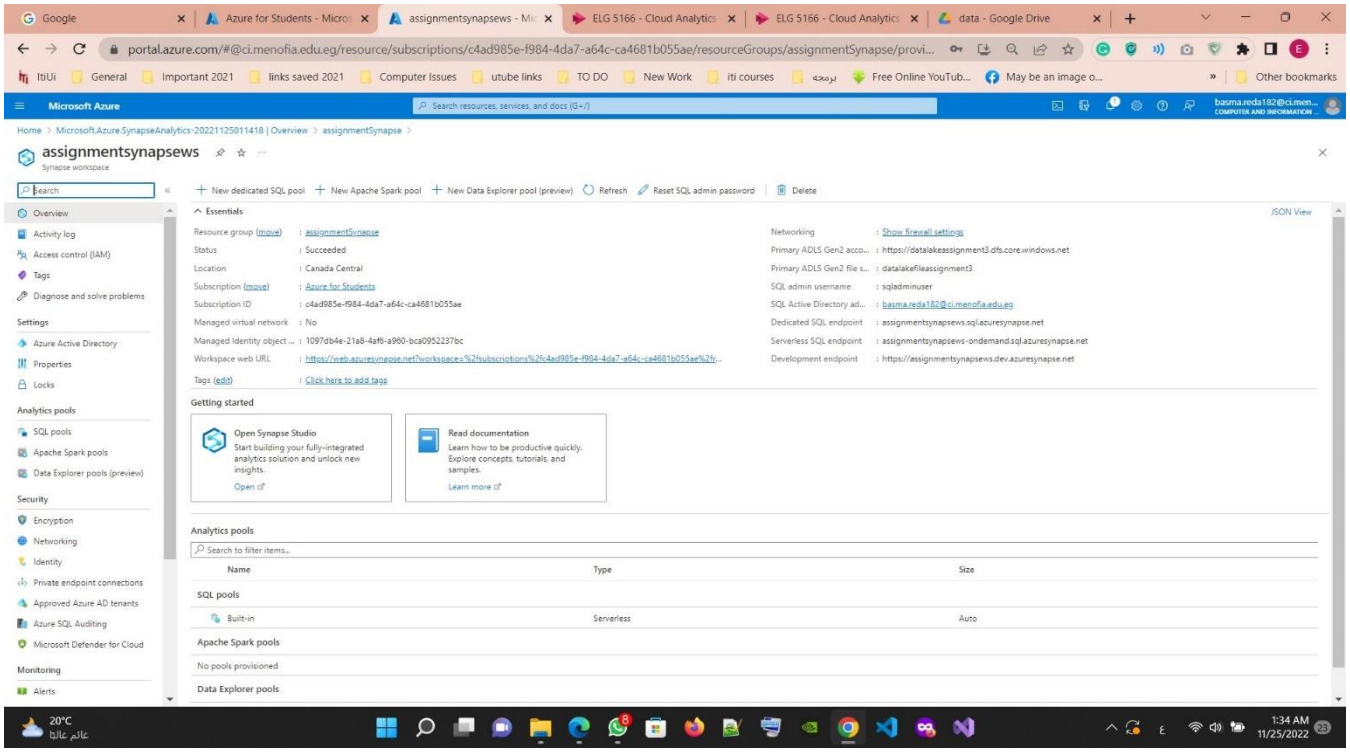

Create Synapse workspace
Validation succeeded
Basics Security Networking Tags Review + create
Product Details
Azure Synapse Analytics workspace by Microsoft
Terms of use | Privacy policy
Serverless SQL est. cost/TB
5.00 USD
Terms
By clicking Create, I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. For additional details see Azure Marketplace Terms.
Basics
Subscription Azure for Students
Resource group assignmentSynapse
Region Canada Central
Workspace name (new) assignmentsynapsews
Data Lake Storage Gen2 account (new) https://datalakeassignment3.dfs.core.windows.net
Data Lake Storage Gen2 file system (new) datalakefileassignment3
Managed resource group assignmentsynapse_mr
Role assignments The Storage Blob Data Contributor role will be assigned on the specified Data Lake Storage Gen2 account to both the workspace managed identity and the current user.
Security
Authentication method Use both local and Azure Active Directory (Azure AD) authentication
SQL Server admin login sqladminuser
SQL Password
Double encryption No
Networking
Create
< Previous
Next >
Download a template for automation


Microsoft Azure
Search resources, services, and docs (G+/)
Home
Microsoft.Azure.SynapseAnalytics-20221125011418 | Overview
Deployment
Search
Overview
Inputs
Outputs
Template
Delete
Cancel
Redeploy
Download
Refresh
Deployment is in progress
Deployment name: Microsoft.Azure.SynapseAnalytics-20221125011...
Subscription: Azure for Students
Resource group: assignmentSynapse
Start time: 11/25/2022, 1:26:27 AM
Correlation ID: 99741271-6ef9-4c23-9f70-a19a73ac4273
Deployment details
Resource Type Status Operation details
assignmentsynapsews Microsoft.Synapse/workspaces Created Operation details
datalakeassignment3/default/datala Microsoft.Storage/storageAccoun... Created Operation details
datalakeassignment3 Microsoft.Storage/storageAccounts OK Operation details
Give feedback
Tell us about your experience with deployment
Microsoft Defender for Cloud
Secure your apps and infrastructure
Go to Microsoft Defender for Cloud >
Free Microsoft tutorials
Start learning today >
Work with an expert
Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support.
Find an Azure expert >

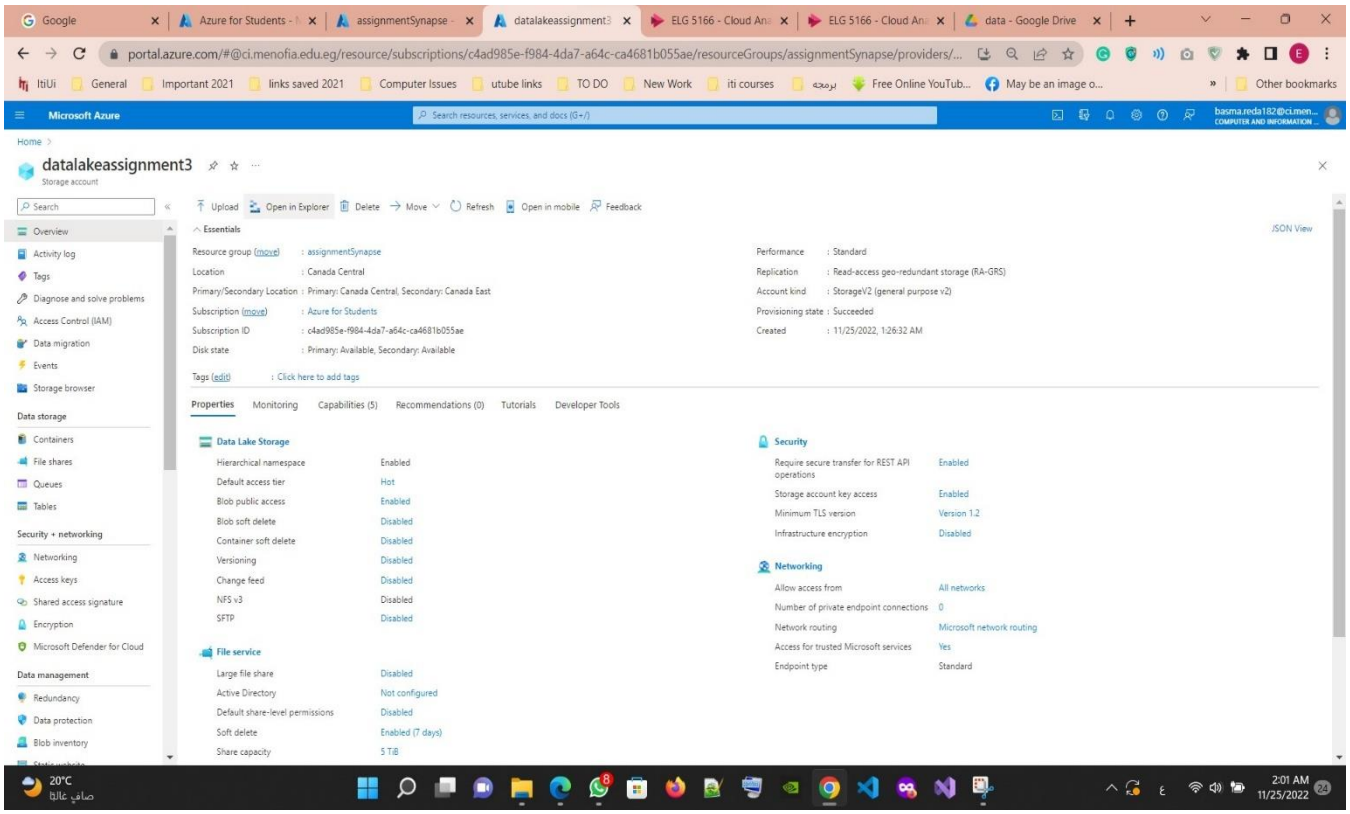


4- Go to the Synapse that has been created:

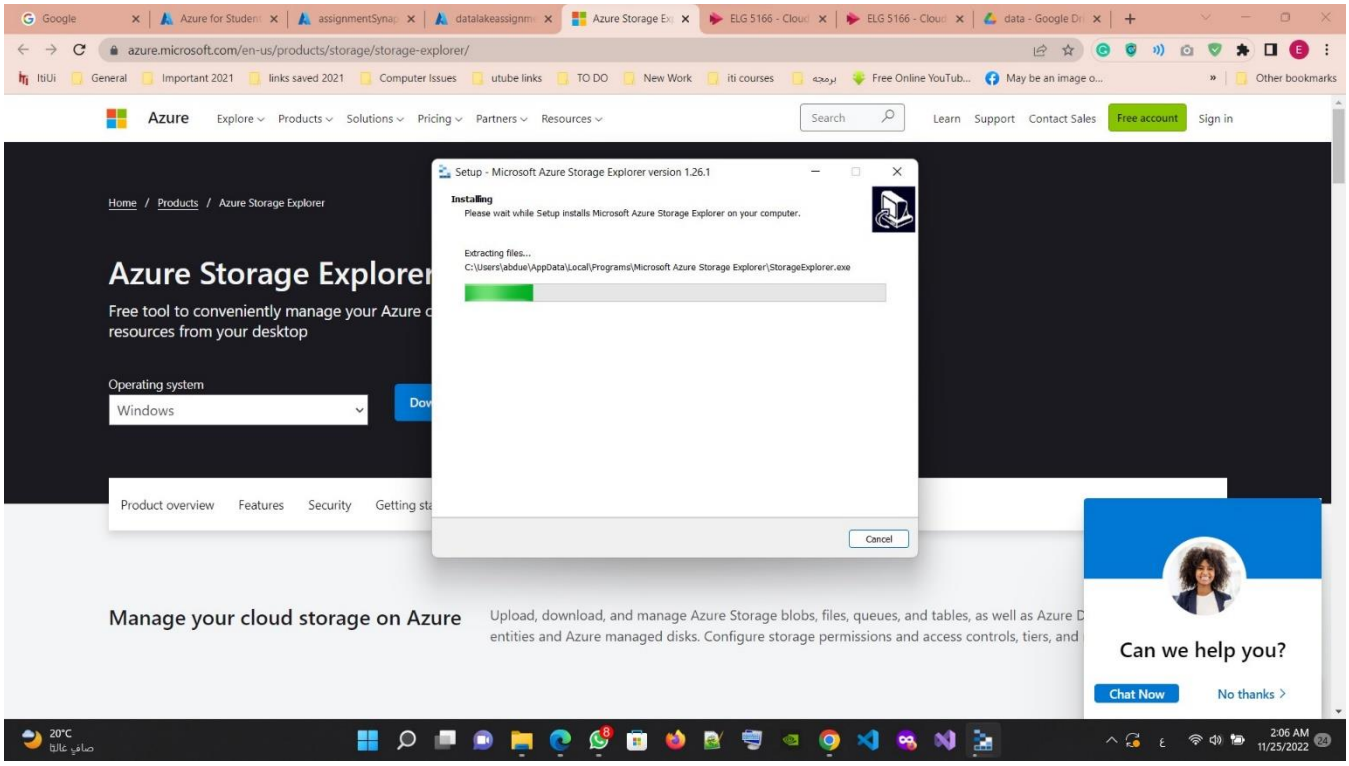




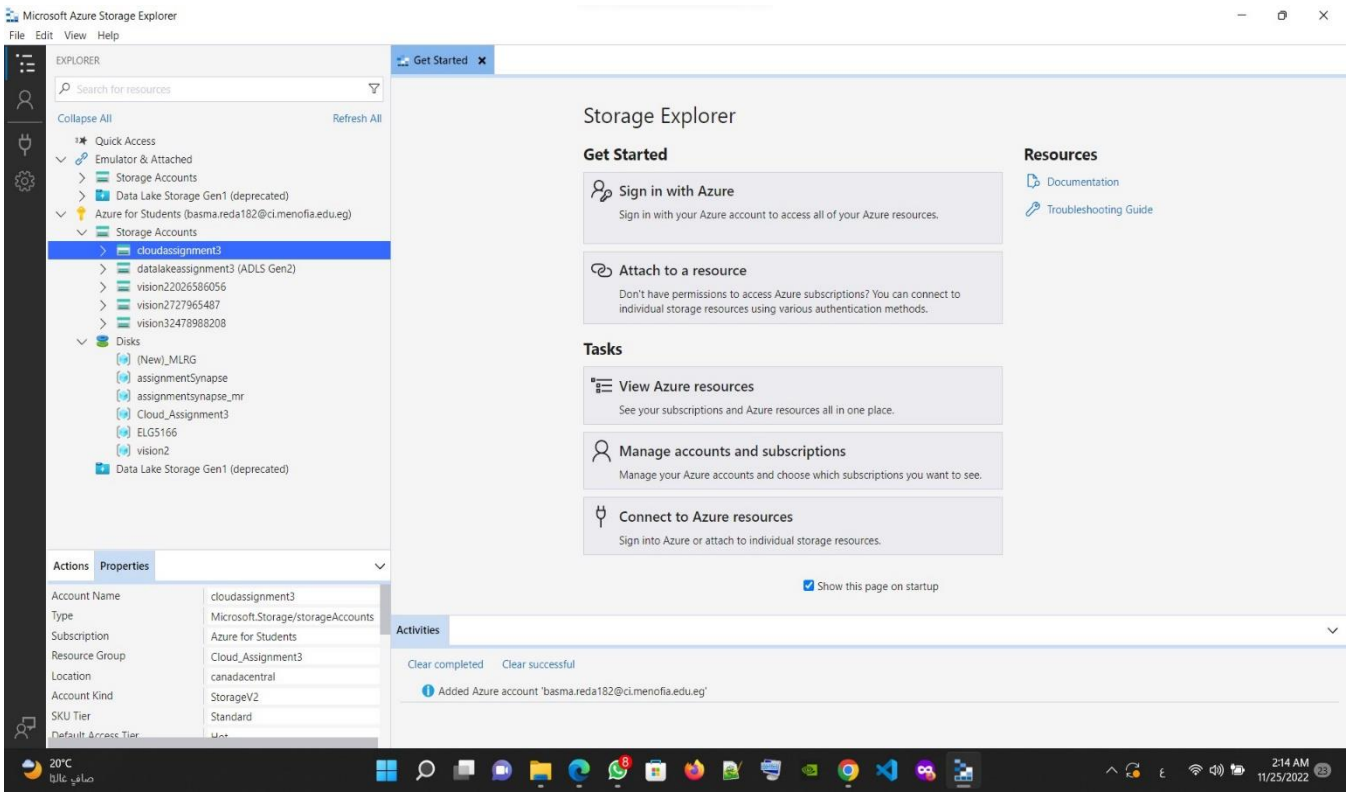
5- Create a storage account to store the data:



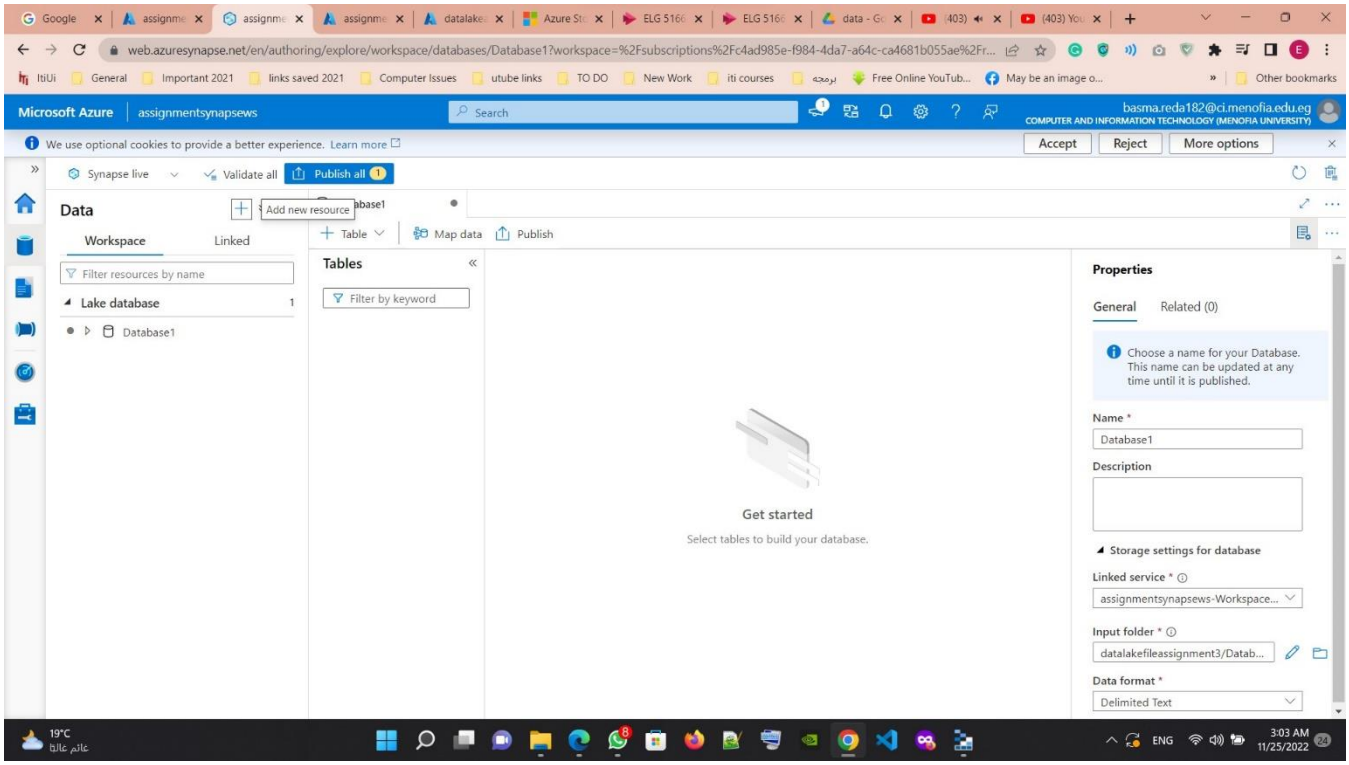
6- Install and download the MSI file:



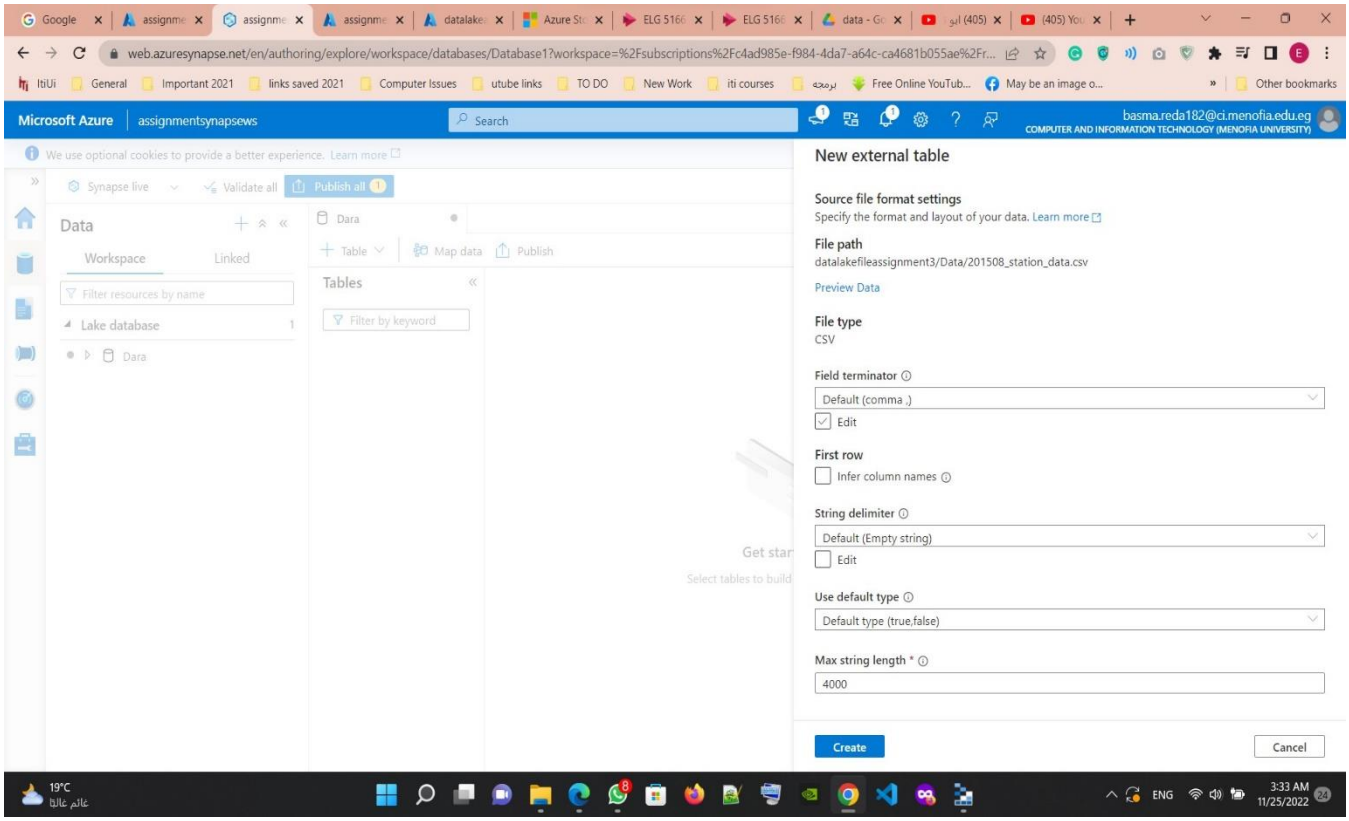
7- Get started with the Synapse:



8- Open the Synapse Workspace:

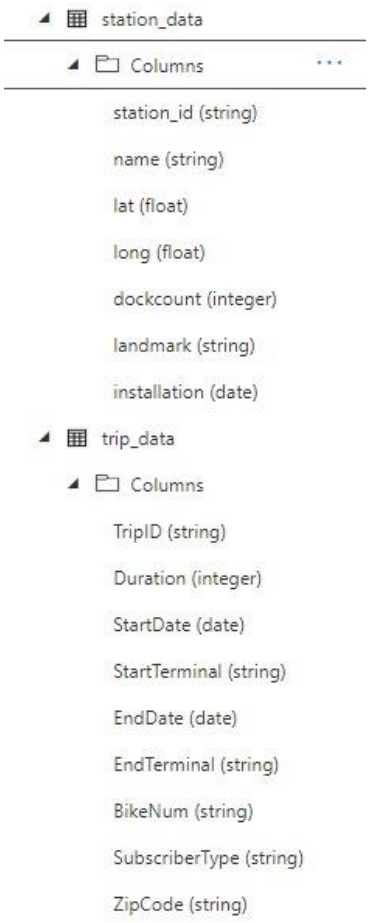


9- Create a new external table:



A) Top 20 zip codes for bike up.

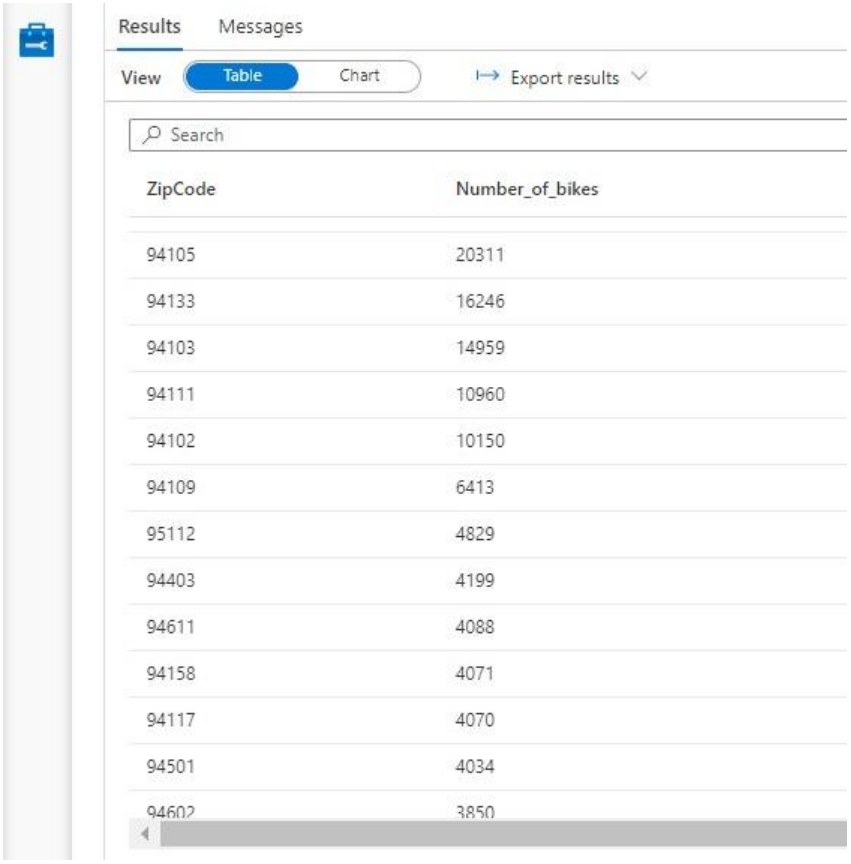
Set the right format for the data:



Query:

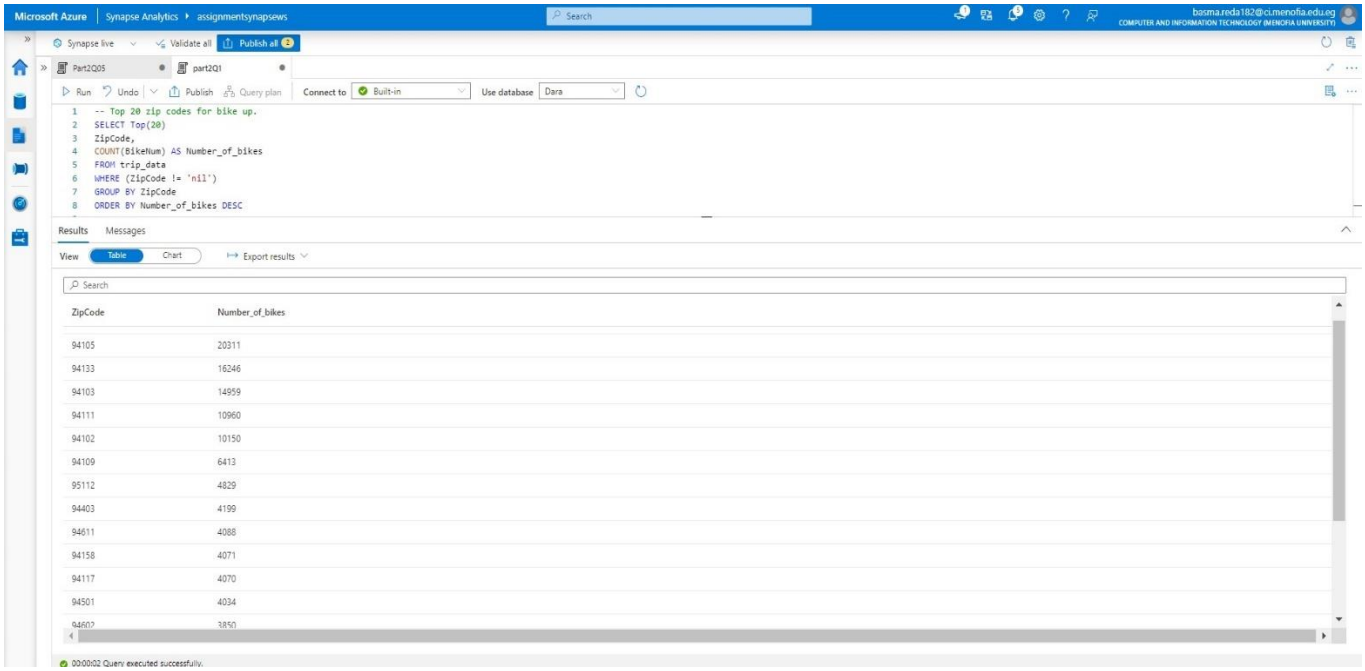
```
-- Top 20 zip codes for bike up.
SELECT Top(20)
ZipCode,
COUNT(BikeNum) AS Number_of_bikes
FROM trip_data
WHERE (ZipCode != 'nil')
GROUP BY ZipCode
ORDER BY Number_of_bikes DESC
```

Output:



ZipCode	Number_of_bikes
94105	20311
94133	16246
94103	14959
94111	10960
94102	10150
94109	6413
95112	4829
94403	4199
94611	4088
94158	4071
94117	4070
94501	4034
94602	3850

Query and the output in Azure Synapse:



Save the results in a CSV file:

B) Monthly duration aggregate across the rental subscriber types, ordered in descending order of the busiest months (use a meaningful measure for the aggregate)

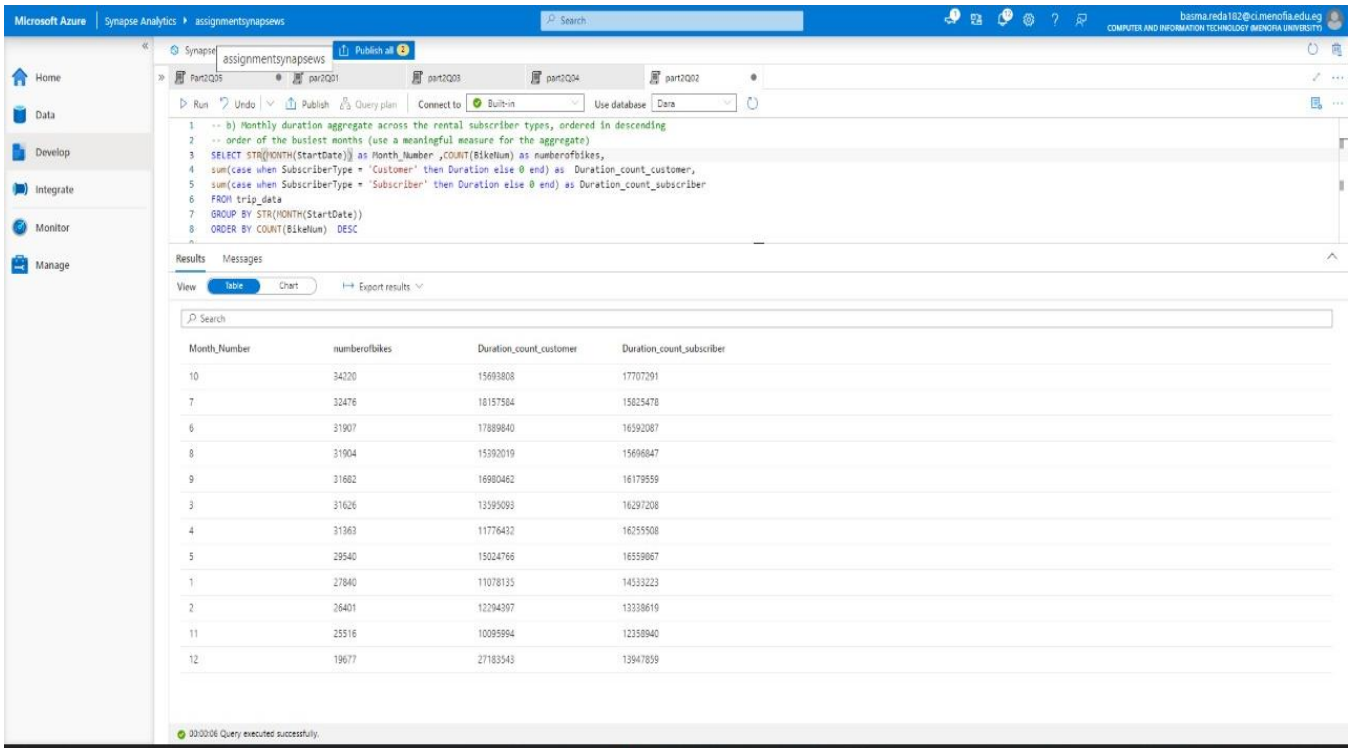
Query:

```
-- b) Monthly duration aggregate across the rental subscriber types, ordered in descending
-- order of the busiest months (use a meaningful measure for the aggregate)
SELECT STR(MONTH(StartDate)) as Month_Number ,COUNT(BikeNum) as numberofbikes,
sum(case when SubscriberType = 'Customer' then Duration else 0 end) as  Duration_count_customer,
sum(case when SubscriberType = 'Subscriber' then Duration else 0 end) as Duration_count_subscriber

FROM trip_data
GROUP BY STR(MONTH(StartDate))
ORDER BY COUNT(BikeNum)  DESC
```

Output:

Month_Number	numberofbikes	Duration_count_customer	Duration_count_subscriber
10	34220	15693808	17707291
7	32476	18157584	15825478
6	31907	17889840	16592087
8	31904	15392019	15696847
9	31682	16980462	16179559
3	31626	13595093	16297208
4	31363	11776432	16255508
5	29540	15024766	16559067
1	27840	11078135	14533223
2	26401	12294397	13338619
11	25516	10095994	12358940
12	19677	27183543	13947859

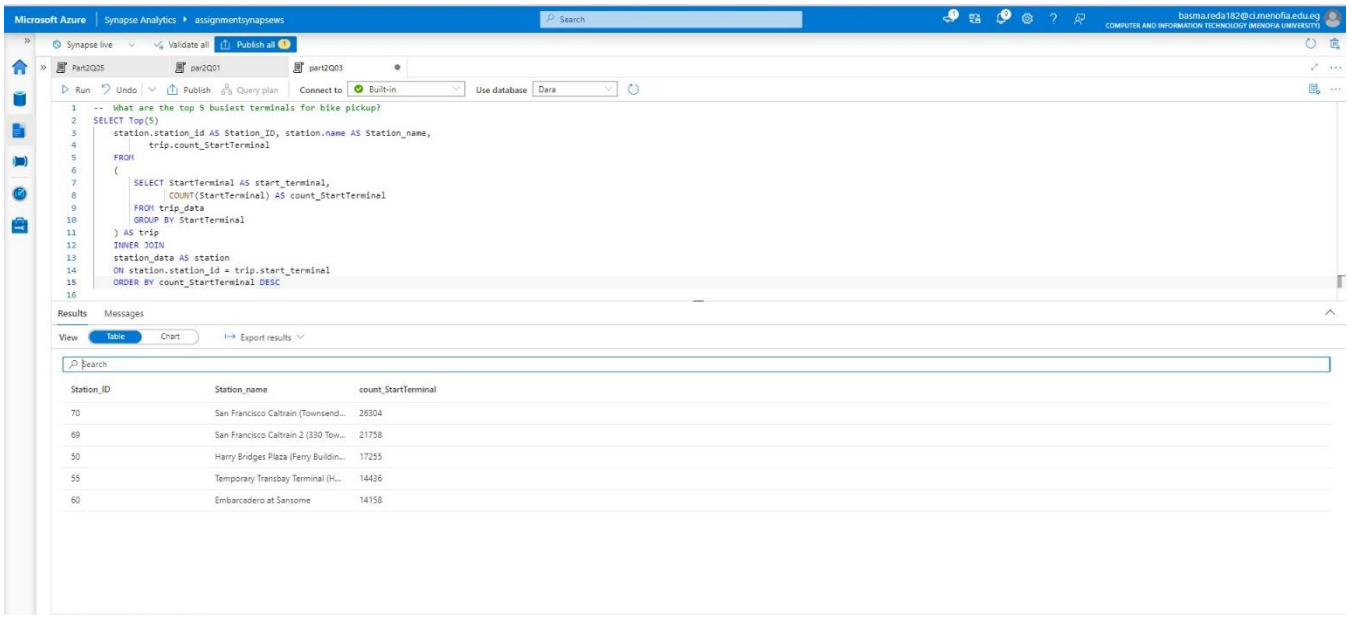
Screenshot from Azure:



Save the output in a CSV file.

C) What are the top 5 busiest terminals for bike pickup?

Screenshot from Azure:



Query:

```
--  What are the top 5 busiest terminals for bike pickup?
SELECT Top(5)
    station.station_id AS Station_ID, station.name AS Station_name,
           trip.count_StartTerminal
    FROM
    (
        SELECT StartTerminal AS start_terminal,
               COUNT(StartTerminal) AS count_StartTerminal
        FROM trip_data
        GROUP BY StartTerminal
    ) AS trip
    INNER JOIN
    station_data AS station
    ON station.station_id = trip.start_terminal
    ORDER BY count_StartTerminal DESC
```

Output:

Results Messages

View Table Chart Export results

Search

Station_ID	Station_name	count_StartTerminal
70	San Francisco Caltrain (Townsend...	26304
69	San Francisco Caltrain 2 (330 Tow...	21758
50	Harry Bridges Plaza (Ferry Buildin...	17255
55	Temporary Transbay Terminal (H...	14436
60	Embarcadero at Sansome	14158

Save the results is a CSV file.

D) Which 5 terminals has the least drop-offs?

Query:

```
/*d) Which 5 terminal has the least drop-offs?*/
SELECT Top(5) station.station_id AS Station_ID, station.name AS Station_name,
    trip.drop_offs_Count
FROM
(
    SELECT EndTerminal AS end_terminal,
           COUNT(EndTerminal) AS drop_offs_Count
    FROM trip_data
    GROUP BY EndTerminal
) AS trip
INNER JOIN station_data AS station
    ON station.station_id = trip.end_terminal
    ORDER BY drop_offs_Count ASC
```

Output:

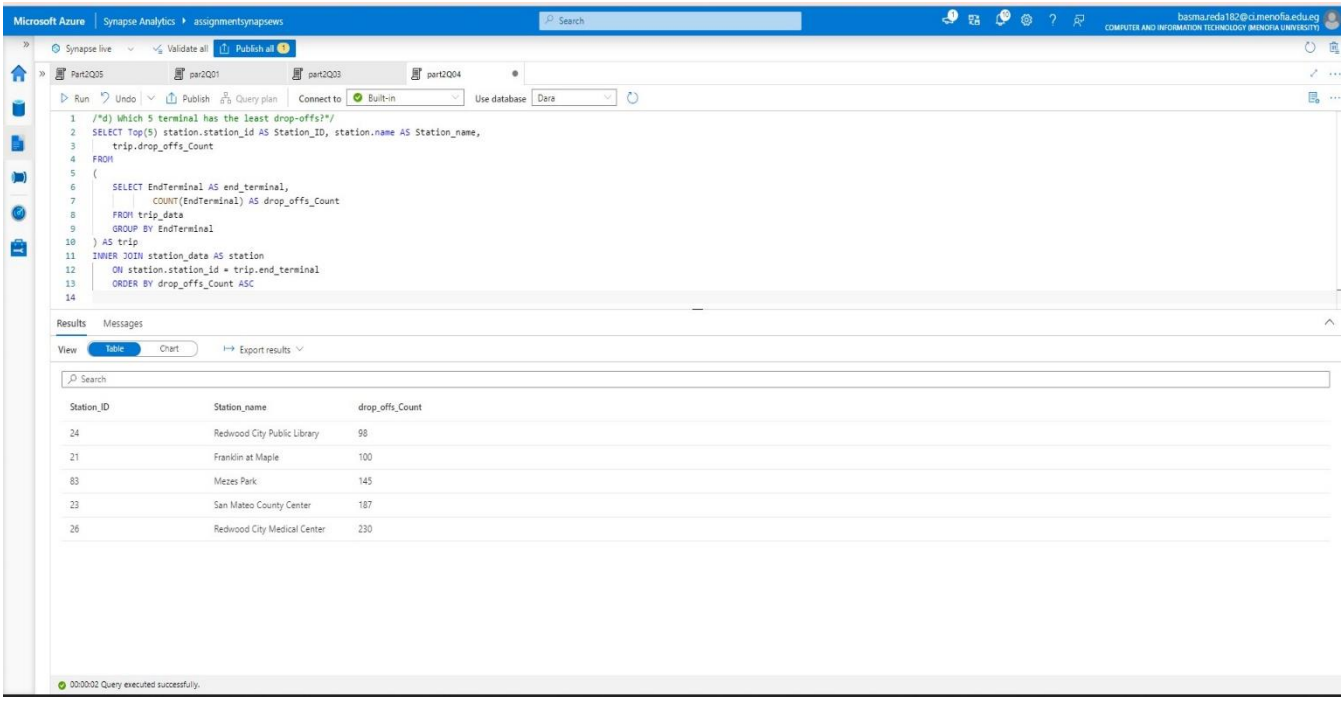
Results Messages

View Table Chart Export results

Search

Station_ID	Station_name	drop_offs_Count
24	Redwood City Public Library	98
21	Franklin at Maple	100
83	Mezes Park	145
23	San Mateo County Center	187
26	Redwood City Medical Center	230

Screenshot from Azure:



Save the results in a CSV file.

E) Produce the monthly summary of bike rentals (format -month/year ex. 06/2020)

Query:

```
-- Produce the monthly summary of bike rentals (format - month/year ex. 06/2020)
SELECT STR(MONTH(StartDate))+'/'+STR(YEAR(StartDate)) AS month_year, COUNT([TripID]) AS
Trip_Count,
SUM(Duration) AS Total_Duration_count,
COUNT(BikeNum) AS Total_Bikes_count,
sum(case when SubscriberType = 'Customer' then 1 else 0 end) as  Total_customers_count,
sum(case when SubscriberType = 'Subscriber' then 1 else 0 end) as Total_subscribers_count
FROM trip_data
GROUP BY STR(MONTH(StartDate))+'/'+STR(YEAR(StartDate) )
```

Output:

Results Messages

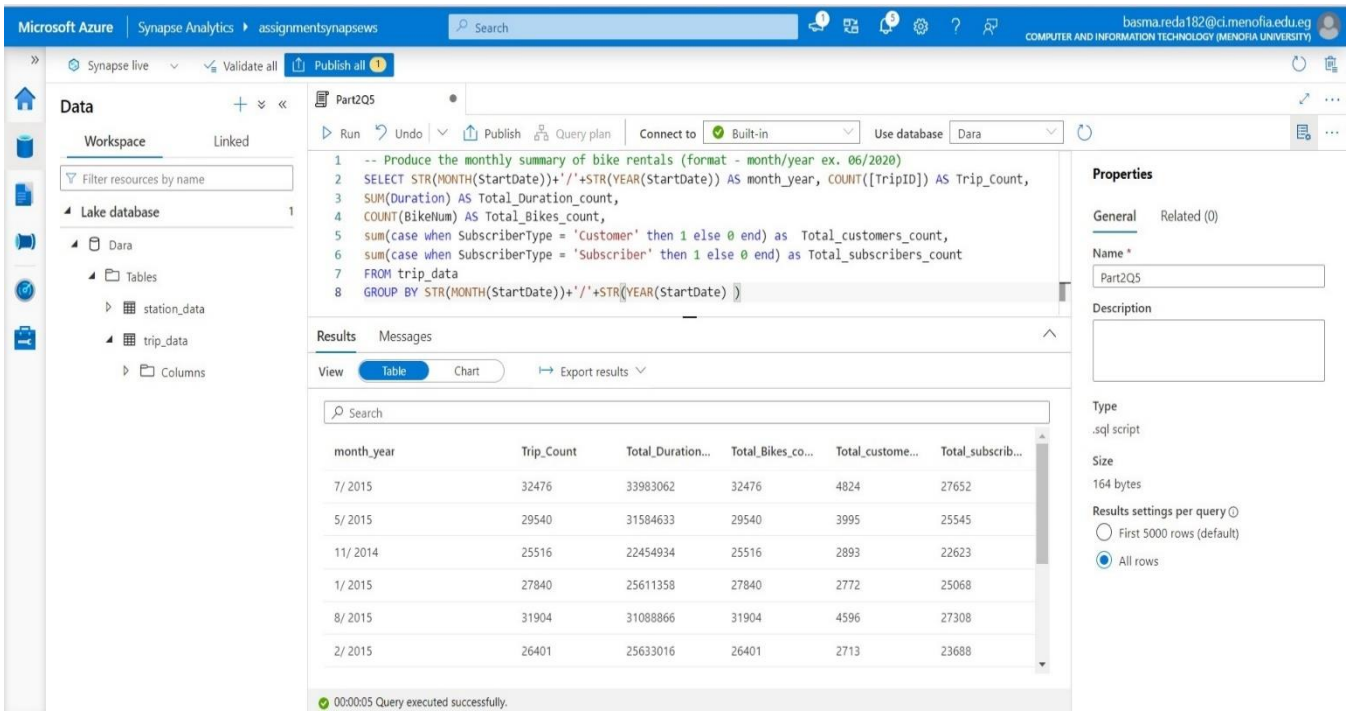
View Table Chart Export results

Search

month_year	Trip_Count	Total_Duration...	Total_Bikes_co...	Total_custome...	Total_subscrib...
7/ 2015	32476	33983062	32476	4824	27652
5/ 2015	29540	31584633	29540	3995	25545
11/ 2014	25516	22454934	25516	2893	22623
1/ 2015	27840	25611358	27840	2772	25068
8/ 2015	31904	31088866	31904	4596	27308
2/ 2015	26401	25633016	26401	2713	23688

00:00:05 Query executed successfully.

Screenshot from Azure:



Save the results in a CSV file.

Part 3: Definitions

1) *Please compare briefly, based on at least 3 criteria, the differences in architecture between Apache Spark Structured Streaming and Azure Event Hubs & Synapse Analytics.*

	Spark Structured Streaming	Azure Event Hubs	Azure Synapse Analytics
Programming Languages	Uses the data's API in Java, Scala, R or python for streaming, stream-to-batch join, event-time windows and more.	Uses different client SDKs including .NET, java, python, JavaScript, Go and C.[3]	T-SQL, KQL, Python, Scala, Spark SQL, and .Net , serverless or dedicated resources.[1]
Scalability	It is scalable, thus, the received data can be triggered to append in a limitless constant stream of data.[2]	Can be controlled in scalability level and the timing, that can scale from megabytes to terabytes of data.[3]	Can acquire scalable real-time insights that includes old and new data.
Recovery	It uses checkpoints and logs' mechanisms to make certain of End-to-End connection when semantics are failing.	Uses geo-disaster recovery and geo-replication so that the data can be retrieved in any emergency.	It supports 8 hours recovery point objective (RPO). Data can be stored in the main region and any of the snapshots that was taken in the last 7 days can be retrieved.[1]
Reusability	The code and results can be moved to another cloud provider to be reused.[2]	The code cannot be extracted from Azure to be used, as it is a privatized language and solution.[3]	It can use "Flowlets" feature that are reusable containers. Synapse pipelines and dataflows can handle the checkpoints by marking "enable change feed" feature.
Security	No security is enabled but can be added through configurations.	Can encrypt data at rest by "managed-keys" or "customer-managed-keys".	It has four levels of security: Network security, access management, threat protection and information protection. Security includes threat protection by monitor logs and event hubs, ability to use Microsoft defender for cloud, TLS Encryption-in-transit, TDE Encryption-at-rest, can use Azure key vault and dynamic data masking.[1]

2) *Describe briefly 3 benefits of Azure Synapse Analytics over Apache Spark. Illustrate them briefly with some use cases.*

Synapse supports Deeply integrated Apache Spark and U-SQL has some advantages over spark:

Firstly, Azure Synapse Analytics supports SQL, T-SQL, Spark SQL, Python, Java, Scala and .NET but spark only supports Python and Scala, so, Synapse can be used in a project using a wide range of languages and different scripts. Secondly, Azure synapse is easier to use it enables the database to be enabled from one place which results lower cost, less managing complexity and being able to access and manage multiple systems at once, on the other hand, Apache spark keeps adding cost as long as the cluster is running, so Synapse is cheaper after all. Thirdly, Azure Synapse has the option to examine data using SQL and can deal with and ingest relational and non-relational data such as data warehouse and data lake respectively. Synapse makes it possible for users to access and analyze structured and unstructured data. Generally, Apache Spark and Azure Synapse can be categorized as "Big Data" tools.[4]

3) *What are the 5 characteristics of Azure Data Lake Storage that distinguish it from other Distributed Dataset Storage infrastructures such as Hadoop?*

Data can be combined from multiple resources into one

Data lake storage can combine many data resources into a single location on-premises infrastructure, on the other hand, it can be difficult to combine data in Hadoop.

Hadoop usage for the cloud

Azure data lake can make a Hadoop solution by changing the constraints into more usual deployments. [5]

Data size

In Azure data lake, there is no data limit unlike other cloud choices it is limited to few terabytes of data. Nevertheless, Hadoop can deal with petabytes of data in storage and processing.[5]

Security

In Data Lake, data is secured in transit and at rest. It has many security features like multi-factor authentication (MFA) and single sign-on (SSO), but, Hadoop does not deal with data security, adding to that, data is stored as plain text in Hadoop Distributed File System (HDFS) which is a huge risk and an open book for attackers.

High-speed outputs

Azure Data Lake provides high speed connections needed for some applications like mobile phones and gadgets are mostly made in short data transmissions, this means higher credibility in large scale usages.[5]

Ability to use parallel processing

Parallel processing is supported in Azure Data Lake that outputs a relatively close performance to that of on-premises Hadoop solution.[5]

References:

[1] Difference between synapse and DataBricks (2021, October 12). Microsoft.

differnce between synapse and databricks - Microsoft Q&A

[2] Spark Streaming vs. Structured Streaming (2019, March 12). Big Data Zone.

Spark Streaming vs. Structured Streaming - DZone Big Data

[3] Overview of features - Azure Event Hubs - Azure Event Hubs (2022, September 12). Microsoft.

Overview of features - Azure Event Hubs - Azure Event Hubs | Microsoft Learn

[4] Azure synapse Analytics (2021, December 11).

What Is Azure Synapse Analytics, And Why Should You Use It? (afon.com.sg)

[5*] key features of Azure data Lake (2022, September 30). Microsoft*.

Azure Data Lake Storage Gen2 Introduction | Microsoft Learn