



---

## 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

<b>Personal Ethics &amp; Academic Integrity Statement .....</b>	<b>2</b>
<b>Part 1: Event Hub Analytics .....</b>	<b>3</b>
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. ....	12
B) Go to your event hub's namespace and show that the entire message set was received. Provide one or more screenshots. ....	17
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. ....	18
<b>Part 2: Azure Synapse Analytics: .....</b>	<b>21</b>
A) Top 20 zip codes for bike up.....	28
B) Monthly duration aggregate across the rental subscriber types, ordered in descending order of the busiest months (use a meaningful measure for the aggregate) .....	30
C) What are the top 5 busiest terminals for bike pickup?.....	31
D) Which 5 terminals has the least drop-offs? .....	32
E) Produce the monthly summary of bike rentals (format -month/year ex. 06/2020).....	33
<b>Part 3: Definitions .....</b>	<b>35</b>
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.....	35
2) Describe briefly 3 benefits of Azure Synapse Analytics over Apache Spark. Illustrate them briefly with some use cases. ....	36
3) What are the 5 characteristics of Azure Data Lake Storage that distinguish it from other Distributed Dataset Storage infrastructures such as Hadoop? .....	37
<b>References.....</b>	<b>38</b>

## 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.

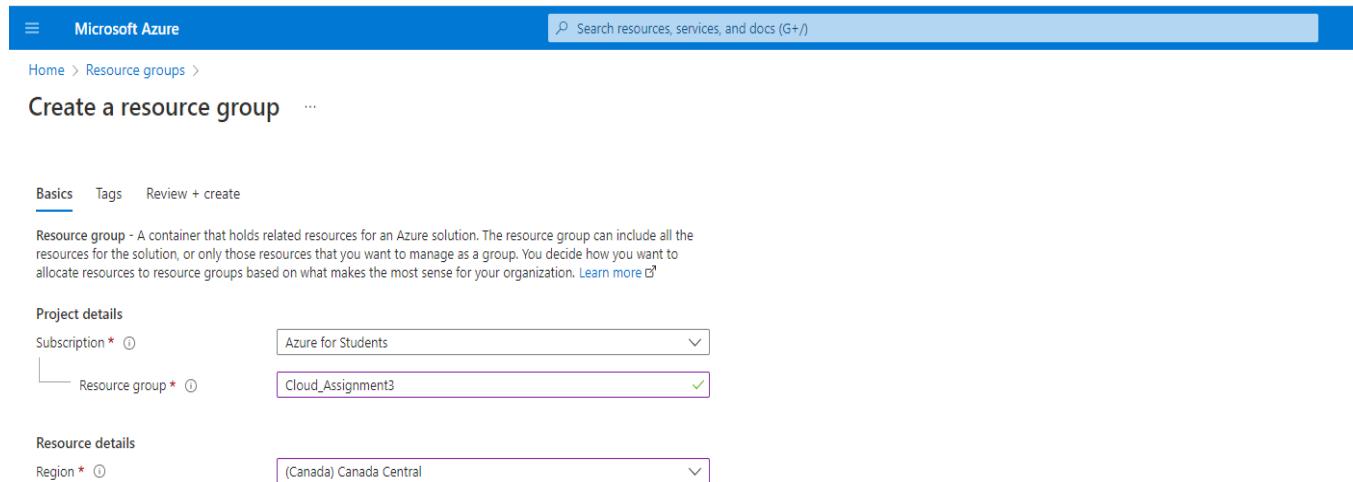
# Part 1: Event Hub Analytics

## Setting up the environment:

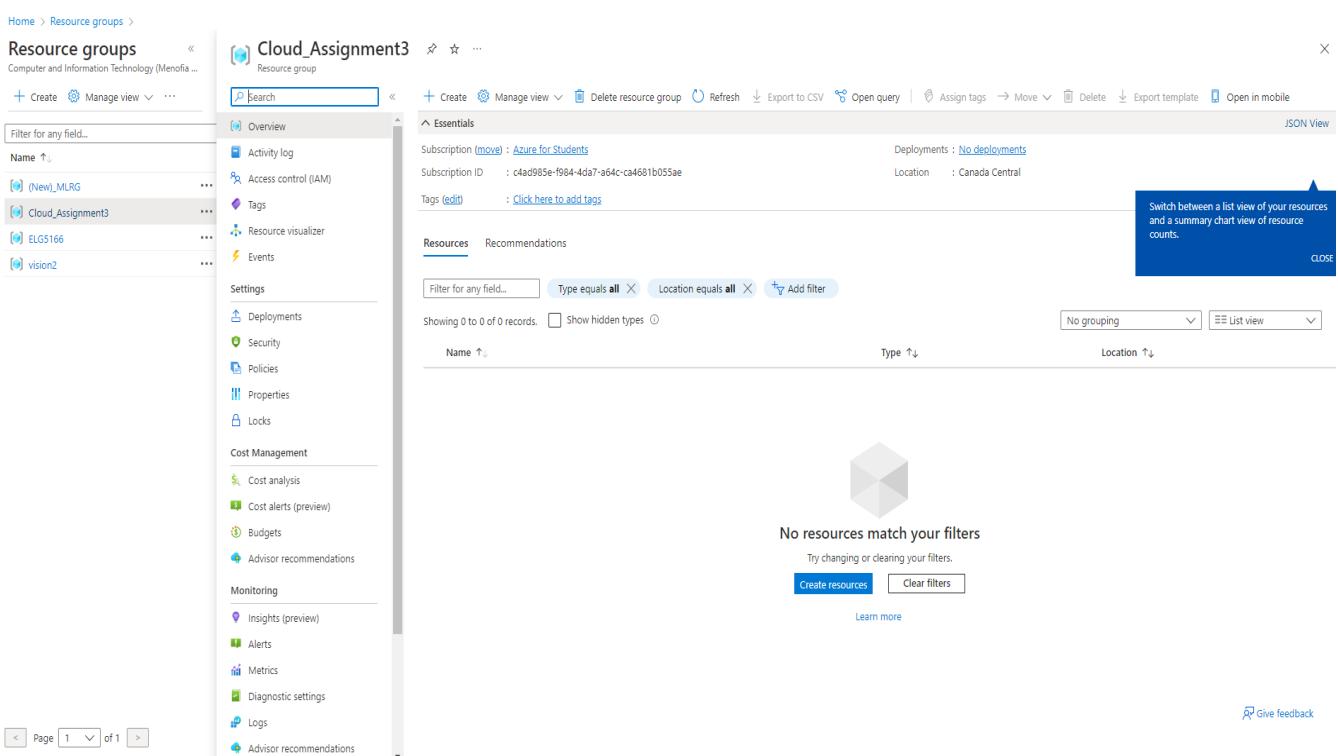
- 1- Signup into Azure as a student and claim the credit.
- 2- Install visual studio community edition.
- 3- Install visual studio code dependencies.

## After setting up the environment:

### 1- Create a resource group:



The screenshot shows the 'Create a resource group' wizard in the Microsoft Azure portal. The top navigation bar is blue with the text 'Microsoft Azure' and a search bar. Below the navigation, the breadcrumb path 'Home > Resource groups >' is visible. The main title 'Create a resource group' is in bold. Below the title, there are three tabs: 'Basics' (selected), 'Tags', and 'Review + create'. The 'Basics' tab contains 'Project details' and 'Resource details' sections. In 'Project details', the 'Subscription' dropdown is set to 'Azure for Students' and the 'Resource group' input field is 'Cloud\_Assignment3'. In 'Resource details', the 'Region' dropdown is set to '(Canada) Canada Central'. The bottom of the wizard shows a summary of the selected resource group details.



The screenshot shows the 'Resource groups' blade for the 'Cloud\_Assignment3' resource group. The left sidebar lists other resource groups: '(New)\_MLRG', 'Cloud\_Assignment3' (selected), 'ELGS166', and 'vision2'. The main content area shows the 'Cloud\_Assignment3' resource group details. The 'Overview' section includes a search bar, a toolbar with 'Create', 'Manage view', 'Delete resource group', 'Refresh', 'Export to CSV', 'Open query', 'Assign tags', 'Move', 'Delete', 'Export template', and 'Open in mobile'. The 'Essentials' section displays the subscription (Azure for Students), subscription ID (c44d9b5e-f984-4da7-a64c-ca4681b055ae), location (Canada Central), and tags (empty). The 'Resources' section shows a table with columns 'Name', 'Type', 'Status', and 'Last activity'. A message at the bottom states 'No resources match your filters' with a 'Create resources' button. The bottom of the blade shows a navigation bar with 'Page 1 of 1' and a 'Give feedback' link.

## 2- Create Event Hub Namespace:

Home > Event Hubs >

### Create Namespace

Event Hubs

Basics Advanced Networking Tags Review + create

**Project Details**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* Azure for Students

Resource group \* Cloud\_Assignment3 [Create new](#)

**Instance Details**

Enter required settings for this namespace, including a price tier and configuring the number of units (capacity).

Namespace name \* CloudAssignment3 .servicebus.windows.net

Location \* Canada Central   
 ⓘ The region selected supports Availability zones. Your namespace will have Availability Zones enabled. [Learn more](#).

Pricing tier \* Basic (~\$11 USD per TU per Month)   
 [Browse the available plans and their features](#)

Throughput Units \* 1

[Review + create](#) [< Previous](#) [Next: Advanced >](#)

Home >

### CloudAssignment3 | Overview

Deployment

Search [Delete](#) [Cancel](#) [Redeploy](#) [Download](#) [Refresh](#)

Overview [Inputs](#) [Outputs](#) [Template](#)

✓ Your deployment is complete

Deployment name: CloudAssignment3  
Subscription: Azure for Students  
Resource group: Cloud\_Assignment3

Start time: 11/20/2022, 1:53:44 AM  
Correlation ID: 9a3c129c-d97a-498f-bf97-768da151b27a [D](#)

Deployment details [Next steps](#)

[Go to resource](#)

The Namespace is empty, because no streams done yet.

The screenshot shows the Azure Event Hubs Namespace overview page for the CloudAssignment3 resource. The page includes a navigation bar, a search bar, and a main content area with tabs for Overview, Essentials, Requests, Messages, and Throughput. The Overview tab is selected. The main content area displays the following details:

Setting	Value
Resource group (move)	Cloud Assignment3
Status	Active
Location	Canada Central
Subscription (move)	Azure for Students
Subscription ID	c4ad95e-f984-4da7-a64c-ca4681b055ae
Host name	CloudAssignment3.servicebus.windows.net
Created	Sunday, November 20, 2022 at 01:53:51 GMT+2
Updated	Sunday, November 20, 2022 at 01:54:44 GMT+2
Zone Redundancy	Enabled
Pricing tier	Basic
Throughput Units	1 unit
Auto-inflate throughput	Not Supported
Local Authentication	Enabled

Below the table, there is a section for 'NAMESPACE CONTENTS' with three status indicators: 0 EVENT HUBS (NOT SUPPORTED), KAFKA SURFACE (NOT SUPPORTED), and ZONE REDUNDANCY (ENABLED). The 'Requests' chart shows data for the last 1 hour, with values for Incoming Requests, Successful Requests, Server Errors, and User Errors. The 'Messages' chart shows data for the last 1 hour, with values for Incoming Messages, Outgoing Messages, Captured Messages, and Capture Backlog. The 'Throughput' chart shows data for the last 1 hour, with values for Incoming Bytes, Outgoing Bytes, and Captured Bytes.

### 3- Create Event Hub for bike data:

The screenshot shows the 'Create Event Hub' wizard in the Azure portal, currently on the 'Basics' tab. The page title is 'Create Event Hub' and the sub-page title is 'Event Hubs'. The 'Basics' tab is selected, showing the following configuration:

- Event Hub Details:** Enter required settings for this event hub, including partition count and message retention.
- Name:** Bike\_Data
- Partition count:** 2
- Message retention:** 1

At the bottom of the page, there are navigation buttons: 'Review + create', '< Previous', and 'Next: Capture >'.

**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 >](#)**CloudAssignment3 | Event Hubs** ...

Event Hubs Namespace

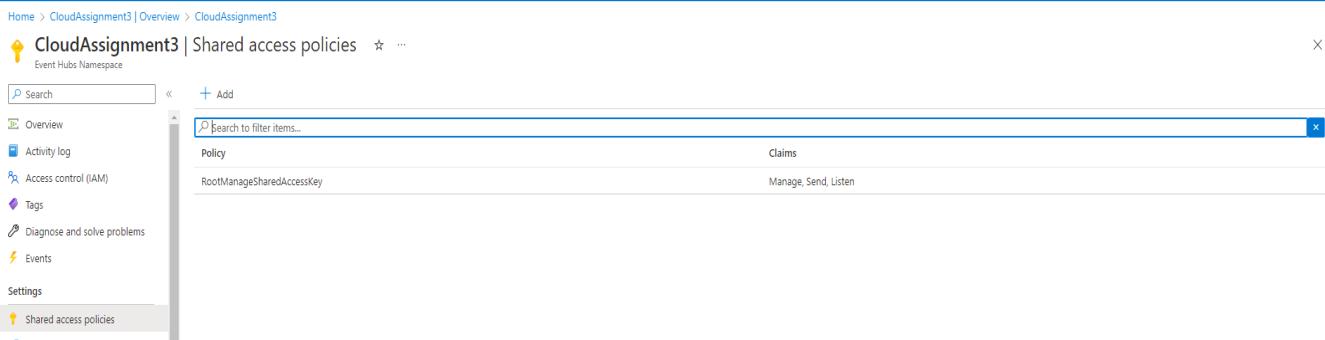
X

[Search](#)[+ Event Hub](#) [Search to filter items...](#)

Name	Status	Message retention	Partition count
bike_data	Active	1 day	2

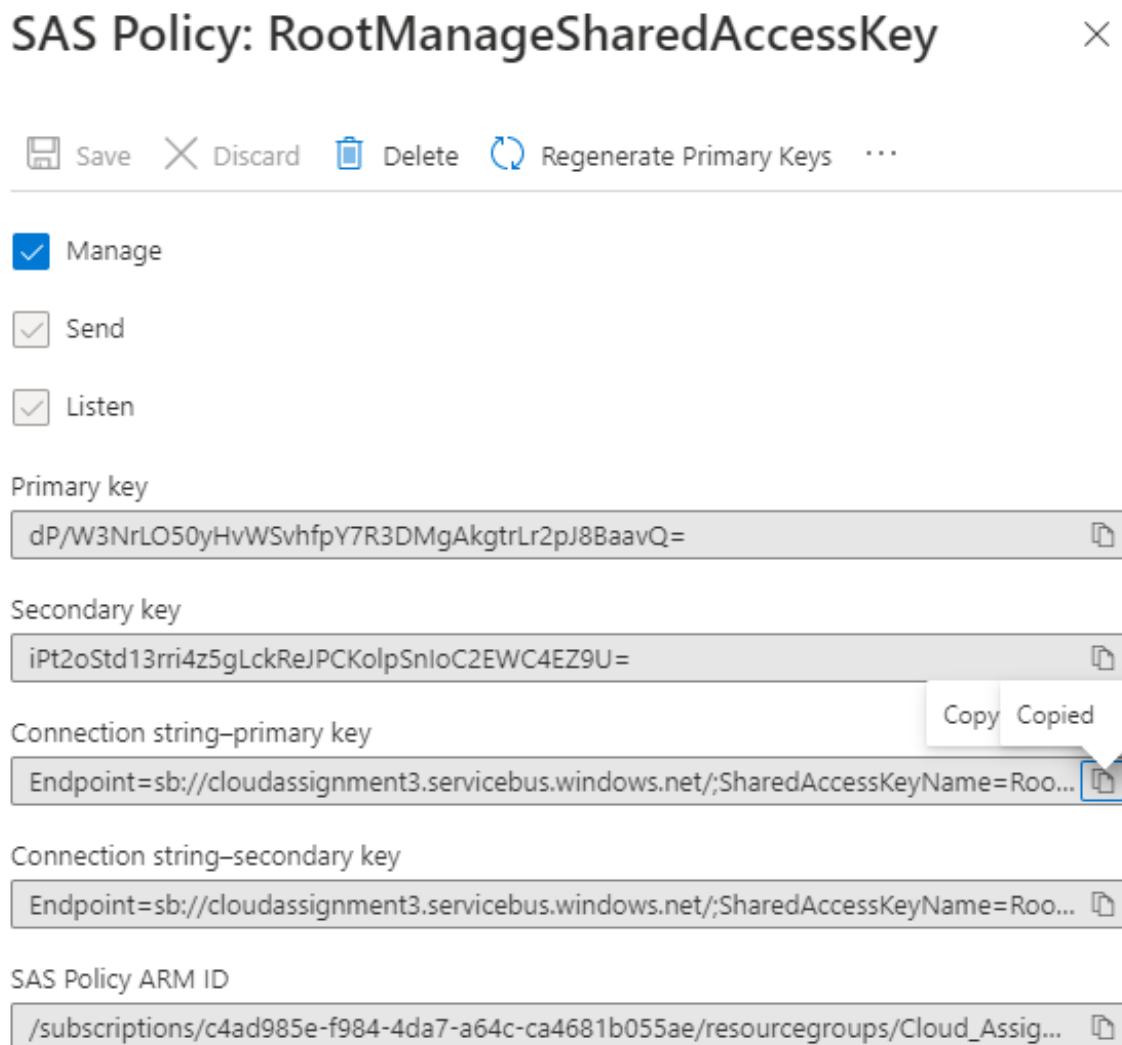
**Overview**[Activity log](#)[Access control \(IAM\)](#)[Tags](#)[Diagnose and solve problems](#)**Events****Settings**[Shared access policies](#)[Scale](#)[Geo-Recovery](#)[Encryption](#)[Configuration](#)[Properties](#)[Locks](#)**Entities**[Event Hubs](#)**Monitoring**[Alerts](#)[Metrics](#)[Diagnostic settings](#)[Logs](#)**Automation**[Tasks \(preview\)](#)[Export template](#)

#### 4- Go to the shared access policies:



The screenshot shows the Azure portal interface for a Cloud Assignment3 Event Hubs Namespace. The 'Shared access policies' section is selected in the left sidebar. A single policy, 'RootManageSharedAccessKey', is listed with the claim 'Manage, Send, Listen'.

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



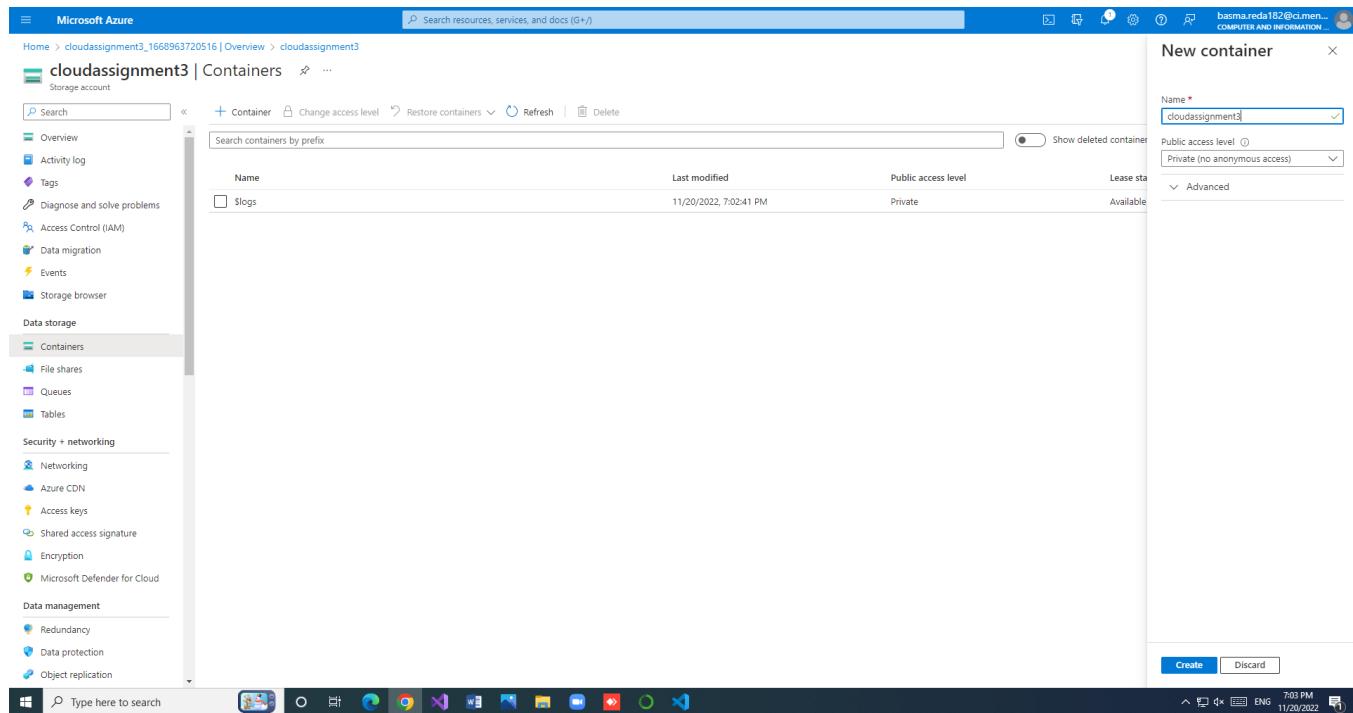
The screenshot shows the 'SAS Policy: RootManageSharedAccessKey' settings page. The 'Primary key' field contains the value 'dP/W3NrLO50yHvWSvhfpY7R3DMgAkgtrLr2pJ8BaavQ=' and has a 'Copy' button to its right. The 'Secondary key' field contains the value 'iPt2oStd13rr14z5gLckReJPCKolpSnloC2EWC4EZ9U=' and also has a 'Copy' button. Below these, a 'Connection string-primary key' field contains the value 'Endpoint=sb://cloudassignment3.servicebus.windows.net/;SharedAccessKeyName=Roo...' with a 'Copy' button. Another 'Connection string-secondary key' field with a 'Copy' button is also present. At the bottom, the 'SAS Policy ARM ID' is shown as a long URL.

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

The screenshot shows the 'Create a storage account' wizard in the Microsoft Azure portal. The 'Project details' step is completed, showing a subscription of 'Azure for Students' and a resource group named 'Cloud\_Assignment3'. The 'Instance details' step is in progress, with the storage account name set to 'cloudassignment3', region set to '(Canada) Canada Central', and performance set to 'Standard'. Redundancy is set to 'Geo-redundant storage (GRS)' with the 'Make read access to data available in the event of regional unavailability' checkbox checked. The 'Review' button is visible at the bottom.

The screenshot shows the 'cloudassignment3\_1668963720516 | Overview' page in the Microsoft Azure portal. The deployment is marked as 'complete' with a green checkmark. Deployment details are listed: name 'cloudassignment3\_1668963720516', subscription 'Azure for Students', and resource group 'Cloud\_Assignment3'. The start time is 11/20/2022, 7:02:09 PM, and the correlation ID is cb89a2fe-2e02-4ef6-85f8-c040fddeabdb. A 'Go to resource' button is present at the bottom.

## 7- Create a container:



Microsoft Azure

Home > cloudassignment3\_1668963720516 | Overview > cloudassignment3

cloudassignment3 | Containers

Storage account

Search

+ Container Change access level Restore containers Refresh Delete

Search containers by prefix

Name	Last modified	Public access level	Lease state
slogs	11/20/2022, 7:02:41 PM	Private	Available

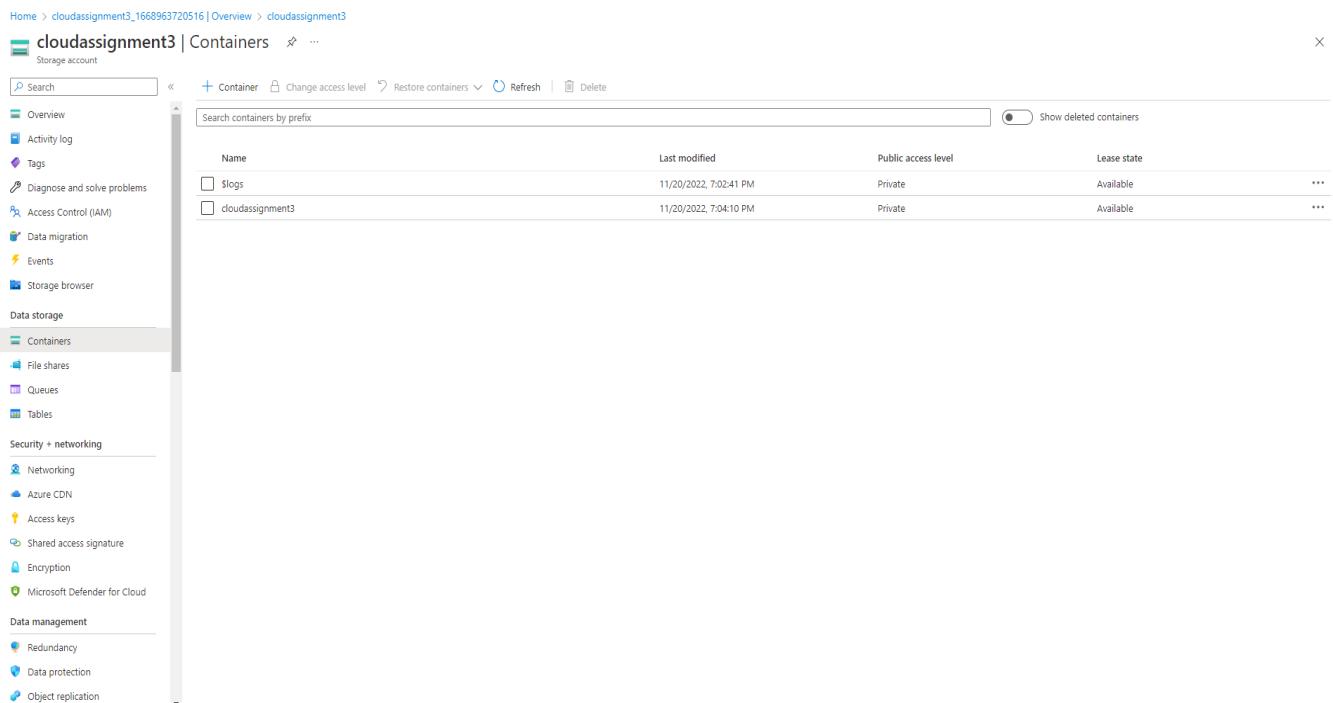
New container

Name:

Public access level:  Private (no anonymous access)

Advanced

Create Discard



Home > cloudassignment3\_1668963720516 | Overview > cloudassignment3

cloudassignment3 | Containers

Storage account

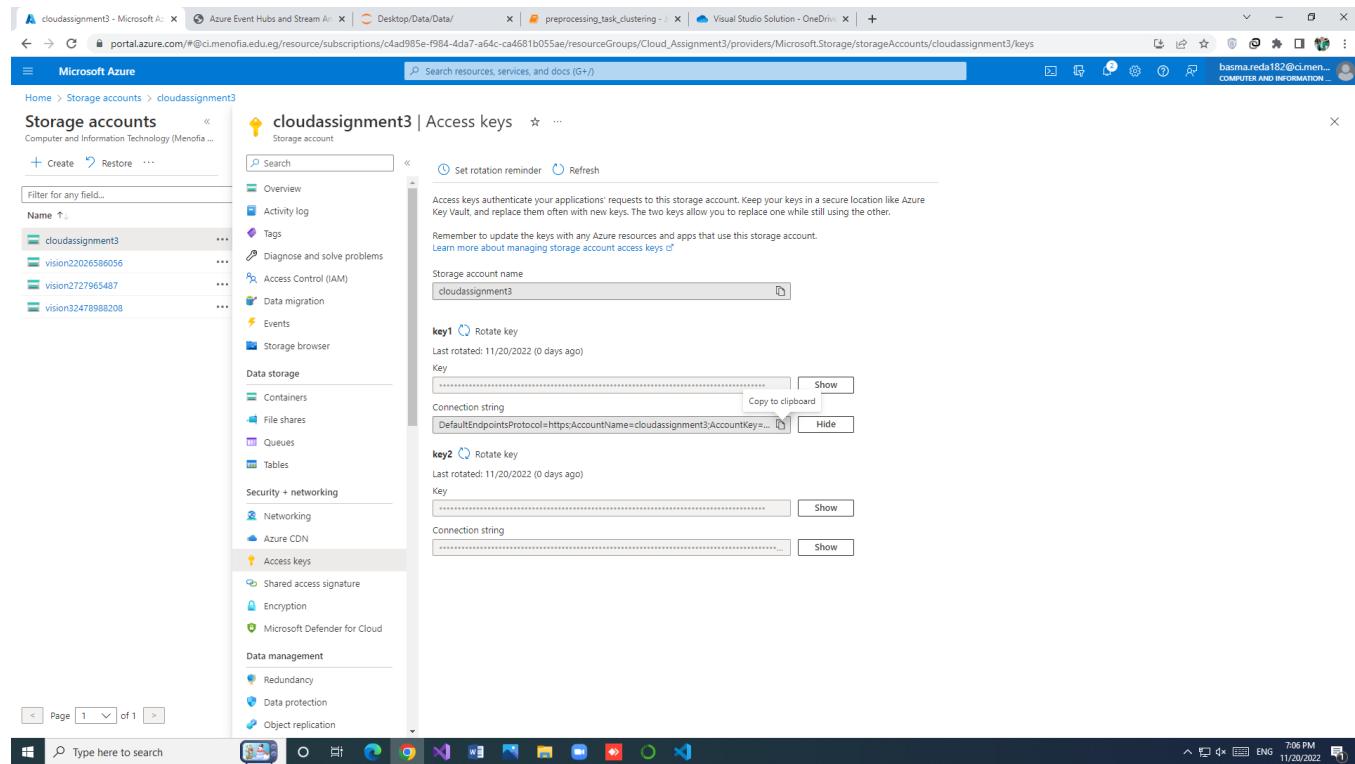
Search

+ Container Change access level Restore containers Refresh Delete

Search containers by prefix

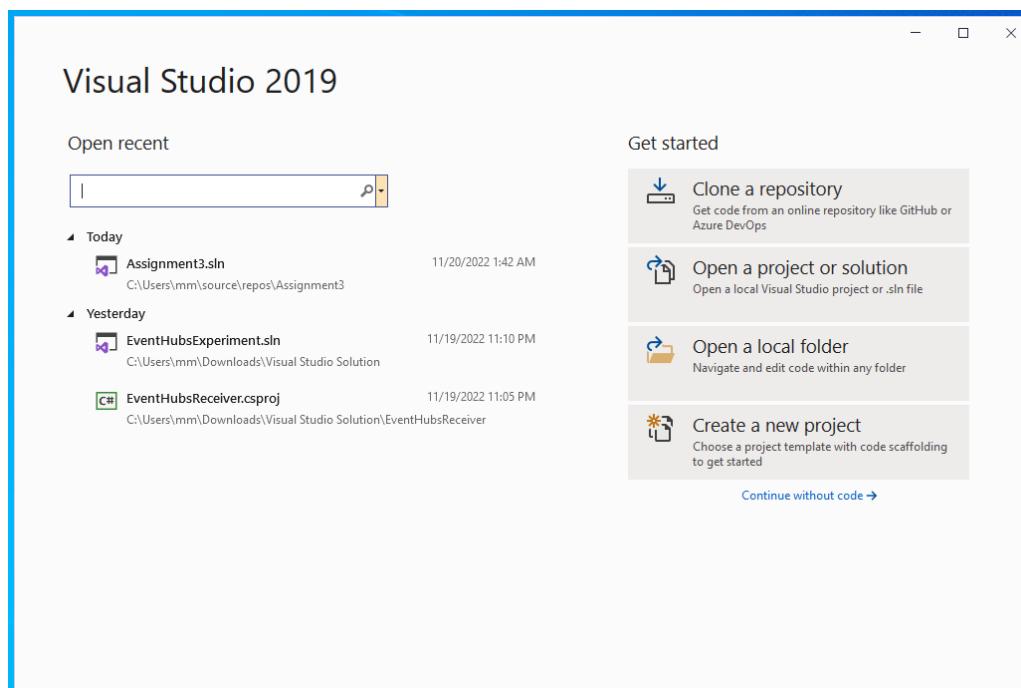
Name	Last modified	Public access level	Lease state
slogs	11/20/2022, 7:02:41 PM	Private	Available
cloudassignment3	11/20/2022, 7:04:10 PM	Private	Available

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

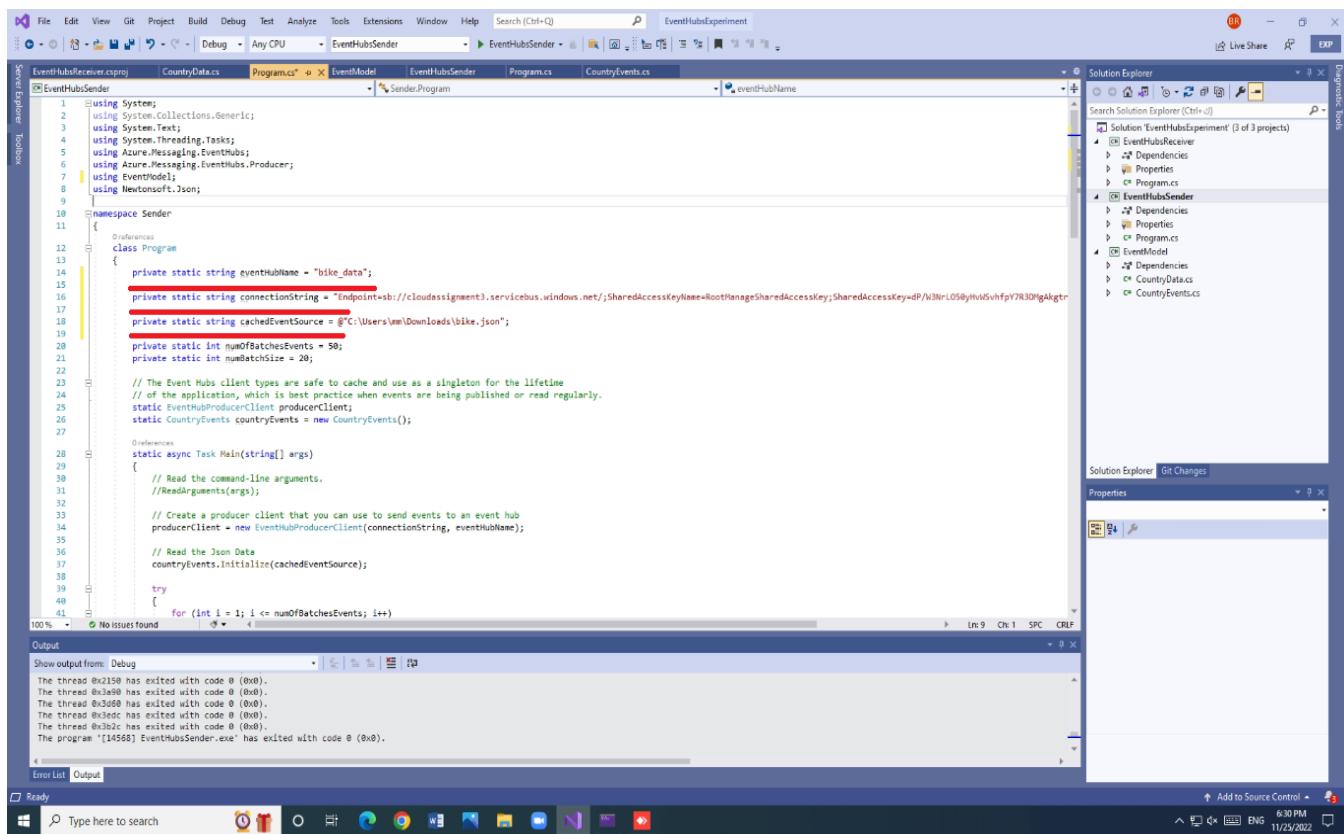


The screenshot shows the Azure Storage Accounts page for the 'cloudassignment3' account. The 'Access keys' section is selected. It displays two keys: 'key1' and 'key2'. Each key has a 'Rotate key' button, a 'Last rotated' timestamp (11/20/2022), and a 'Key' field with a 'Show' button. Below each key is a 'Connection string' field with a 'Copy to clipboard' button. The 'key1' connection string is partially visible as 'DefaultEndpointsProtocol=https;AccountName=cloudassignment3;AccountKey=...'. The 'key2' connection string is also partially visible. The left sidebar shows other storage account options like Containers, File shares, Queues, and Tables.

## 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:



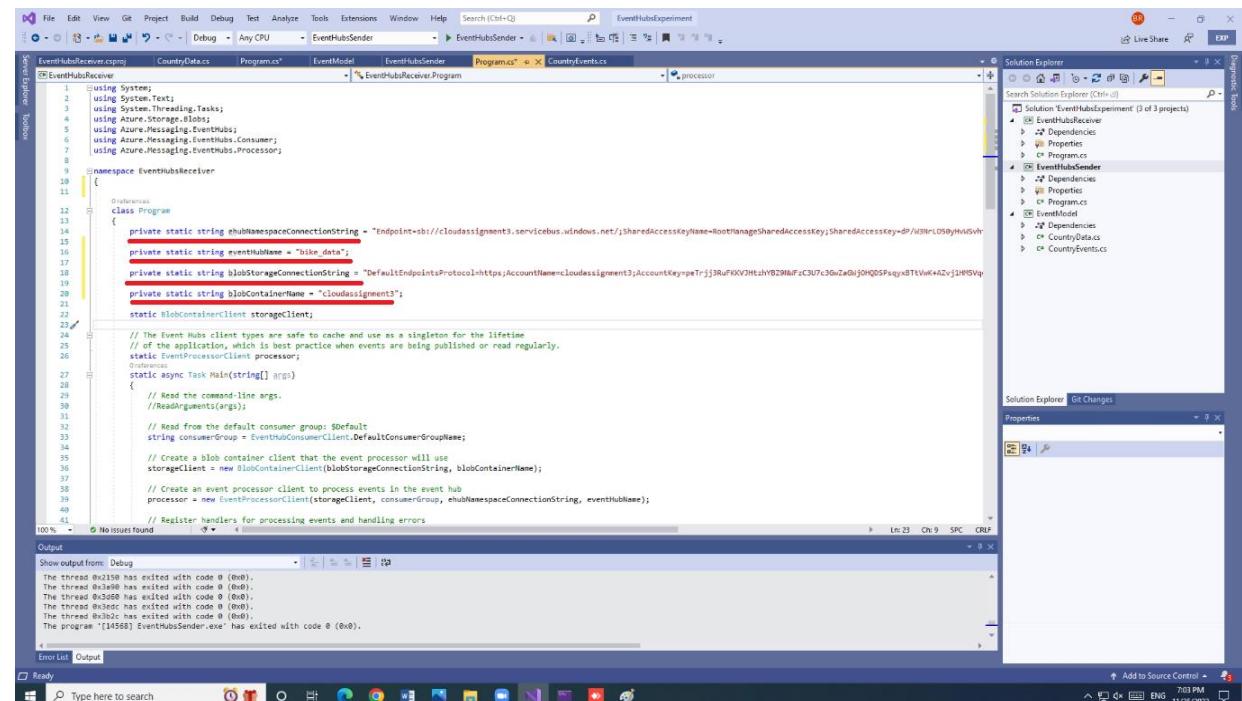
```
1 using System;
2 using System.Collections.Generic;
3 using System.Text;
4 using System.Threading.Tasks;
5 using Azure.Messaging.EventHubs;
6 using Azure.Messaging.EventHubs.Producer;
7 using EventModel;
8 using Newtonsoft.Json;
9
10 namespace Sender
11 {
12     class Program
13     {
14         private static string eventHubName = "bike_data";
15
16         private static string connectionString = "Endpoint=sb://cloudassignment3.servicebus.windows.net/;SharedAccessKeyName=RootManageSharedAccessKey;SharedAccessKey=dP/N3NrL058yHv8vhfpY7R3Dg4kgr";
17
18         private static string cachedEventSource = @"C:\Users\mn\Downloads\bike.json";
19
20         private static int numOfBatchesEvents = 50;
21
22         private static int batchSizeSize = 20;
23
24         // The Event Hubs client types are safe to cache and use as a singleton for the lifetime
25         // of the application, which is best practice when events are being published or read regularly.
26         static EventHubProducerClient producerClient;
27         static CountryEvents countryEvents = new CountryEvents();
28
29         static async Task Main(string[] args)
30         {
31             // Read the command-line arguments.
32             //ReadArguments(args);
33
34             // Create a producer client that you can use to send events to an event hub
35             producerClient = new EventHubProducerClient(connectionString, eventHubName);
36
37             // Read the Json Data
38             countryEvents.Initialize(cachedEventSource);
39
40             try
41             {
42                 for (int i = 1; i <= numOfBatchesEvents; i++)
43             }
44         }
45     }
46 }
```

Output

```
Show output from: Debug
The thread 0x2150 has exited with code 0 (0x0).
The thread 0x3e00 has exited with code 0 (0x0).
The thread 0x3600 has exited with code 0 (0x0).
The thread 0x3e0c has exited with code 0 (0x0).
The thread 0x3e2c has exited with code 0 (0x0).
The program '[14568] EventHubSender.exe' has exited with code 0 (0x0).
```

Error List Output

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



```
1 using System;
2 using System.Text;
3 using System.Threading.Tasks;
4 using Azure.Storage.Blobs;
5 using Azure.Messaging.EventHubs;
6 using Azure.Messaging.EventHubs.Consumer;
7 using Azure.Messaging.EventHubs.Processors;
8
9 namespace EventHubReceiver
10 {
11     class Program
12     {
13         private static string eventHubNamespaceConnectionString = "Endpoint=sb://cloudassignment3.servicebus.windows.net/;SharedAccessKeyName=RootManageSharedAccessKey;SharedAccessKey=dP/N3NrL058yHv8vhfpY7R3Dg4kgr";
14
15         private static string eventHubName = "bike_data";
16
17         private static string blobStorageConnectionString = "DefaultEndpointsProtocol=https;AccountName=cloudassignment3;AccountKey=peTrjj3Ru9Xv3HtshB29Mfcz3U73Gw2aGjJ0HQD9Psqy8TtVwv+AlvJ1H5Vq";
18
19         private static string blobContainerName = "cloudassignment";
20
21         static BlobContainerClient storageClient;
22
23         // The Event Hubs client types are safe to cache and use as a singleton for the lifetime
24         // of the application, which is best practice when events are being published or read regularly.
25         static EventProcessorClient processor;
26
27         static async Task Main(string[] args)
28         {
29             // Read the command-line args.
30             //ReadArguments(args);
31
32             // Read from the default consumer group: $Default
33             string consumerGroup = EventHubConsumerClient.DefaultConsumerGroupName;
34
35             // Create a blob container client that the event processor will use
36             storageClient = new BlobContainerClient(blobStorageConnectionString, blobContainerName);
37
38             // Create an event processor client to process events in the event hub
39             processor = new EventProcessorClient(storageClient, consumerGroup, hubNamespaceConnectionString, eventHubName);
40
41             // Register handlers for processing events and handling errors
42         }
43     }
44 }
```

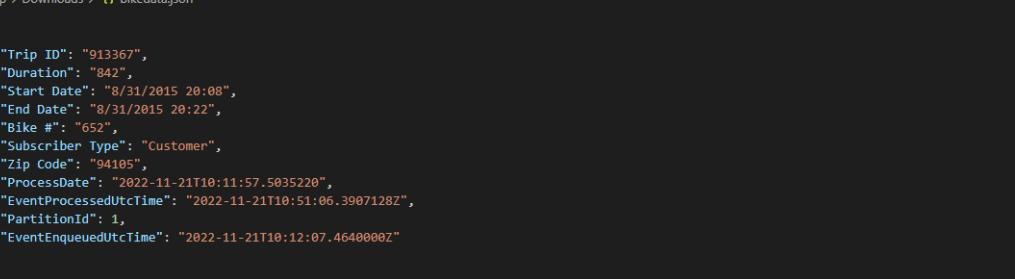
Output

```
Show output from: Debug
The thread 0x2150 has exited with code 0 (0x0).
The thread 0x3e00 has exited with code 0 (0x0).
The thread 0x3600 has exited with code 0 (0x0).
The thread 0x3e0c has exited with code 0 (0x0).
The thread 0x3e2c has exited with code 0 (0x0).
The program '[14568] EventHubReceiver.exe' has exited with code 0 (0x0).
```

Error List Output

- 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:



```
1  [
2  {
3    "Trip ID": "913367",
4    "Duration": "842",
5    "Start Date": "8/31/2015 20:08",
6    "End Date": "8/31/2015 20:22",
7    "Bike #": "652",
8    "Subscriber Type": "Customer",
9    "Zip Code": "94105",
10   "ProcessDate": "2022-11-21T10:11:57.5035220",
11   "EventProcessedUtcTime": "2022-11-21T10:51:06.3907128Z",
12   "PartitionId": 1,
13   "EventEnqueuedUtcTime": "2022-11-21T10:12:07.4640000Z"
14 },
15 {
16   "Trip ID": "913441",
17   "Duration": "387",
18   "Start Date": "8/31/2015 21:39",
19   "End Date": "8/31/2015 21:46",
20   "Bike #": "383",
21   "Subscriber Type": "Subscriber",
22   "Zip Code": "94104",
23   "ProcessDate": "2022-11-21T10:11:57.5033850",
24   "EventProcessedUtcTime": "2022-11-21T10:51:03.3438120Z",
25   "PartitionId": 1,
26   "EventEnqueuedUtcTime": "2022-11-21T10:12:07.4640000Z"
27 },
28 {
29   "Trip ID": "913419",
30   "Duration": "704",
31   "Start Date": "8/31/2015 20:58",
32   "End Date": "8/31/2015 21:10",
33   "Bike #": "570"
```

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:**

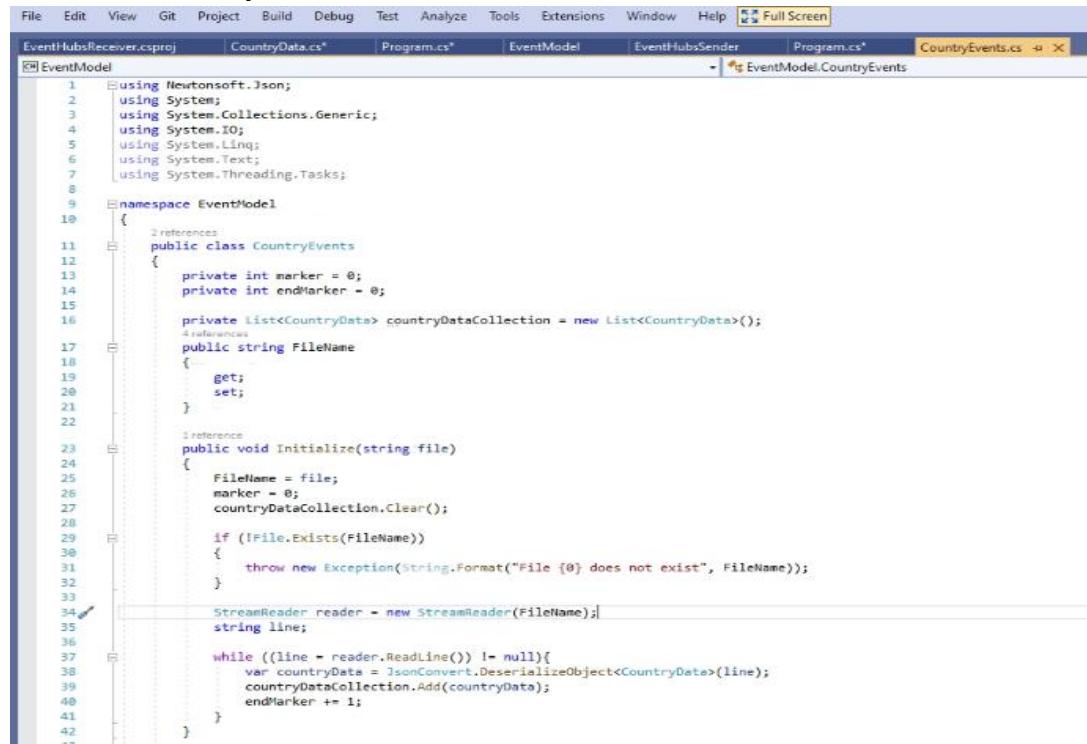
### 3- Change the class data to be able to send these variables:

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



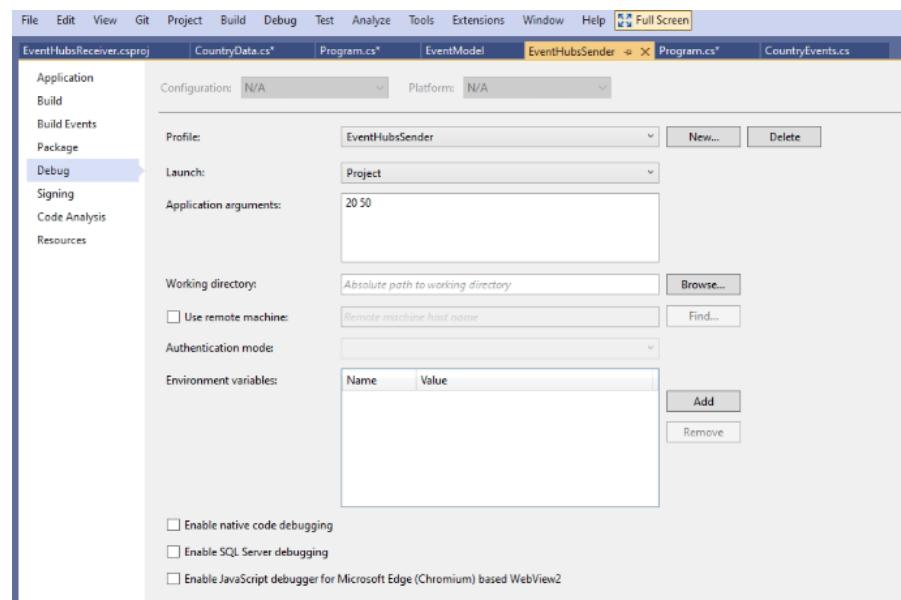
```
File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Full Screen
EventHubsReceiver.csproj | CountryData.cs* | Program.cs* | EventModel | EventHubSender | Program.cs* | CountryEvents.cs
EventModel
1  using System;
2  using Newtonsoft.Json;
3
4  namespace EventModel
5  {
6      public class CountryData
7      {
8          [JsonProperty("Trip ID")]
9          public string TripID { get; set; }
10
11         [JsonProperty("Duration")]
12         public string Duration { get; set; }
13
14         [JsonProperty("Start Date")]
15         public string Start_Date { get; set; }
16
17         [JsonProperty("End Date")]
18         public string EndDate { get; set; }
19
20         [JsonProperty("Bike #")]
21         public string Bike { get; set; }
22
23         [JsonProperty("Subscriber Type")]
24         public string SubscriberType { get; set; }
25
26         [JsonProperty("Zip Code")]
27         public string ZipCode { get; set; }
28
29         [JsonProperty("Process Date")]
30         public DateTime ProcessDate { get; set; }
31     }
32 }
```

### 4- Read the data line by line:

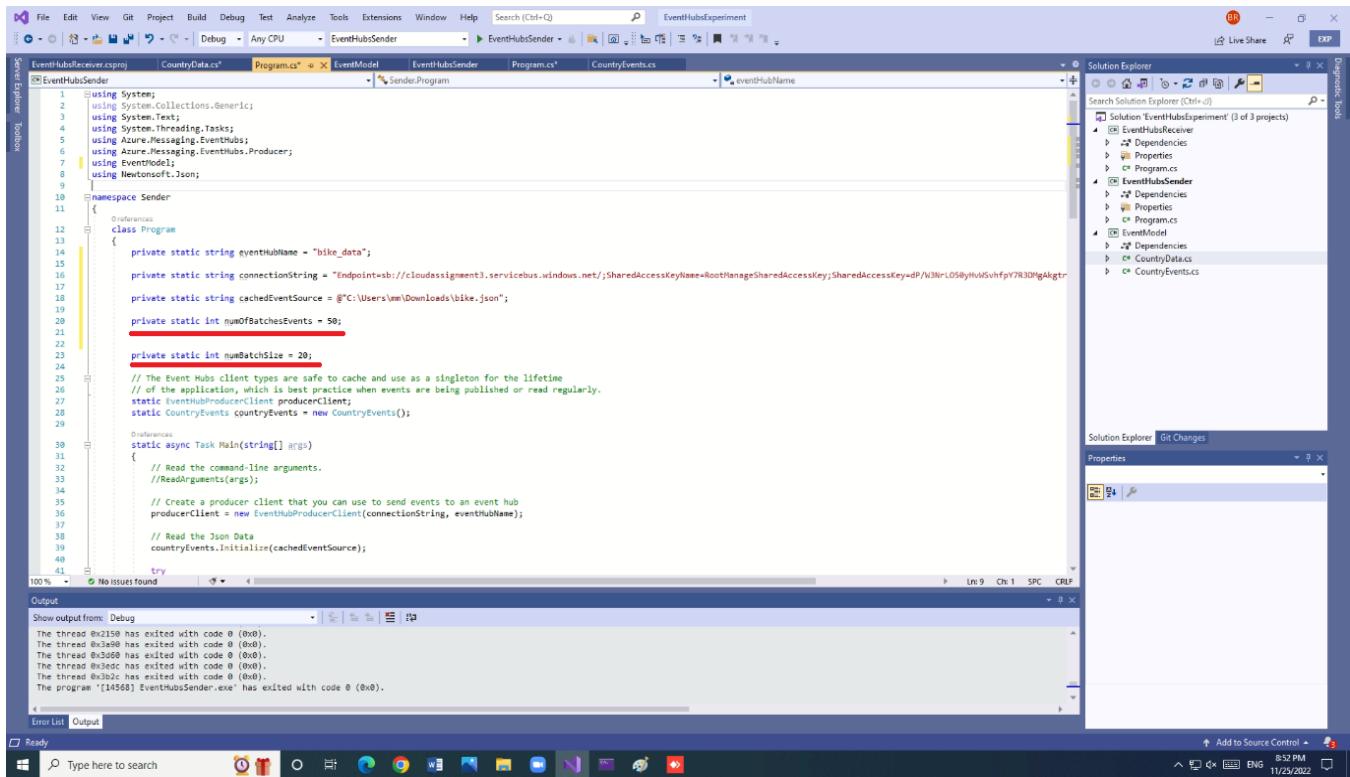


```
File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Full Screen
EventHubsReceiver.csproj | CountryData.cs* | Program.cs* | EventModel | EventHubSender | Program.cs* | CountryEvents.cs
EventModel
1  using Newtonsoft.Json;
2  using System;
3  using System.Collections.Generic;
4  using System.IO;
5  using System.Linq;
6  using System.Text;
7  using System.Threading.Tasks;
8
9  namespace EventModel
10 {
11     public class CountryEvents
12     {
13         private int marker = 0;
14         private int endMarker = 0;
15
16         private List<CountryData> countryDataCollection = new List<CountryData>();
17
18         public string FileName
19         {
20             get;
21             set;
22         }
23
24         public void Initialize(string file)
25         {
26             FileName = file;
27             marker = 0;
28             countryDataCollection.Clear();
29
30             if (!File.Exists(FileName))
31             {
32                 throw new Exception(String.Format("File {0} does not exist", FileName));
33             }
34
35             StreamReader reader = new StreamReader(FileName);
36             string line;
37
38             while ((line = reader.ReadLine()) != null){
39                 var countryData = JsonConvert.DeserializeObject<CountryData>(line);
40                 countryDataCollection.Add(countryData);
41                 endMarker += 1;
42             }
43     }
44 }
```

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

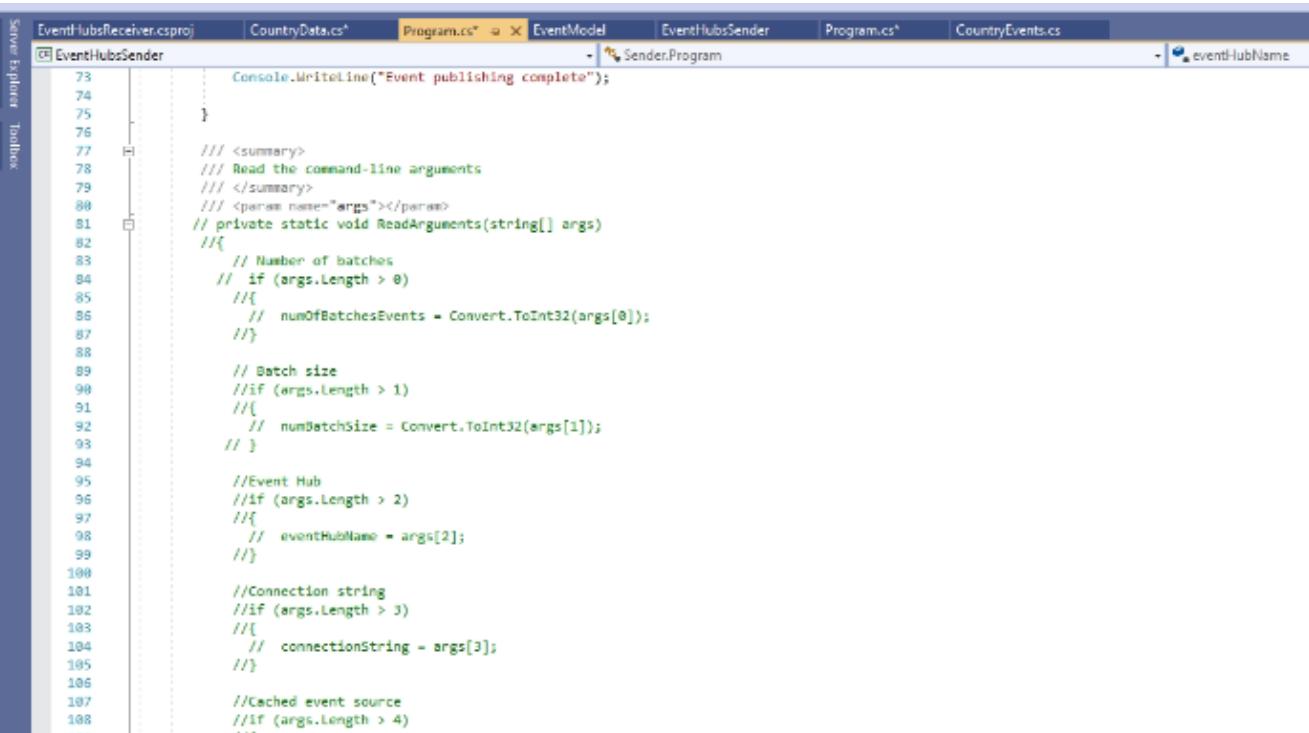


## 6- Set the number of BatchesEvents to 50, and numBatchSize to 20:



```
1  using System;
2  using System.Collections.Generic;
3  using System.Text;
4  using System.Threading.Tasks;
5  using Azure.Messaging.EventHubs;
6  using Azure.Messaging.EventHubs.Producer;
7  using EventModel;
8  using Newtonsoft.Json;
9
10 namespace Sender
11 {
12     class Program
13     {
14         private static string eventHubName = "bikes_data";
15
16         private static string connectionString = "Endpoint=sb://cloudassignment3.servicebus.windows.net/;SharedAccessKeyName=RootManageSharedAccessKey;SharedAccessKey=dP/N3NrL050yHv5vhfpY7R30MgAkgr";
17
18         private static string cachedEventSource = @"C:\Users\amn\Downloads\bike.json";
19
20         private static int numOfBatchesEvents = 50;
21
22
23         private static int numBatchSize = 20;
24
25         // The Event Hubs client types are safe to cache and use as a singleton for the lifetime
26         // of the application, which is best practice when events are being published or read regularly.
27         static EventHubProducerClient producerClient;
28         static CountryEvents countryEvents = new CountryEvents();
29
30         static void Main(string[] args)
31         {
32             // Read the command-line arguments.
33             //ReadArguments(args);
34
35             // Create a producer client that you can use to send events to an event hub
36             producerClient = new EventHubProducerClient(connectionString, eventHubName);
37
38             // Read the Json Data
39             countryEvents.Initialize(cachedEventSource);
40
41             try
42             {
43             }
44             catch
45             {
46             }
47         }
48     }
49 }
```

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



```
73     Console.WriteLine("Event publishing complete");
74 }
75 }
76 }
77 }
78 }
79 }
80 }
81 }
82 }
83 }
84 }
85 }
86 }
87 }
88 }
89 }
90 }
91 }
92 }
93 }
94 }
95 }
96 }
97 }
98 }
99 }
100 }
101 }
102 }
103 }
104 }
105 }
106 }
107 }
108 }
109 }
110 }
111 }
112 }
```

#### 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:

```
43
44     public List<CountryData> GetBatch(int size = 50)
45     {
46         var list = new List<CountryData>();
47         int end = ((marker + size) > endMarker) ? endMarker : marker + 50;
48
49         for (var i = marker; i < end; i++)
50         {
51             list.Add(countryDataCollection[i]);
52         }
53         return list;
54     }
55 }
56
57 }
```

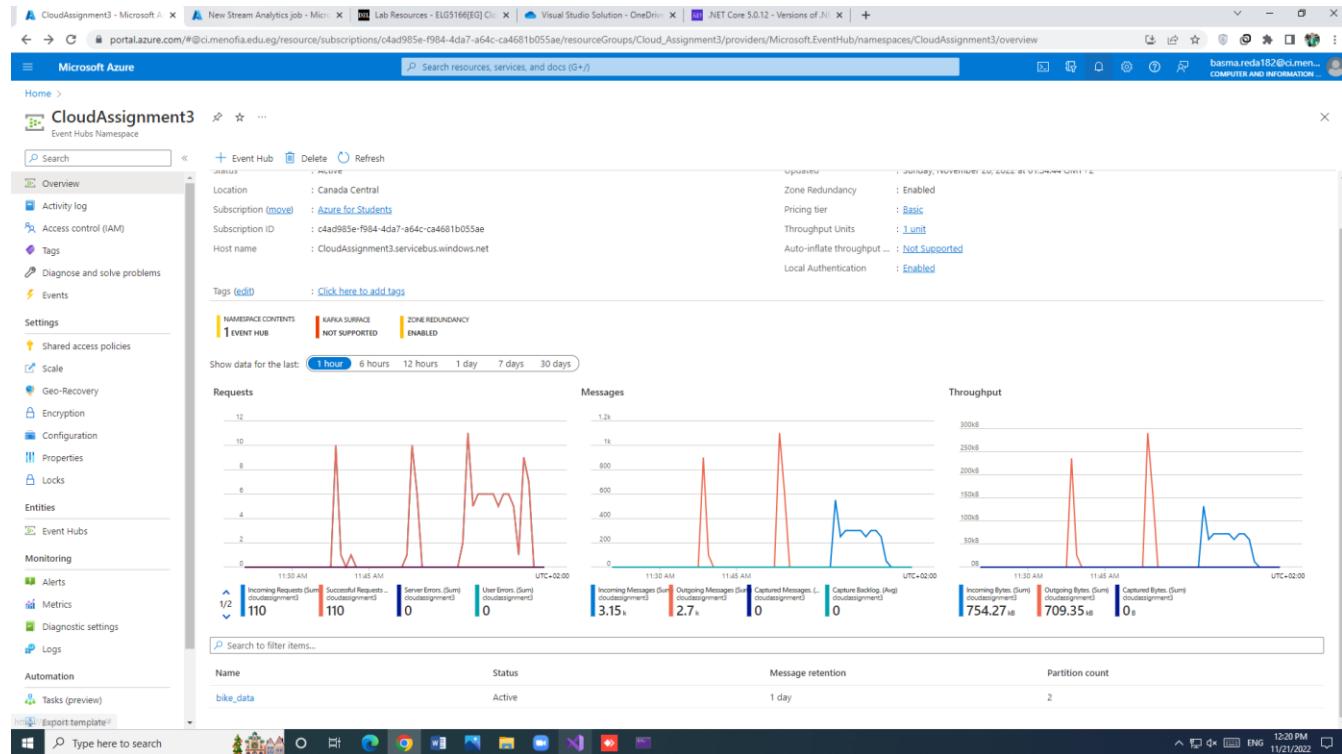
## 10- Finally, we can send the batches to Azure:

## **The batches published successfully:**

## Start the receiver to receive the data:

```
Microsoft Visual Studio Debug Console
,"Bike #": "383", "Subscriber Type": "Subscriber", "Zip Code": "94104", "ProcessDate": "2022-11-21T10:59:19.2071754+02:00"}  
    Received event: {"Trip ID": "913459", "Duration": "1036", "Start Date": "8/31/2015 23:11", "End Date": "8/31/2015 23:28"}  
, "Bike #": "35", "Subscriber Type": "Subscriber", "Zip Code": "95032", "ProcessDate": "2022-11-21T10:59:34.689102+02:00"}  
    Received event: {"Trip ID": "913440", "Duration": "281", "Start Date": "8/31/2015 21:31", "End Date": "8/31/2015 21:36"}  
, "Bike #": "621", "Subscriber Type": "Subscriber", "Zip Code": "94107", "ProcessDate": "2022-11-21T10:59:19.2071806+02:00"}  
    Received event: {"Trip ID": "913455", "Duration": "307", "Start Date": "8/31/2015 23:13", "End Date": "8/31/2015 23:18"}  
, "Bike #": "468", "Subscriber Type": "Subscriber", "Zip Code": "94107", "ProcessDate": "2022-11-21T10:59:34.689107+02:00"}  
    Received event: {"Trip ID": "913435", "Duration": "424", "Start Date": "8/31/2015 21:25", "End Date": "8/31/2015 21:33"}  
, "Bike #": "602", "Subscriber Type": "Subscriber", "Zip Code": "94401", "ProcessDate": "2022-11-21T10:59:19.207186+02:00"}  
    Received event: {"Trip ID": "913454", "Duration": "409", "Start Date": "8/31/2015 23:10", "End Date": "8/31/2015 23:17"}  
, "Bike #": "68", "Subscriber Type": "Subscriber", "Zip Code": "95113", "ProcessDate": "2022-11-21T10:59:34.6891103+02:00"}  
    Received event: {"Trip ID": "913434", "Duration": "283", "Start Date": "8/31/2015 21:19", "End Date": "8/31/2015 21:24"}  
, "Bike #": "521", "Subscriber Type": "Subscriber", "Zip Code": "94107", "ProcessDate": "2022-11-21T10:59:19.2072074+02:00"}  
    Received event: {"Trip ID": "913453", "Duration": "789", "Start Date": "8/31/2015 23:09", "End Date": "8/31/2015 23:22"}  
, "Bike #": "487", "Subscriber Type": "Customer", "Zip Code": "90609", "ProcessDate": "2022-11-21T10:59:34.6891137+02:00"}  
    Received event: {"Trip ID": "913433", "Duration": "145", "Start Date": "8/31/2015 21:17", "End Date": "8/31/2015 21:20"}  
, "Bike #": "75", "Subscriber Type": "Customer", "Zip Code": "69007", "ProcessDate": "2022-11-21T10:59:19.2072127+02:00"}  
    Received event: {"Trip ID": "913452", "Duration": "293", "Start Date": "8/31/2015 23:07", "End Date": "8/31/2015 23:12"}  
, "Bike #": "538", "Subscriber Type": "Subscriber", "Zip Code": "94118", "ProcessDate": "2022-11-21T10:59:34.6891169+02:00"}  
    Received event: {"Trip ID": "913432", "Duration": "703", "Start Date": "8/31/2015 21:16", "End Date": "8/31/2015 21:28"}  
, "Bike #": "426", "Subscriber Type": "Subscriber", "Zip Code": "95032", "ProcessDate": "2022-11-21T10:59:19.2072186+02:00"}  
    Received event: {"Trip ID": "913451", "Duration": "896", "Start Date": "8/31/2015 23:07", "End Date": "8/31/2015 23:22"}  
, "Bike #": "363", "Subscriber Type": "Customer", "Zip Code": "92562", "ProcessDate": "2022-11-21T10:59:34.6891205+02:00"}  
  
C:\Users\mm\Downloads\Event Hub and Azure Stream Analytics\Visual Studio Solution\EventHubsReceiver\bin\Debug\net5.0\Eve  
ntHubsReceiver.exe (process 14176) exited with code 0.  
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso  
le when debugging stops.  
Press any key to close this window . . .
```

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



Microsoft Azure

Home > Resource groups > Cloud\_Assignment3 > CloudAssignment3 | Event Hubs > bike\_data (CloudAssignment3/bike\_data) | Process data >

Query ...

Event Hub instance

Create Stream Analytics job | Query language docs | Share feedback

Azure Stream Analytics lets you perform real-time analytics. Start by testing your query, then deploy your query as Azure Stream Analytics job. Learn more →

Inputs (1) Outputs (1) Functions (0)

Test query

```

1 SELECT
2 *
3 INTO
4 [OutputAlias]
5 FROM
6 [bikedata]

```

Input preview Test results

Showing sample events from 'bikedata'.

View in JSON Table Raw Refresh Download sample data

Trip ID	Duration	Start Date	End Date	Bike #	Subscriber Type	Zip Code	ProcessDate	EventProcessedUtcTime	PartitionId	EventEnqueuedUtcTime
'913452'	'293'	'8/31/2015 23:07'	'8/31/2015 23:12'	'538'	'Subscriber'	'94118'	'2022-11-25T19:45:05...	'2022-11-25T19:48:31...	1	'2022-11-25T19:45:05...
'913455'	'307'	'8/31/2015 23:13'	'8/31/2015 23:18'	'468'	'Subscriber'	'94107'	'2022-11-25T19:45:05...	'2022-11-25T19:48:31...	1	'2022-11-25T19:45:05...
'913453'	'789'	'8/31/2015 23:09'	'8/31/2015 23:22'	'487'	'Customer'	'9069'	'2022-11-25T19:45:05...	'2022-11-25T19:48:31...	1	'2022-11-25T19:45:05...
'913459'	'1036'	'8/31/2015 23:11'	'8/31/2015 23:28'	'35'	'Subscriber'	'95032'	'2022-11-25T19:45:05...	'2022-11-25T19:48:31...	1	'2022-11-25T19:45:05...
'913451'	'896'	'8/31/2015 23:07'	'8/31/2015 23:22'	'363'	'Customer'	'92562'	'2022-11-25T19:45:05...	'2022-11-25T19:48:31...	1	'2022-11-25T19:45:05...
'913450'	'255'	'8/31/2015 22:16'	'8/31/2015 22:20'	'470'	'Subscriber'	'94111'	'2022-11-25T19:45:05...	'2022-11-25T19:48:31...	1	'2022-11-25T19:45:05...
'913449'	'126'	'8/31/2015 22:12'	'8/31/2015 22:15'	'439'	'Subscriber'	'94130'	'2022-11-25T19:45:05...	'2022-11-25T19:48:31...	1	'2022-11-25T19:45:05...
'913448'	'932'	'8/31/2015 21:57'	'8/31/2015 22:12'	'472'	'Subscriber'	'94702'	'2022-11-25T19:45:05...	'2022-11-25T19:48:31...	1	'2022-11-25T19:45:05...
'913454'	'409'	'8/31/2015 23:10'	'8/31/2015 23:17'	'68'	'Subscriber'	'95113'	'2022-11-25T19:45:05...	'2022-11-25T19:48:31...	1	'2022-11-25T19:45:05...

Ln 6, Col 15

Success

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.

Microsoft Azure

Home > Resource groups > Cloud\_Assignment3 > CloudAssignment3 | Event Hubs > bike\_data (CloudAssignment3/bike\_data) | Process data >

Query ...

Event Hub instance

Create Stream Analytics job | Query language docs | Share feedback

Azure Stream Analytics lets you perform real-time analytics. Start by testing your query, then deploy your query as Azure Stream Analytics job. Learn more →

Inputs (1) Outputs (1) Functions (0)

Test query

```

1 SELECT
2 *
3 INTO
4 [OutputAlias]
5 FROM
6 [bikedata]

```

Input preview Test results

Showing sample events from 'bikedata'.

View in JSON Table Raw Refresh Download sample data

Trip ID	Duration	Start Date	End Date	Bike #	Subscriber Type	Zip Code	ProcessDate
'913455'	'424'	'8/31/2015 21:25'	'8/31/2015 21:33'	'602'	'Subscriber'	'94401'	'2022-11-25T19:45:05...
'913455'	'307'	'8/31/2015 23:13'	'8/31/2015 23:18'	'468'	'Subscriber'	'94107'	'2022-11-25T19:45:05...
'913441'	'387'	'8/31/2015 21:39'	'8/31/2015 21:46'	'383'	'Subscriber'	'94104'	'2022-11-25T19:45:05...
'913442'	'633'	'8/31/2015 21:44'	'8/31/2015 21:54'	'531'	'Subscriber'	'94107'	'2022-11-25T19:45:05...
'913443'	'691'	'8/31/2015 21:49'	'8/31/2015 22:01'	'434'	'Subscriber'	'94109'	'2022-11-25T19:45:05...
'913448'	'932'	'8/31/2015 21:57'	'8/31/2015 22:12'	'472'	'Subscriber'	'94702'	'2022-11-25T19:45:05...
'913449'	'126'	'8/31/2015 22:12'	'8/31/2015 22:15'	'439'	'Subscriber'	'94130'	'2022-11-25T19:45:05...
'913450'	'255'	'8/31/2015 22:16'	'8/31/2015 22:20'	'470'	'Subscriber'	'94111'	'2022-11-25T19:45:05...
'913451'	'896'	'8/31/2015 23:07'	'8/31/2015 23:22'	'363'	'Customer'	'92562'	'2022-11-25T19:45:05...

New Stream Analytics job

This will create a new Stream Analytics job. You will be charged according to Azure Stream Analytics billing model. → Learn more.

Job name \* StreamingJobAssignment3

Subscription \* Azure for Students

Resource group \* Cloud\_Assignment3

Location \* Canada Central

Event Hub policy name \* Create new Use existing StreamingJobAssignment3\_policy

Event Hub consumer group \* Create new Use existing \$Default

Create

## To add an output:

Microsoft Azure

Home > Resource groups > Cloud\_Assignment3 > CloudAssignment3 | Event Hubs > bike\_data (CloudAssignment3/bike\_data) | Process data >

StreamingJobAssignment3

Stream Analytics job

Created

Resource group (move) : Cloud\_Assignment3

Location : Canada Central

Status : Created

Subscription (move) : Acure for Students

Subscription ID : cdad985e-f984-4da7-a64c-ca4681b055ae

Tags (edit) : Click here to add tags

Get started Properties Monitoring Tutorials

Build an end-to-end serverless streaming pipeline with just a few clicks

Azure Stream Analytics is a fully managed, real-time stream processing service designed to help you tackle scenarios such as streaming ETL to ADLS Gen2 or Synapse SQL, real-time apps with Cosmos DB or SQL DB, live dashboarding with Power BI, or real-time alerting with Azure Functions. [Learn more](#)

**Ingest data**  
Stream Analytics jobs connect to one or more data inputs. Each input defines a connection to an existing data source. [Add input](#)

**Analyze data**  
Stream Analytics jobs uses Stream Analytics Query Language (SAQL) to transform or analyze your real-time data. [Write query](#)

**Output data**  
Stream Analytics jobs connects to one or more data outputs. There are several output types to which you can send transformed data. [Add output](#)

**Enable logging**  
Turning on diagnostic settings to Log Analytics will allow you to easily troubleshoot any errors your job may encounter. [Configure](#)

blob storage/ADLS Gen2

New output

Output alias \*

Provide Blob storage/ADLS Gen2 settings manually  
 Select Blob storage/ADLS Gen2 from your subscriptions

Subscription

Storage account \*

Container \*  Create new  Use existing

Authentication mode

The Storage Blob Data Contributor role will be granted to the Managed Identity for this Stream Analytics job when you click Save. If grant fails follow the manual grant steps [here](#).

Event serialization format \*

Format

Encoding

**Save**

**The query is:**

```
1 SELECT
2     System.Timestamp() AS WindowEnd, SUM([Bike #]) AS NumberofBikes, SUM([Duration]) AS TotalDuration
3 INTO
4     [StreamingJobAssignment3]
5 FROM
6     [bikedata]
7 GROUP BY System.Timestamp()
```

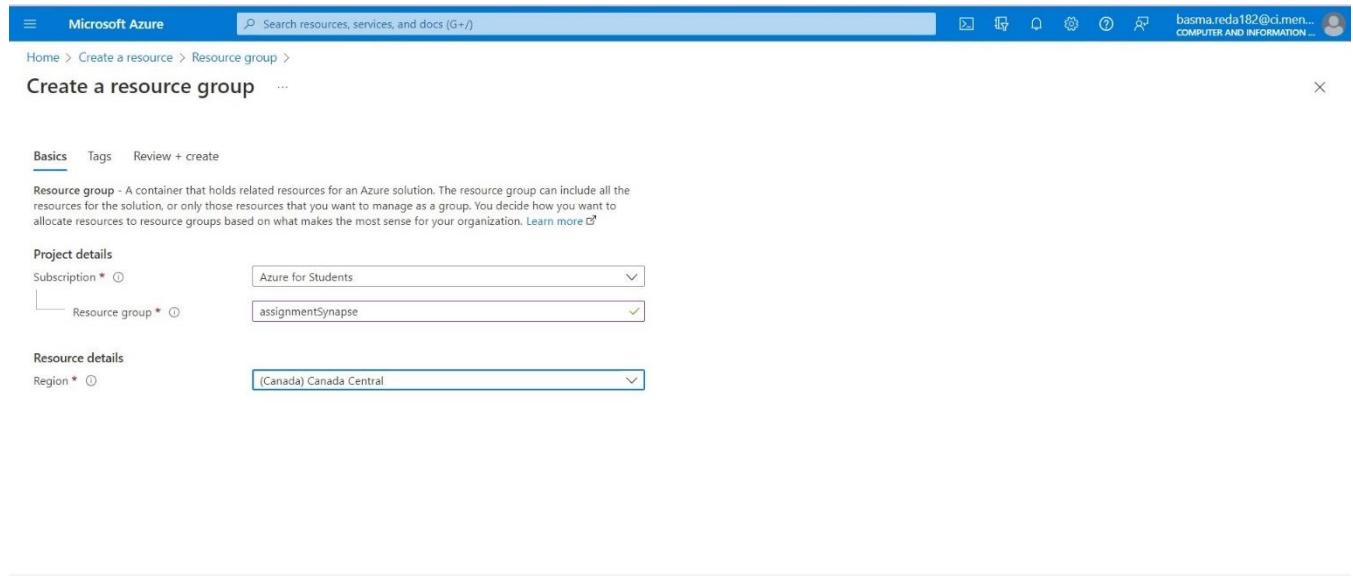
**The output:**

WindowEnd	NumberofBikes	TotalDuration
2022-11-25T20:08:24.905085Z	288	765
2022-11-25T20:08:24.9207080Z	20518	27082

**Save the query and download the results.**

## Part 2: Azure Synapse Analytics:

### 1- Create a resource group:



Microsoft Azure Search resources, services, and docs (G+) Home > Create a resource > Resource group > Create a resource group basma.reda182@ci.men... COMPUTER AND INFORMATION ...

**Create a resource group** ... X

**Basics** Tags Review + create

**Resource group** - A container that holds related resources for an Azure solution. The resource group can include all the resources for the solution, or only those resources that you want to manage as a group. You decide how you want to allocate resources to resource groups based on what makes the most sense for your organization. [Learn more](#)

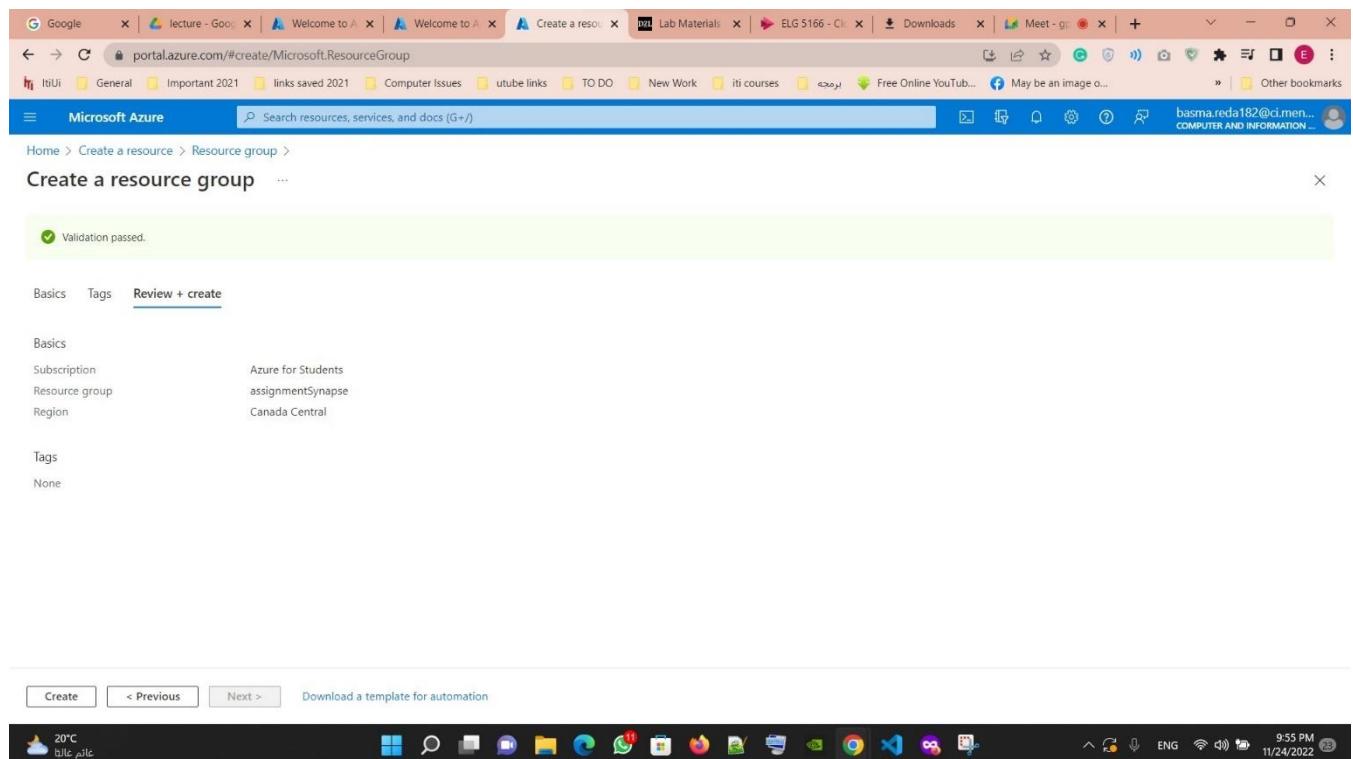
**Project details**

Subscription \* Subscription Azure for Students

Resource group \* Resource group assignmentSynapse

**Resource details**

Region \* Region (Canada) Canada Central



Microsoft Azure Search resources, services, and docs (G+) Home > Create a resource > Resource group > Create a resource group basma.reda182@ci.men... COMPUTER AND INFORMATION ...

**Create a resource group** ... X

Validation passed.

**Basics** Tags Review + create

**Basics**

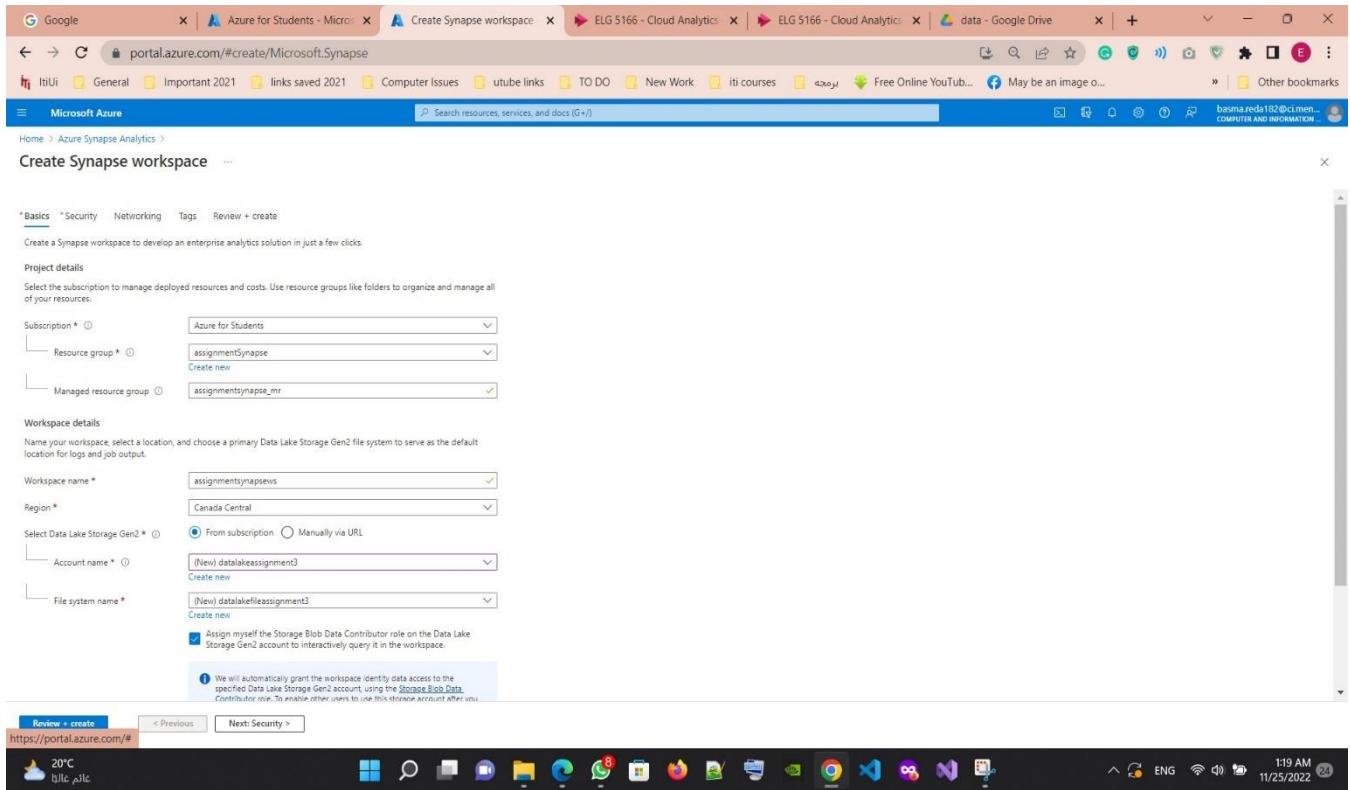
Subscription	Azure for Students
Resource group	assignmentSynapse
Region	Canada Central

**Tags**

None

Create < Previous Next > Download a template for automation

## 2- Create a Synapse Workspace:



Basics \*Security Networking Tags Review + create

Create a Synapse workspace to develop an enterprise analytics solution in just a few clicks.

**Project details**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all of your resources.

Subscription \*  Resource group \*  Create new Managed resource group \*

**Workspace details**

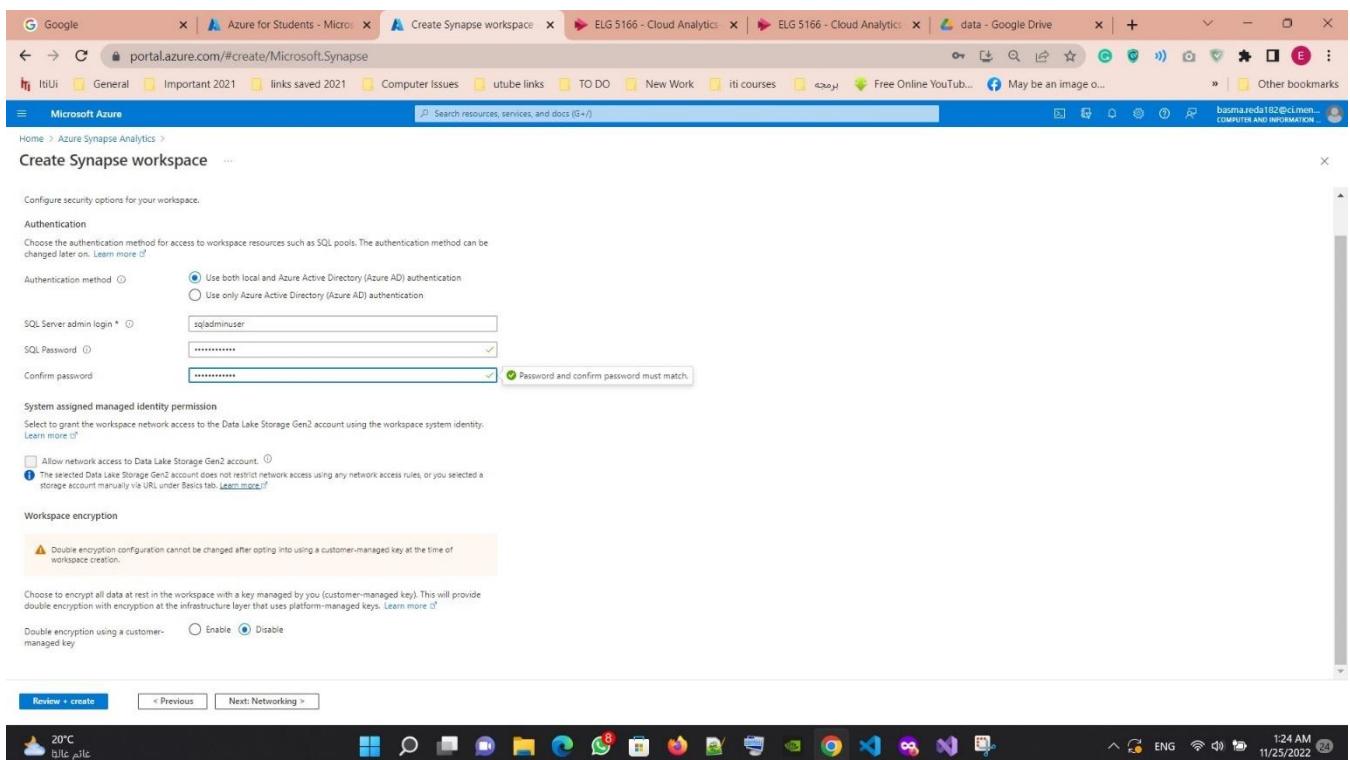
Name your workspace, select a location, and choose a primary Data Lake Storage Gen2 file system to serve as the default location for logs and job output.

Workspace name \*  Region \*  Select Data Lake Storage Gen2 \*   Account name \*  Create new File system name \*  Create new  Assign myself the Storage Blob Data Contributor role on the Data Lake Storage Gen2 account to interactively query it in the workspace.

We will automatically grant the workspace identity data access to the specified Data Lake Storage Gen2 account using the Storage Blob Data Contributor role. To enable other users to use this storage account after you.

**Review + create** < Previous Next: Security > https://portal.azure.com/#

## 3- Review the security of the workspace:



Configure security options for your workspace.

**Authentication**

Choose the authentication method for access to workspace resources such as SQL pools. The authentication method can be changed later on. [Learn more](#)  Use both local and Azure Active Directory (Azure AD) authentication  Use only Azure Active Directory (Azure AD) authentication

SQL Server admin login \*  SQL Password \*  Confirm password   Password and confirm password must match.

**System assigned managed identity permission**

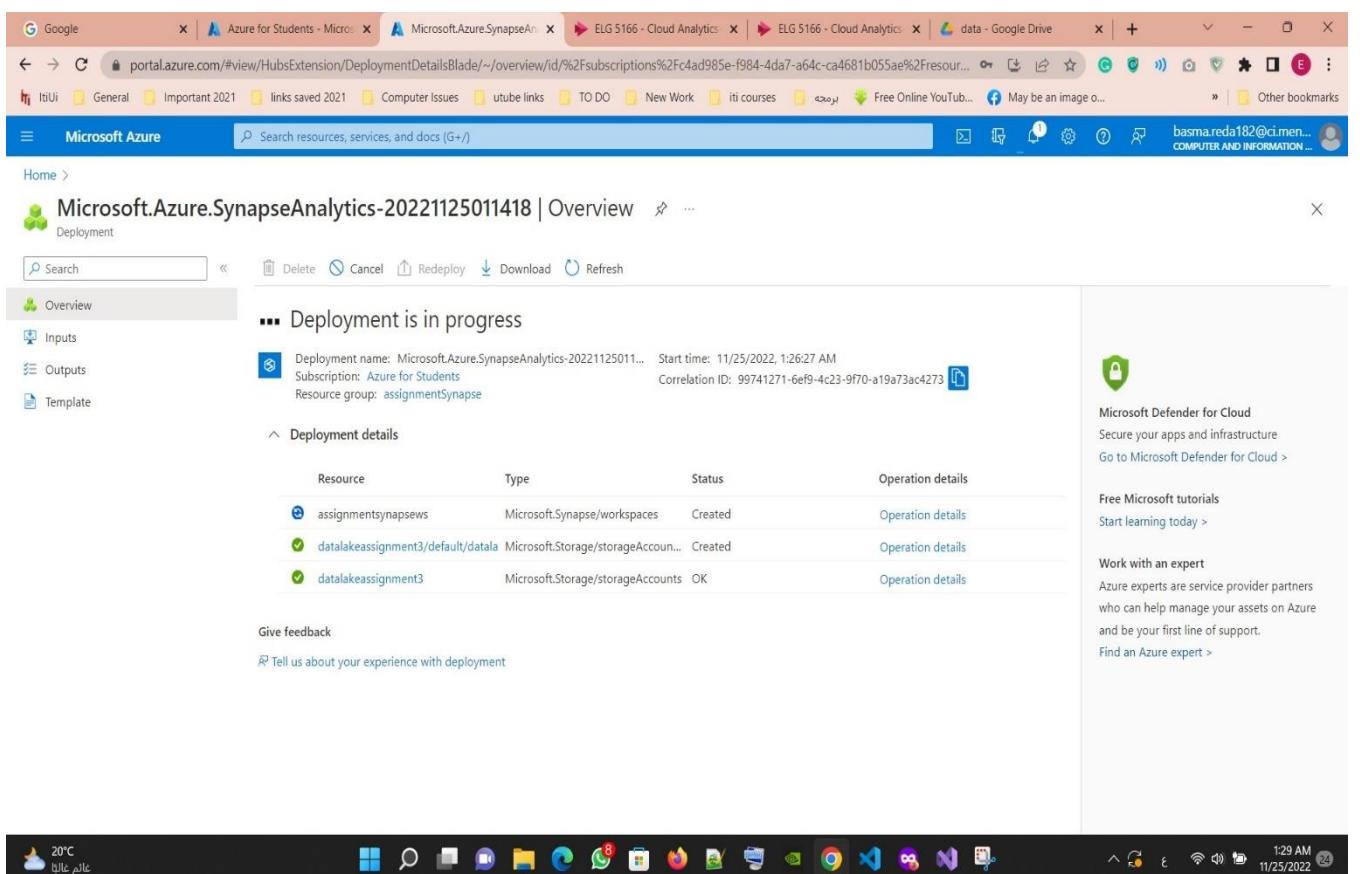
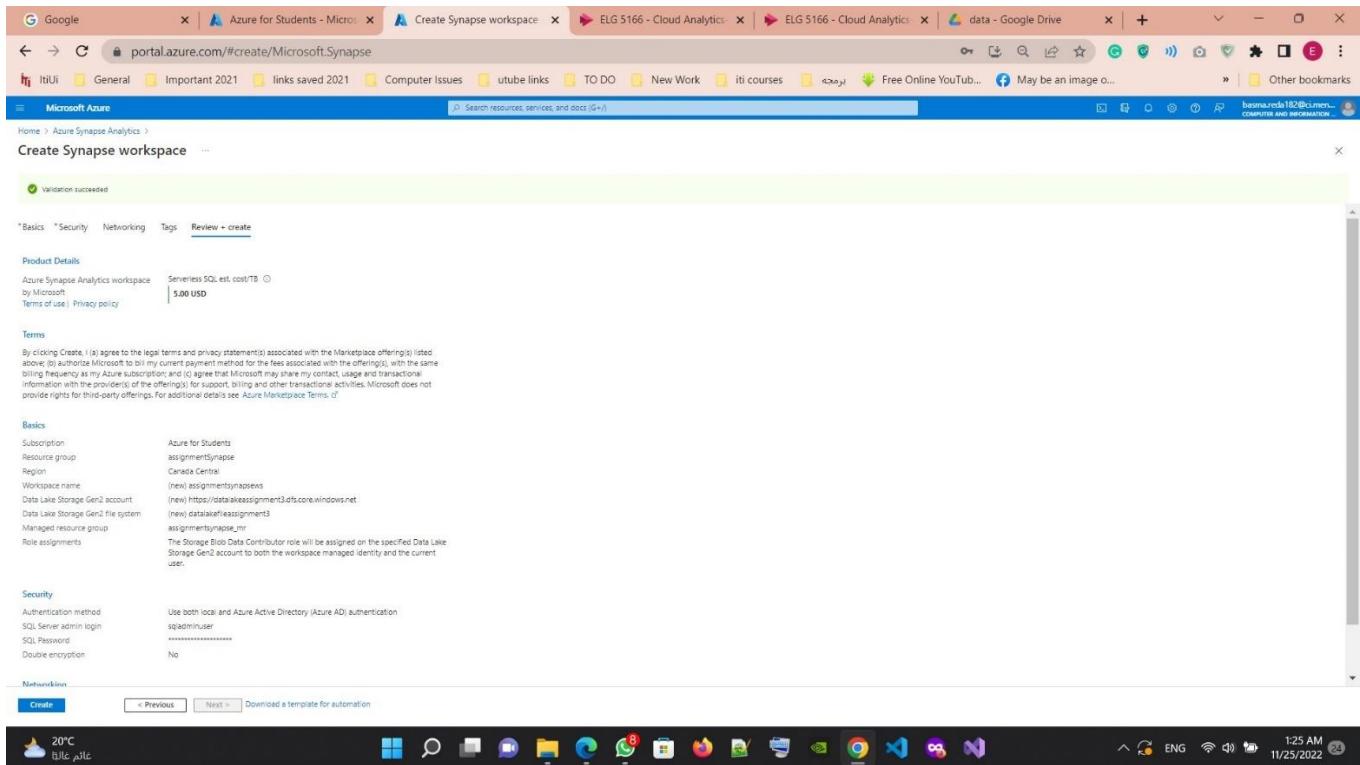
Select to grant the workspace network access to the Data Lake Storage Gen2 account using the workspace system identity. [Learn more](#)  Allow network access to Data Lake Storage Gen2 account. The selected Data Lake Storage Gen2 account does not restrict network access using any network access rules, or you selected a storage account manually via URL under Basics tab. [Learn more](#)

**Workspace encryption**

⚠ Double encryption configuration cannot be changed after opting into using a customer-managed key at the time of workspace creation.

Choose to encrypt all data at rest in the workspace with a key managed by you (customer-managed key). This will provide double encryption with encryption at the infrastructure layer that uses platform-managed keys. [Learn more](#)  Enable  Disable

**Review + create** < Previous Next: Networking > 20°C b1bc p1c 11:24 AM 11/25/2022



Microsoft.Azure.SynapseAnalytics-20221125011418 | Overview

Deployment

Search

Deployment name: Microsoft.Azure.SynapseAnalytics-20221125011... Start time: 11/25/2022, 1:26:27 AM

Subscription: Azure for Students Correlation ID: 99741271-6ef9-4c23-9f70-a19a73ac4273

Resource group: assignmentSynapse

Deployment details

Next steps

Go to resource group

Give feedback

Tell us about your experience with deployment

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 >

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 >

20°C

1:30 AM 11/25/2022

#### 4- Go to the Synapse that has been created:

assignmentSynapse | Overview

Resource group

Search

Create Manage view Delete resource group Refresh Export to CSV Open query Assign tags Move Delete Export template

Subscription (move) : Azure for Students Deployments : 2 Succeeded

Subscription ID : c4ad985e-f984-4da7-a64c-ca4681b055ae Location : Canada Central

Tags (edit) : Click here to add tags

Resources Recommendations

Filter for any field... Type equals all Location equals all Add filter

Showing 1 to 2 of 2 records. Show hidden types

Name	Type	Location
assignmentsynapse	Synapse workspace	Canada Central
datalakeassignment3	Storage account	Canada Central

< Previous Page 1 of 1 Next >

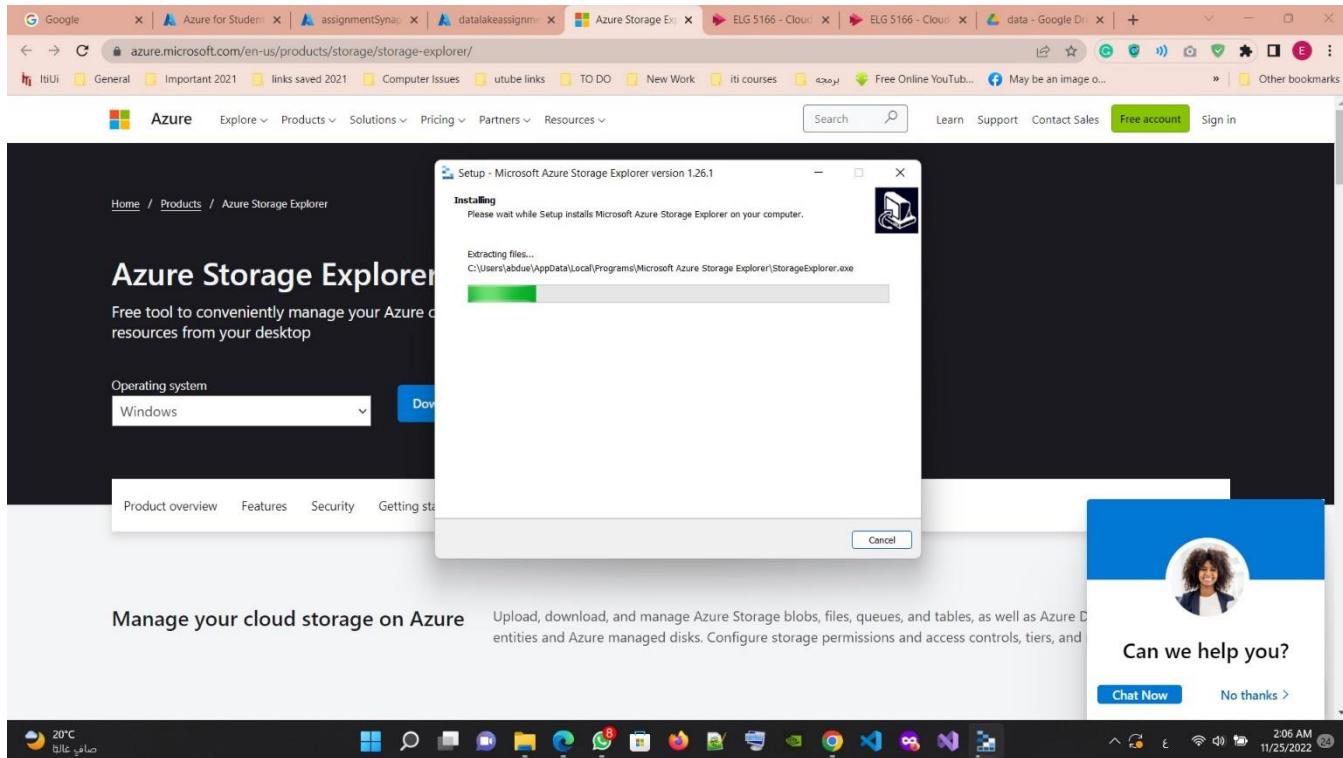
Give feedback

20°C

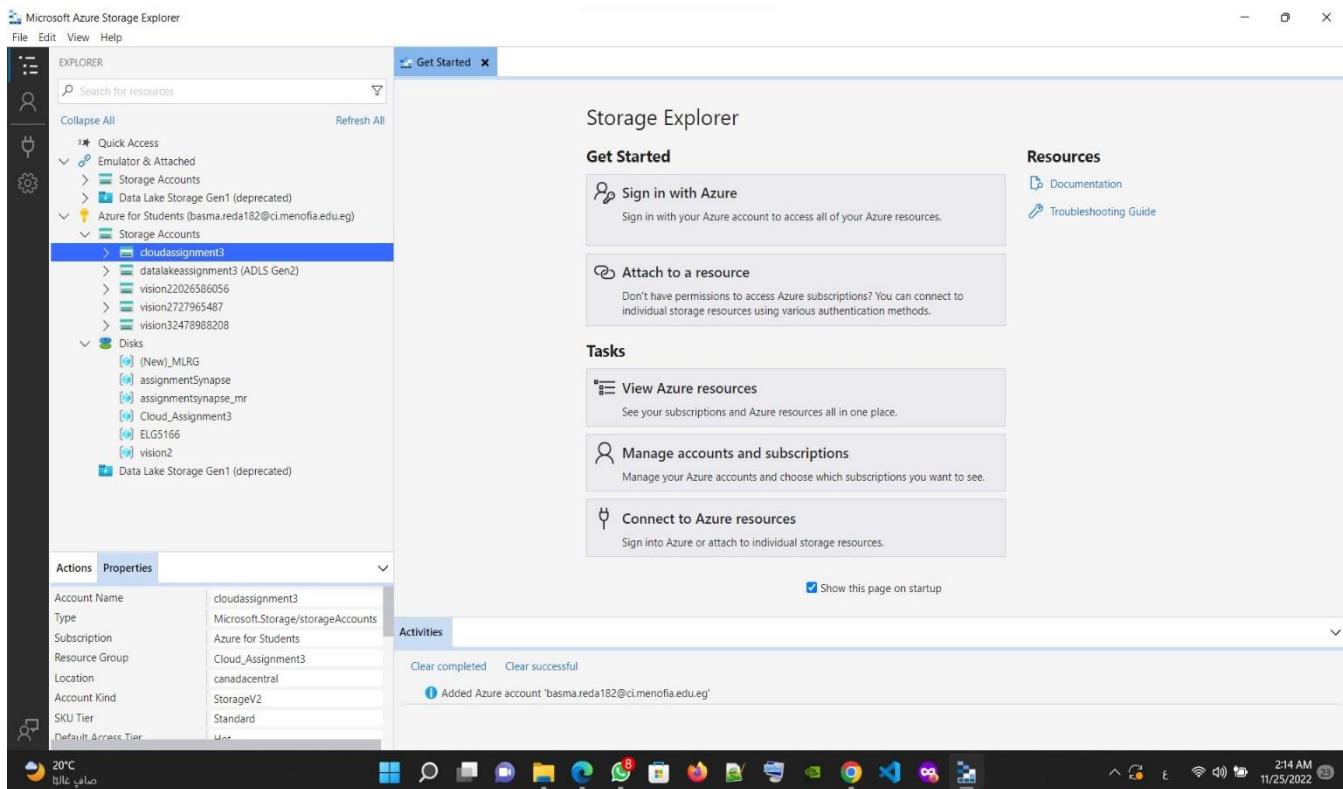
1:32 AM 11/25/2022

## 5- Create a storage account to store the data:

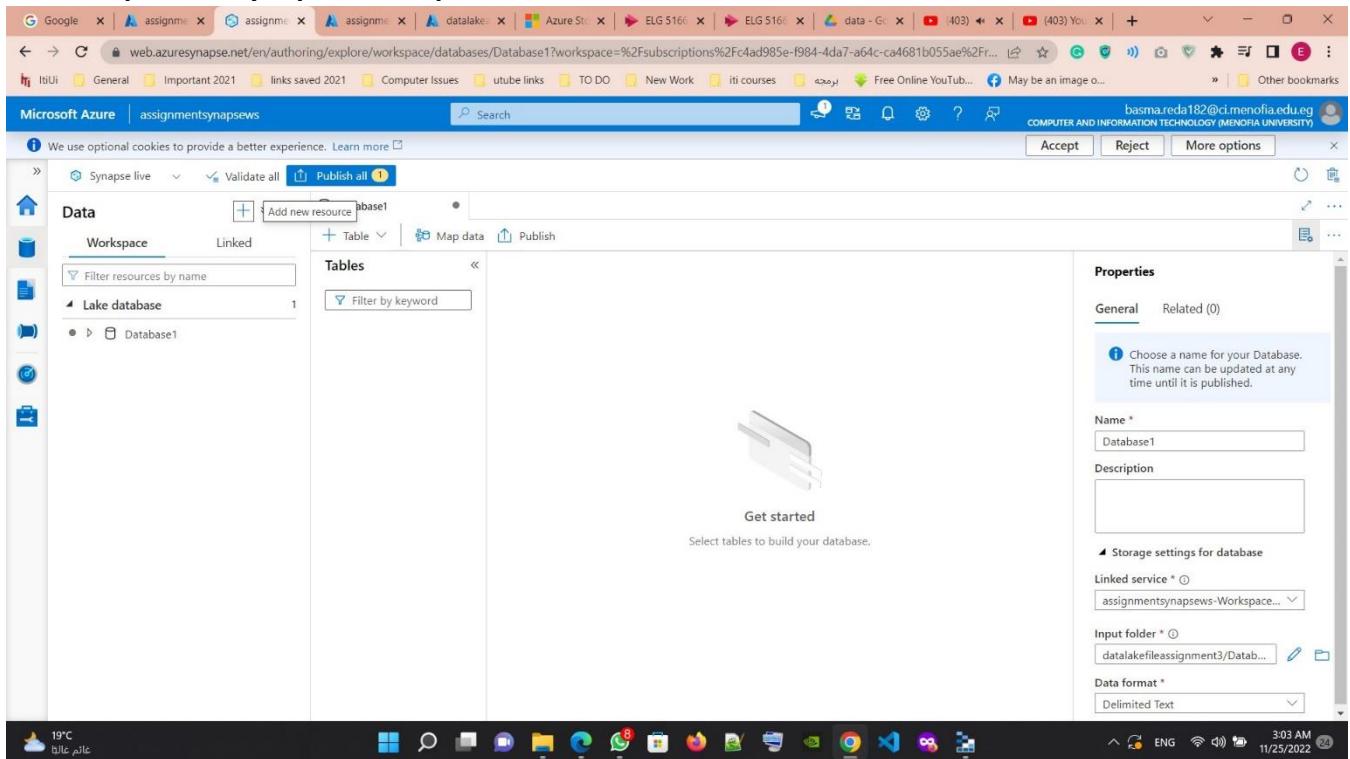
## 6- Install and download the MSI file:



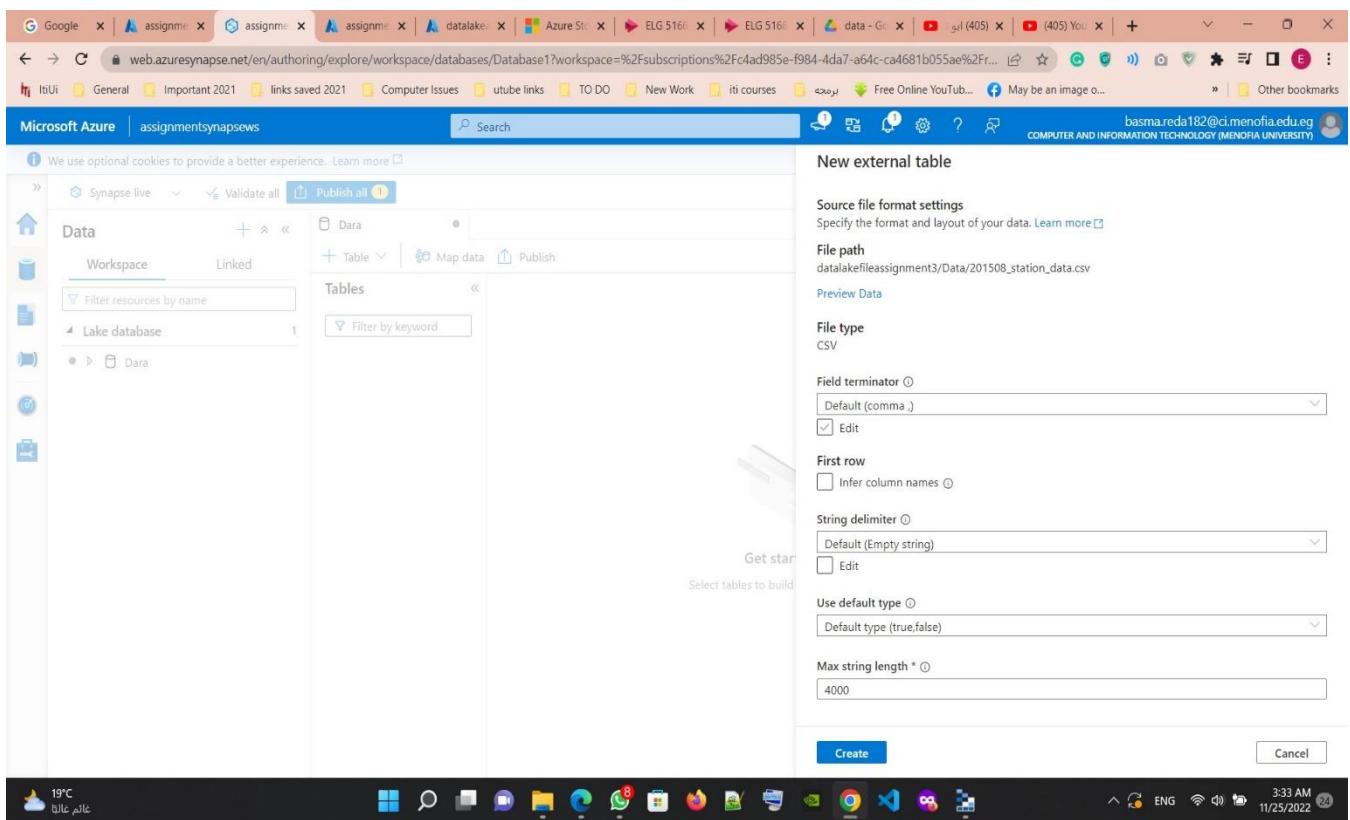
## 7- Get started with the Synapse:



## 8- Open the Synapse Workspace:



## 9- Create a new external table:



A) *Top 20 zip codes for bike up.*

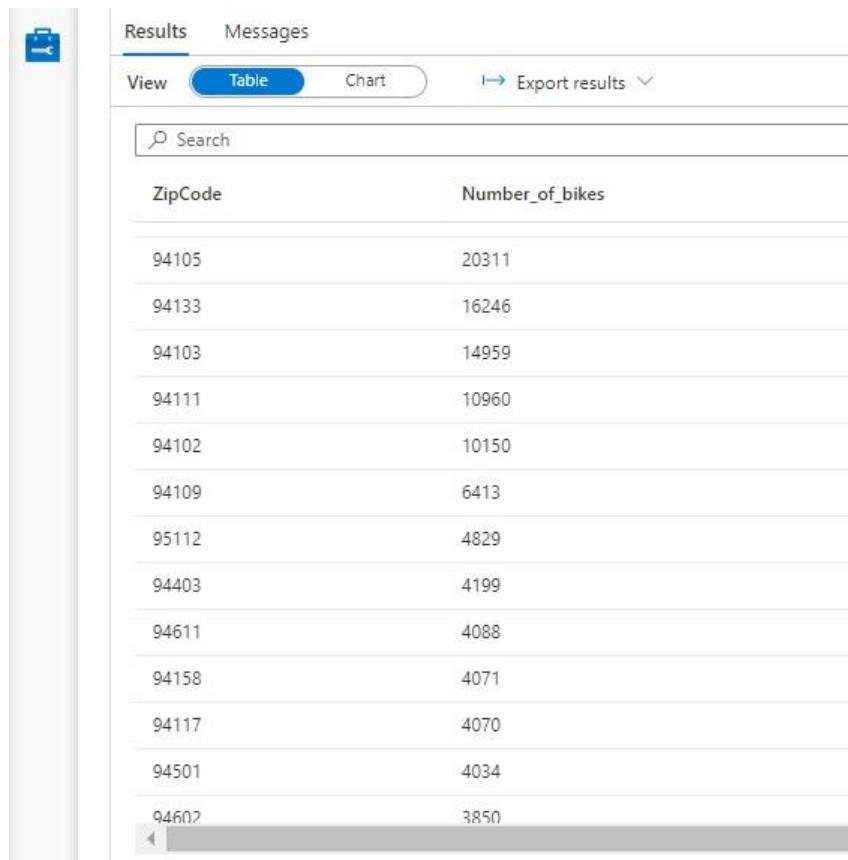
**Set the right format for the data:**

station_data	
Columns	...
station_id (string)	
name (string)	
lat (float)	
long (float)	
dockcount (integer)	
landmark (string)	
installation (date)	
trip_data	
Columns	...
TripID (string)	
Duration (integer)	
StartDate (date)	
StartTerminal (string)	
EndDate (date)	
EndTerminal (string)	
BikeNum (string)	
SubscriberType (string)	
ZipCode (string)	

**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
```

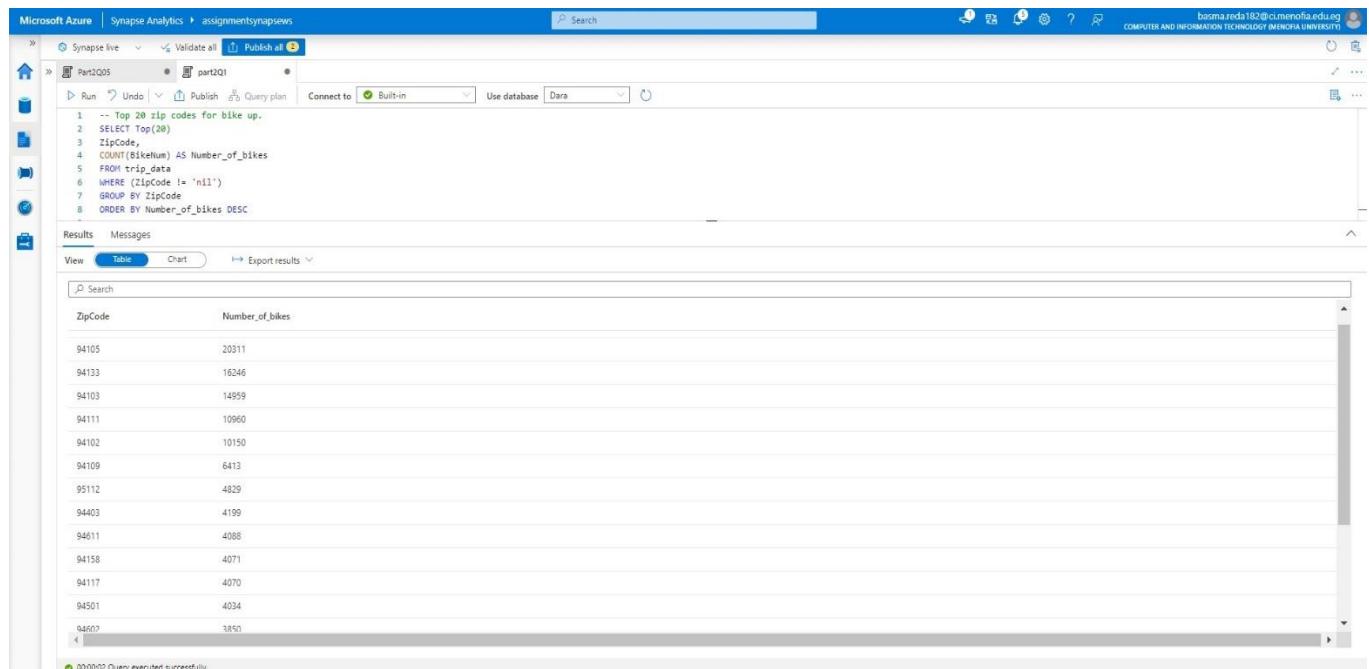
## Output:



The screenshot shows the Azure Synapse Analytics results interface. The top navigation bar includes 'Results' and 'Messages' tabs, with 'Results' being the active tab. Below the tabs are 'View' (set to 'Table'), 'Table' (selected), 'Chart', and 'Export results' buttons. A search bar is present. The main content area displays a table with two columns: 'ZipCode' and 'Number\_of\_bikes'. The data is as follows:

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:



The screenshot shows the Azure Synapse Analytics query editor. The top navigation bar includes 'Synapse live', 'Validate all', 'Publish all', 'Search', and a user profile. The main area shows a query plan with a 'part2Q5' step. Below the plan is a code editor with the following SQL query:

```
1 -- Top 20 zip codes for bike up.
2 SELECT Top(20)
3 ZipCode,
4 COUNT(BikeHelm) AS Number_of_bikes
5 FROM trip_data
6 WHERE (ZipCode != 'n/a')
7 GROUP BY ZipCode
8 ORDER BY Number_of_bikes DESC
```

The results tab shows the same table as the previous screenshot, with the data:

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

At the bottom, a message indicates '000002 Query executed successfully.'

## 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:**

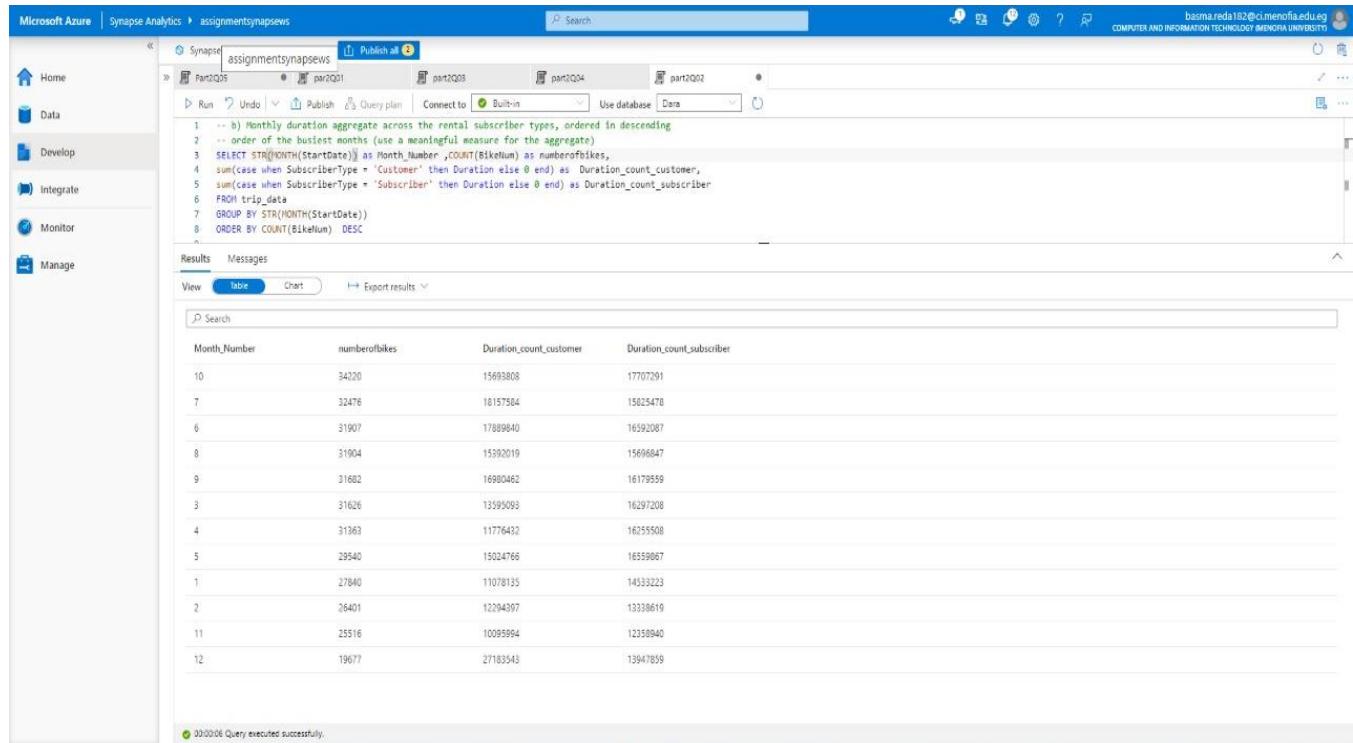
```
-- b) Monthly duration aggregate across the rental subscriber types, ordered in descending
-- order of the busiest months (use a meaningful measure for the aggregate)
SELECT STR(MONTH(StartDate)) as Month_Number ,COUNT(BikeNum) as numberofbikes,
sum(case when SubscriberType = 'Customer' then Duration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



Microsoft Azure | Synapse Analytics > assignmentsynapsnews

Home Data Develop Integrate Monitor Manage

Synapse assignmentsynapsnews Publish all

Run Undo Publish Query plan Connect to Built-in Use database Data

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

2 SELECT STR(MONTH(StartDate)) as Month\_Number ,COUNT(BikeNum) as numberofbikes,

3 sum(case when SubscriberType = 'Customer' then Duration else 0 end) as Duration\_count\_customer,

4 sum(case when SubscriberType = 'Subscriber' then Duration else 0 end) as Duration\_count\_subscriber

5 FROM trip\_data

6 GROUP BY STR(MONTH(StartDate))

7 ORDER BY COUNT(BikeNum) DESC

Results Messages

View Table Chart Export results

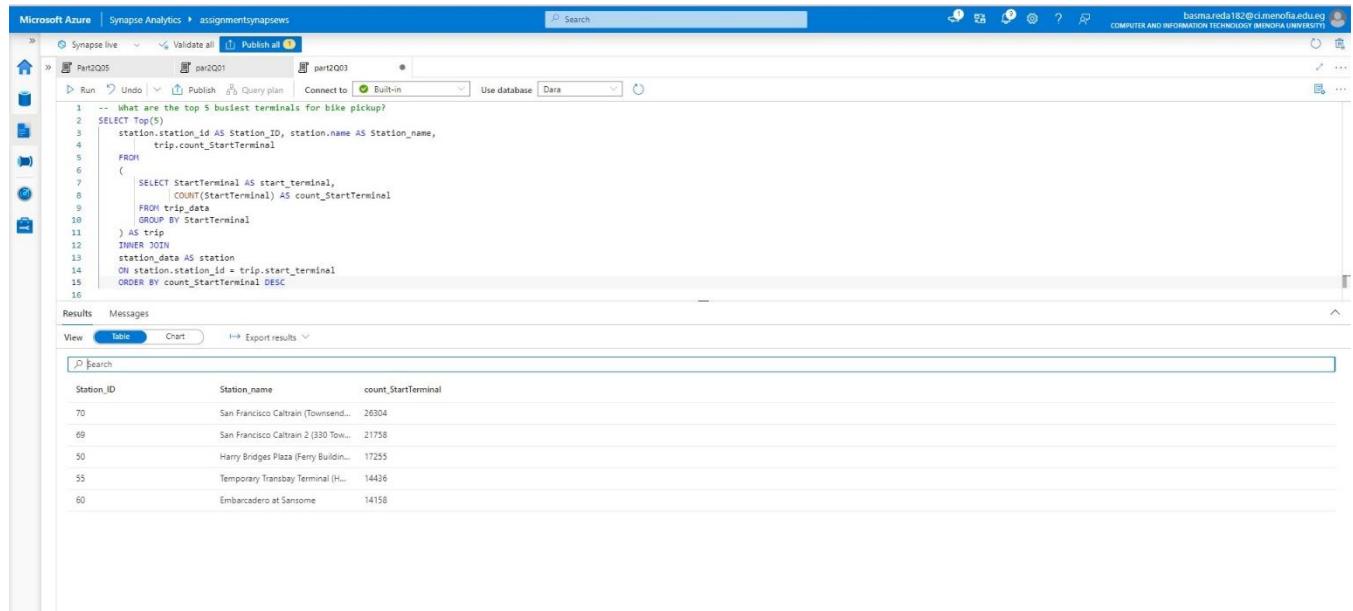
Month_Number	numberofbikes	Duration_count_customer	Duration_count_subscriber
10	34220	15693808	17707291
7	32476	18157504	15025478
6	31907	17889840	16592087
8	31904	1592019	15698647
9	31682	1690462	16179559
3	31626	13595093	16297208
4	31363	11776432	16255908
5	29540	15024766	16559667
1	27840	11078135	14533223
2	26401	12204397	13338619
11	25516	10095994	12350840
12	19077	27183543	13947859

000006 Query executed successfully.

## Save the output in a CSV file.

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

## Screenshot from Azure:



Microsoft Azure | Synapse Analytics > assignmentsynapsnews

Home Data Develop Integrate Monitor Manage

Synapse live Validate all Publish all

Run Undo Publish Query plan Connect to Built-in Use database Data

1 -- What are the top 5 busiest terminals for bike pickup?

2 SELECT Top (5) station.station\_id AS Station\_ID, station.name AS Station\_name,

3 trip.count\_StartTerminal

4 FROM

5 (

6 SELECT StartTerminal AS start\_terminal,

7 | COUNT(StartTerminal) AS count\_StartTerminal

8 FROM trip\_data

9 GROUP BY StartTerminal

10 ) AS trip

11 INNER JOIN

12 station\_data AS station

13 ON station.station\_id = trip.start\_terminal

14 ORDER BY count\_StartTerminal DESC

15

16

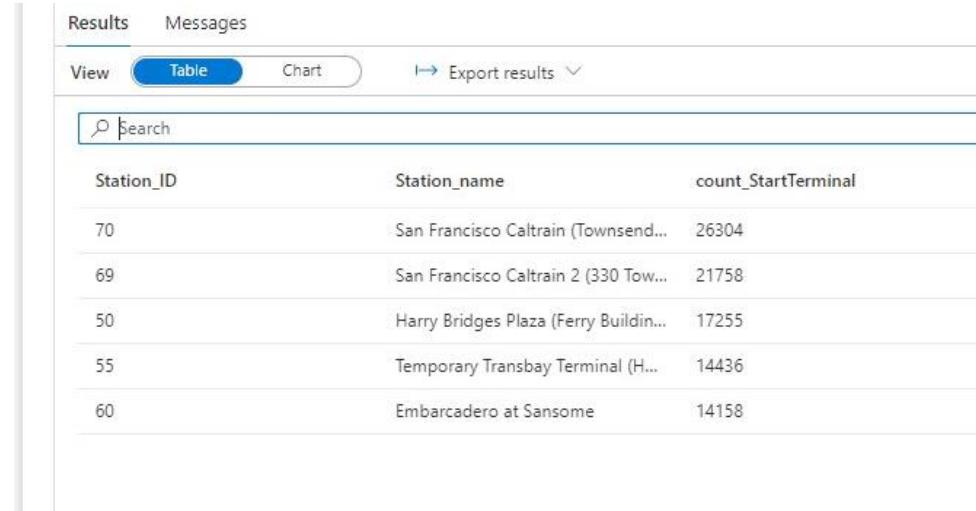
Results Messages

View Table Chart Export results

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

**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:**

The screenshot shows a SQL query results window with the following interface elements:

- Top navigation: Results, Messages.
- View dropdown: Table (selected), Chart.
- Export results button.
- Search bar.
- Table data:

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?*

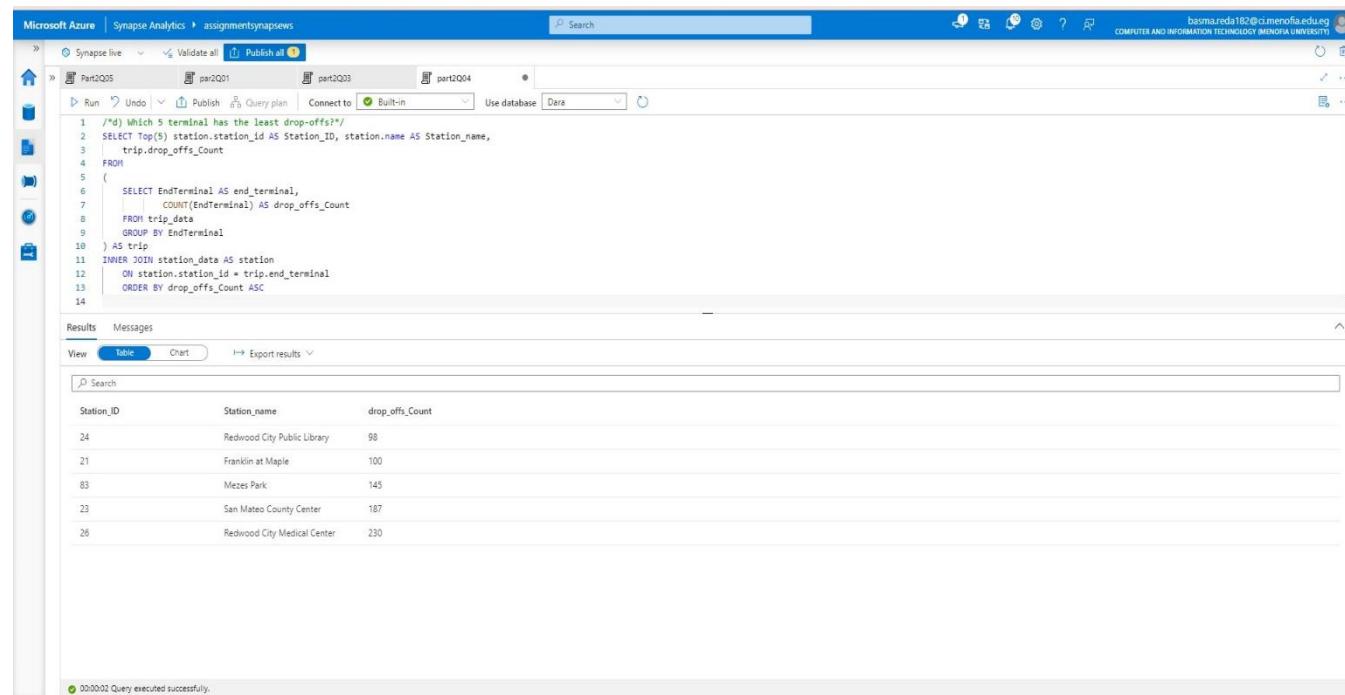
**Query:**

```
/*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:

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:



The screenshot shows the Microsoft Azure Synapse Analytics interface. In the top navigation bar, 'Synapse live' is selected. The main area displays a query editor with a query and its results. The results pane shows the same data as the previous table, listing five stations with their names and drop-off counts.

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

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_custo...	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:

Microsoft Azure | Synapse Analytics > assignmentsynapsew

Search

Validate all Publish all 1

Data

Workspace Linked

Filter resources by name

Lake database 1

Dara

Tables

station\_data

trip\_data

Columns

Part2QS

Run Undo Publish Query plan Connect to Built-in Use database Dara

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

Results Messages

View Table Chart Export results

Search

month_year	Trip_Count	Total_Duration...	Total_Bikes_co...	Total_custo...	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.

Properties

General Related (0)

Name \* Part2QS

Description

Type .sql script

Size 164 bytes

Results settings per query

First 5000 rows (default)

All rows

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](#)