ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

MSc in Business Analytics – DMBI
Assignment #3

Azure Stream Analytics

Vogiatzis George: p2821827

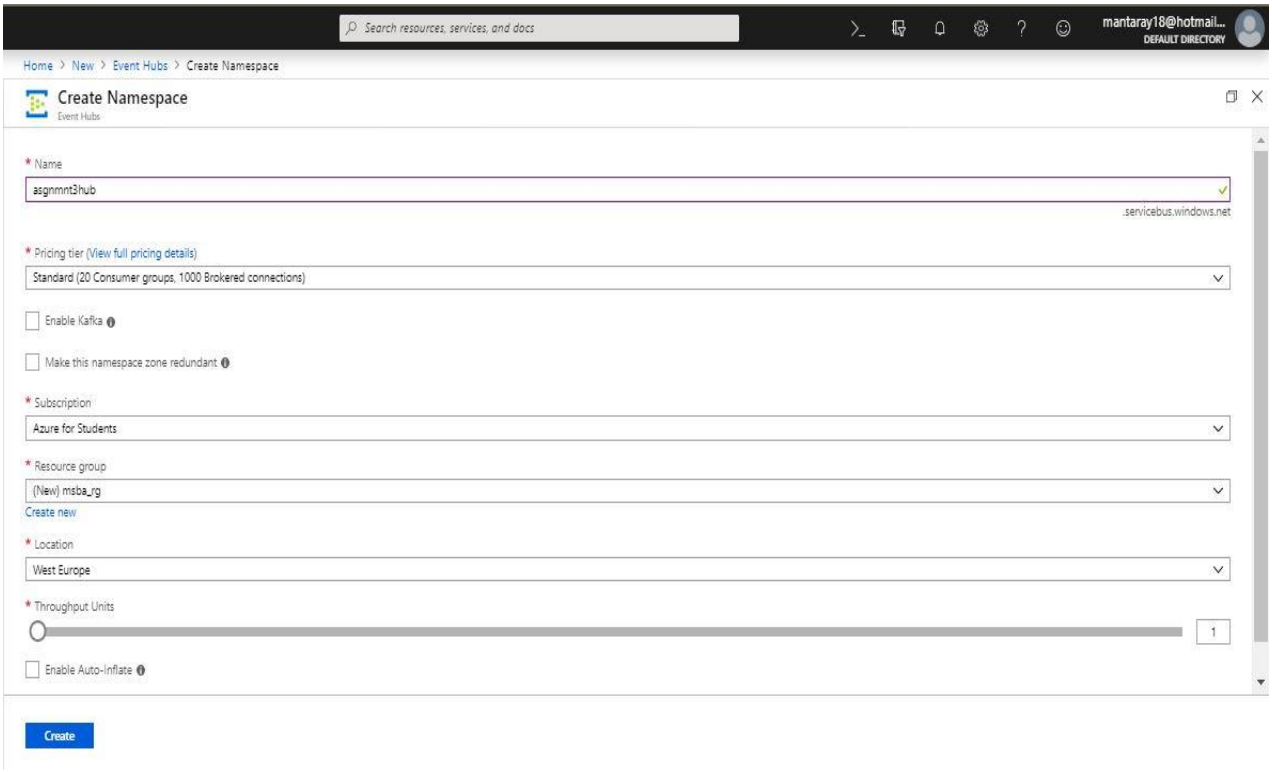
Zourou Mirsini: p2821828

Fall Quarter Pt. 2018

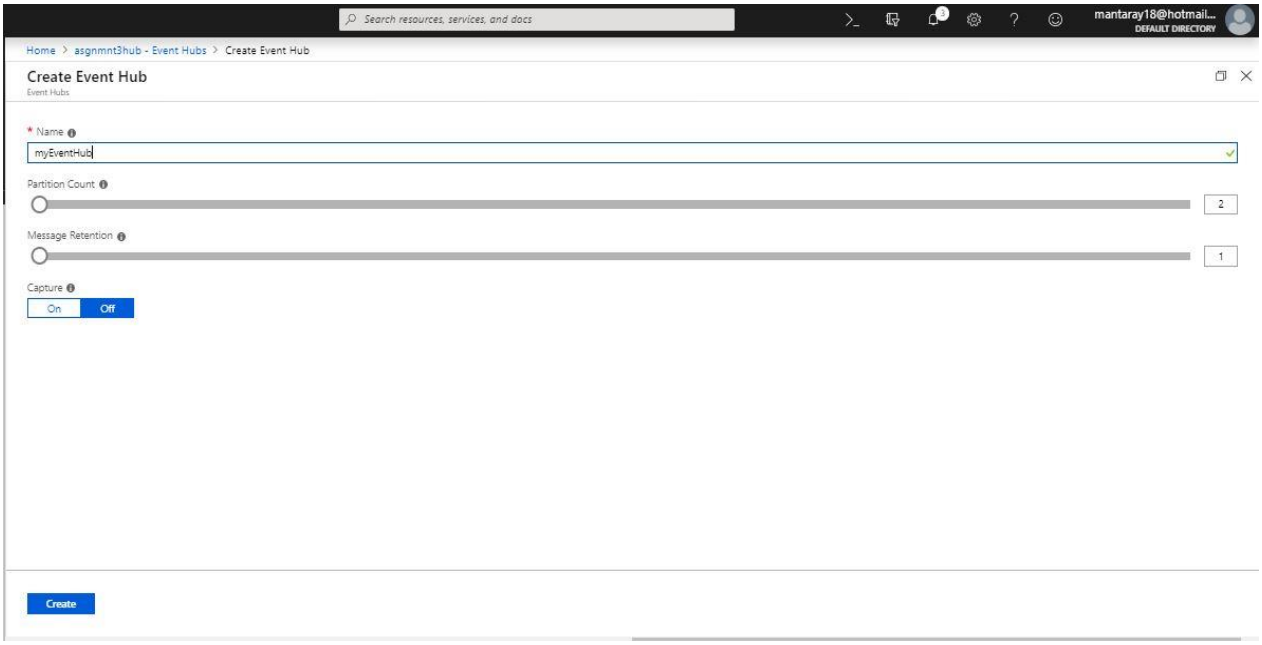
Contents

Steps of the procedure

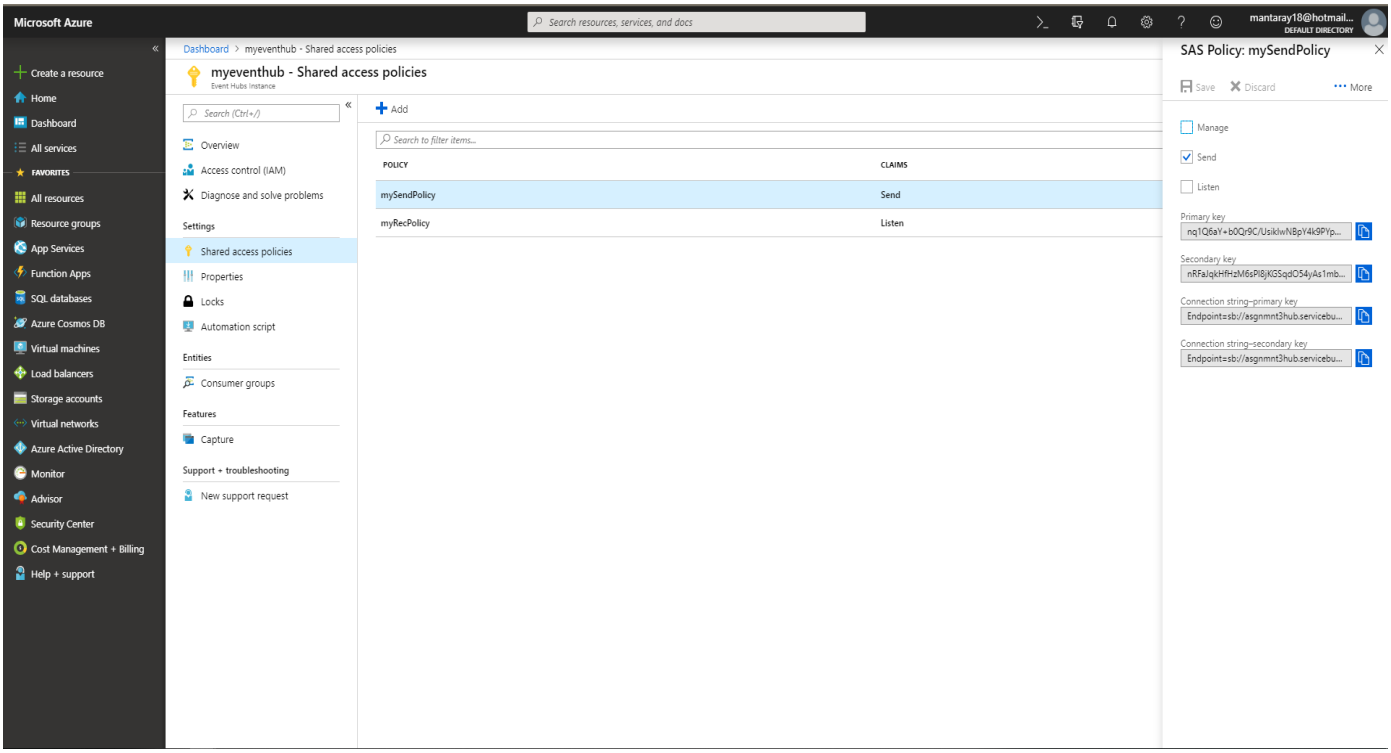
Namespace creation

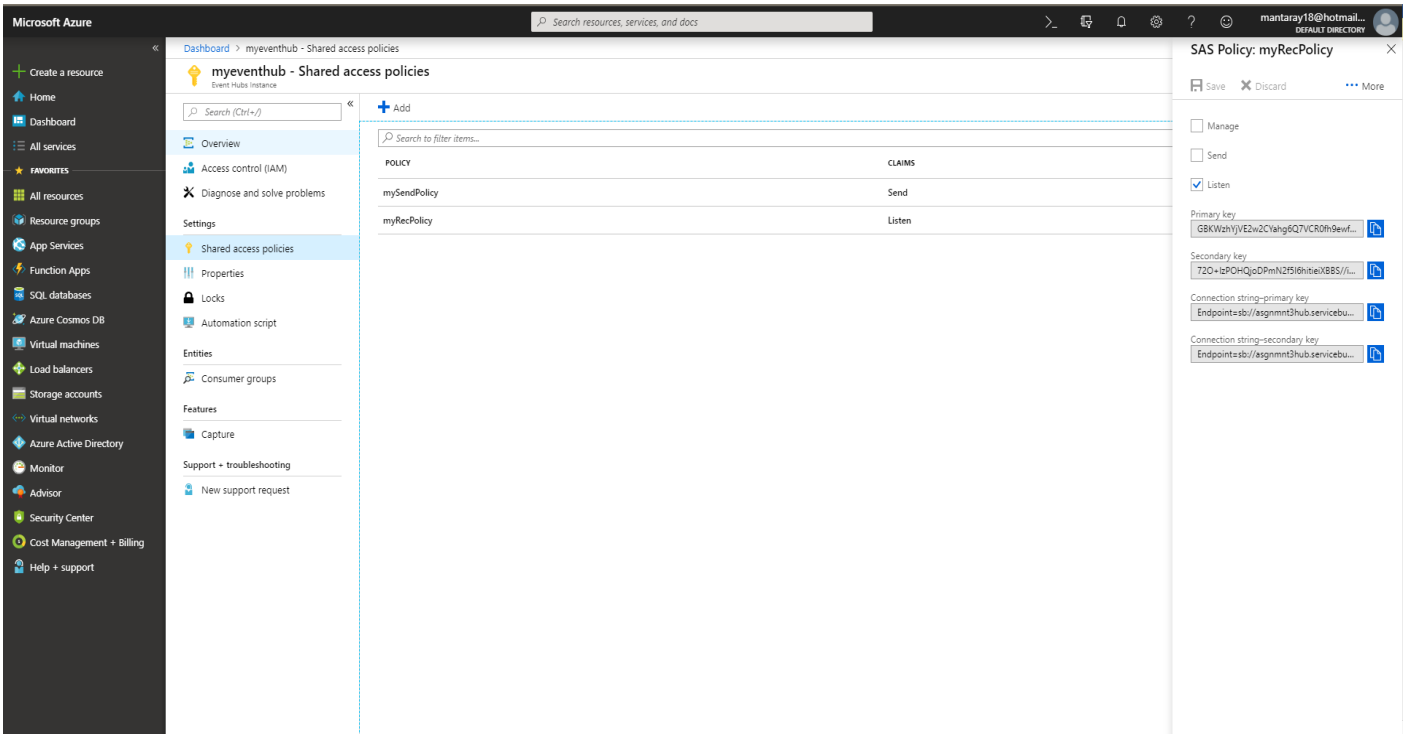


Event hub creation

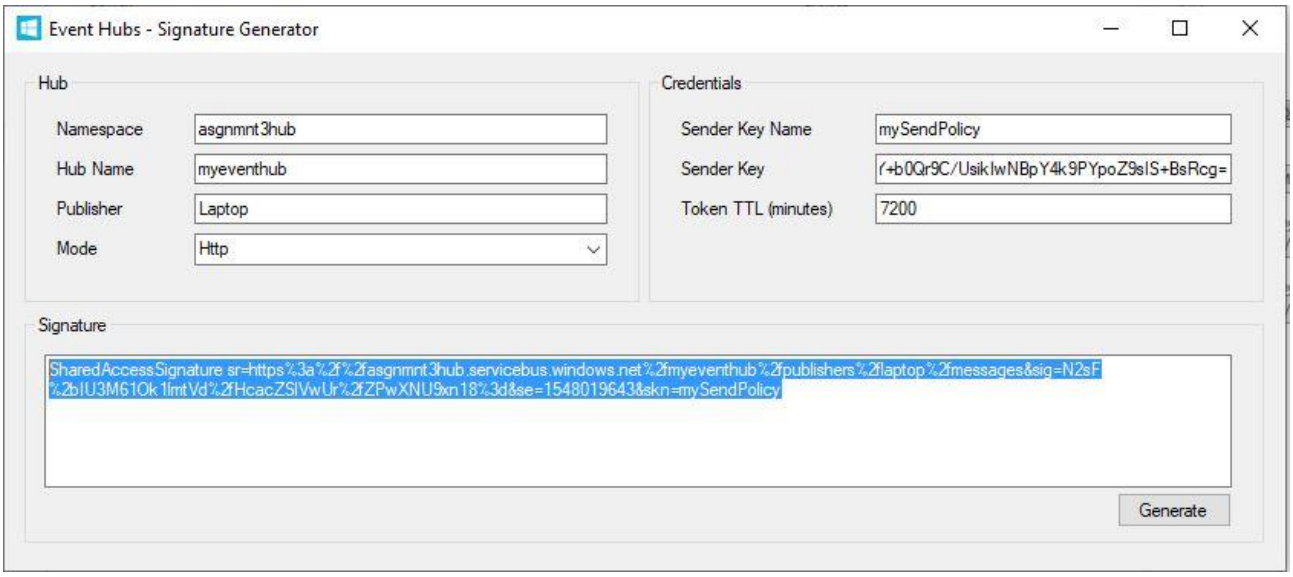


Setting policies





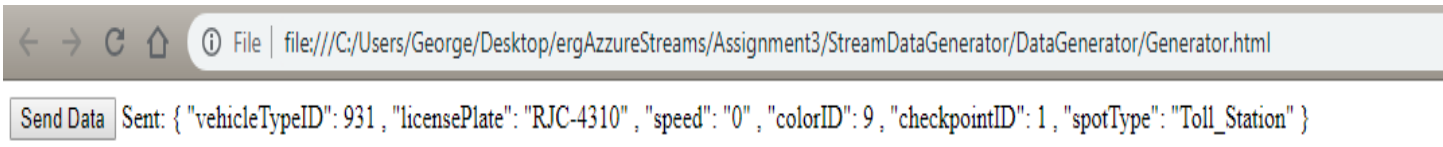
Setting up signature



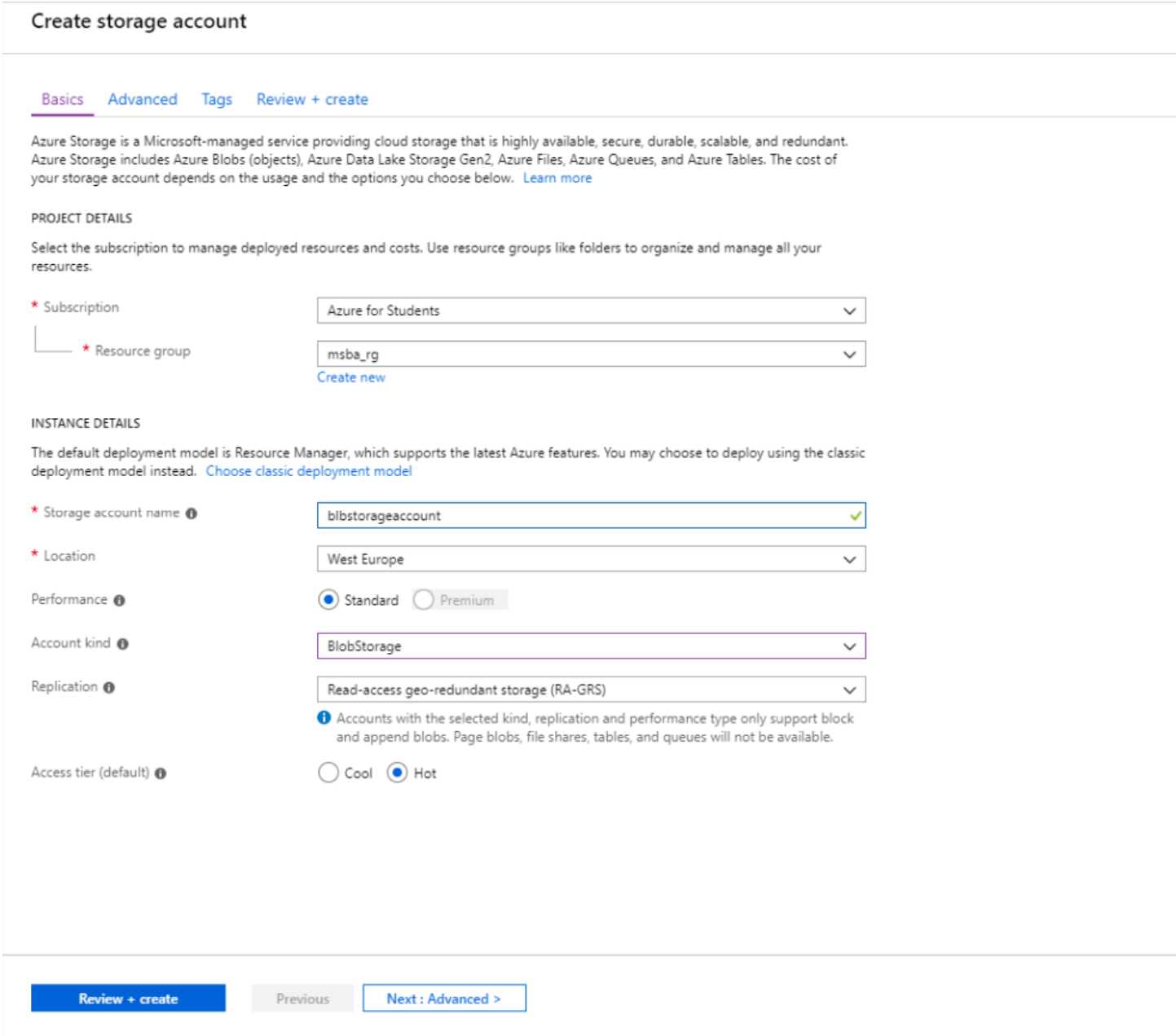
Editing html configuration

```
/***************/
/*** CONFIG ***/
/***************/

//Use the signature generator: https://github.com/sandrinodimattia/RedDog/releases
var sas = "SharedAccessSignature sr=https%3a%2f%2fmscba-aueb.servicebus.windows.net%2feventhubdemo%2fpublishers%2flaptop%
var serviceNamespace = "mscba-aueb";
var hubName = "eventhubdemo";
var deviceName = "Laptop";
```

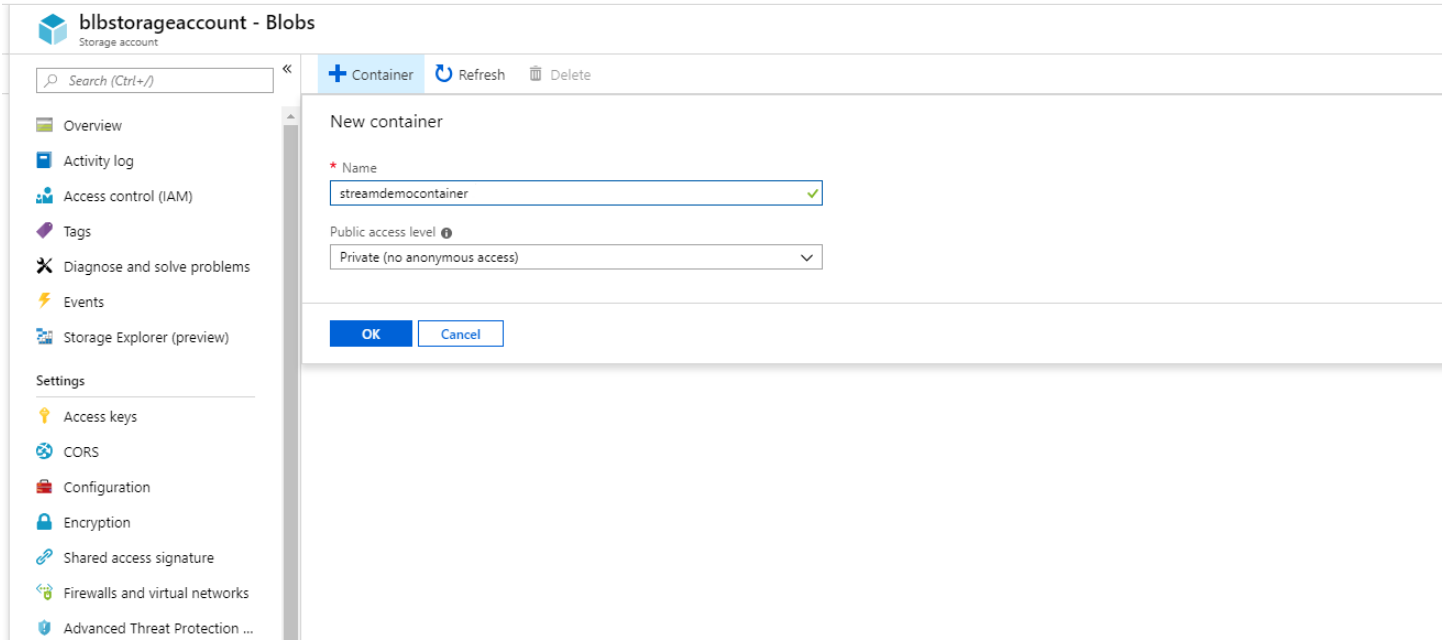
Feeding event hub with Generator.html



Blob Storage Account Creation

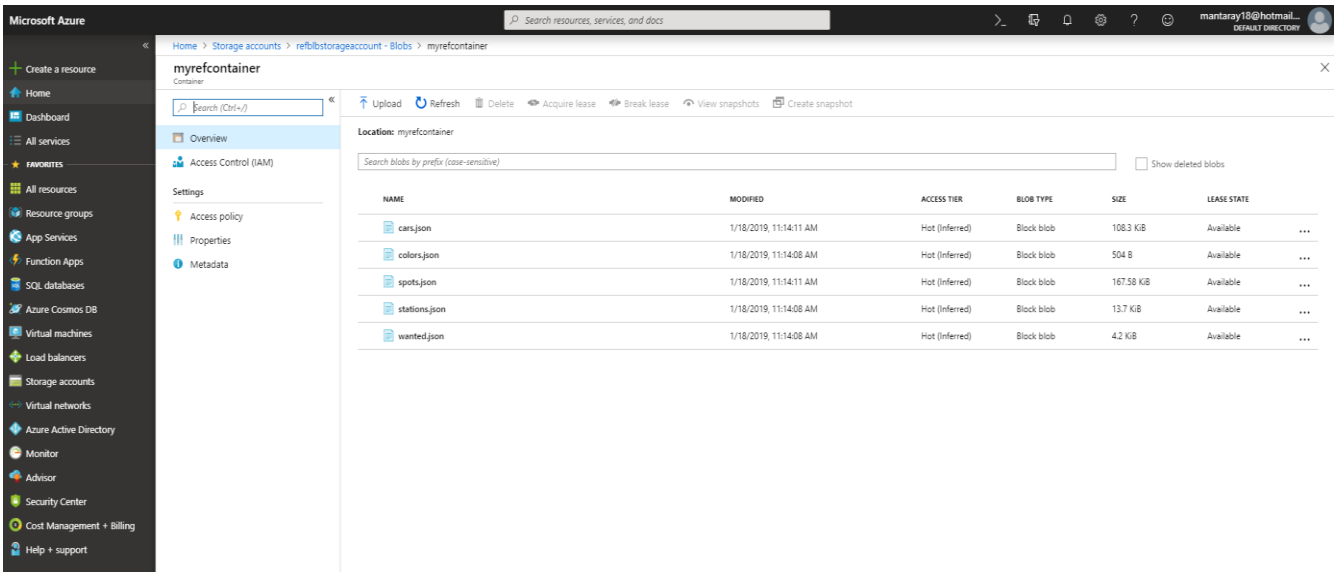


Storage account container creation

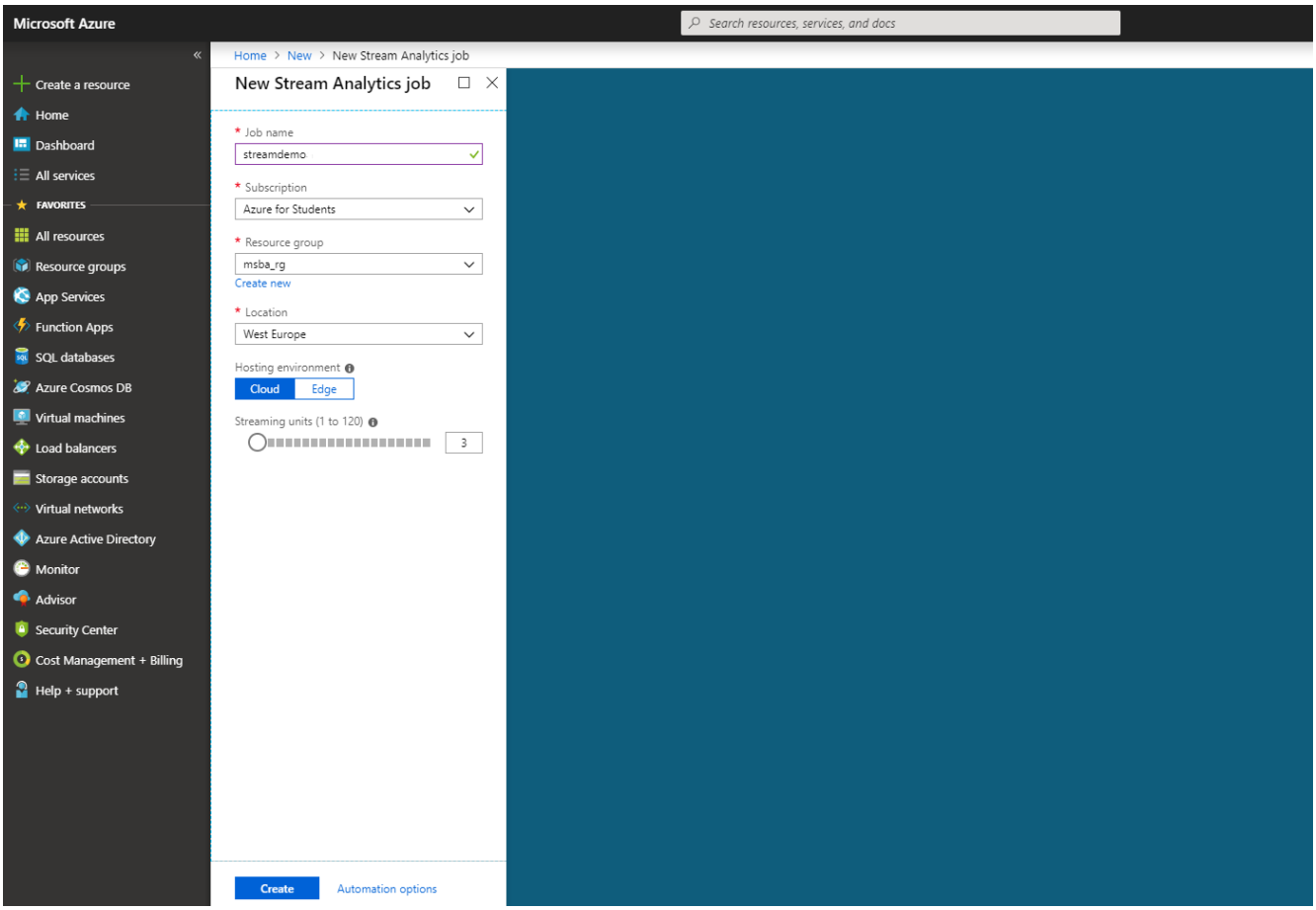


Reference data files

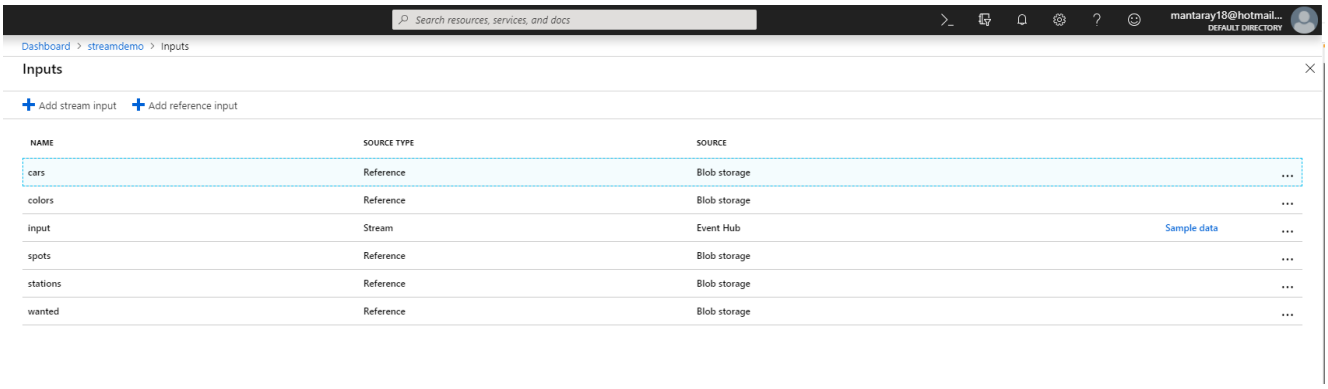
To make better use of our reference data files we first had to convert the txt files to csv via Exel and then to json with the help of an online converter. Particularly for the txt with the wanted license plates we had to add a “column” name on the beginning of the file to convert to a valid json file. After the transformations we uploaded our json reference file to another blob storage account

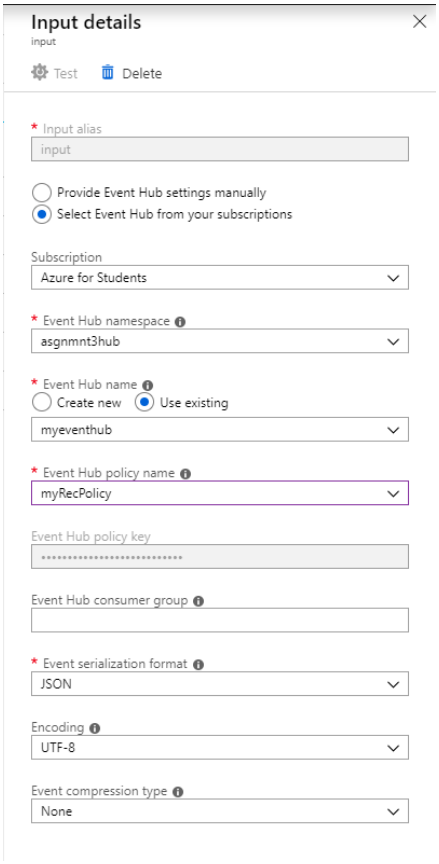


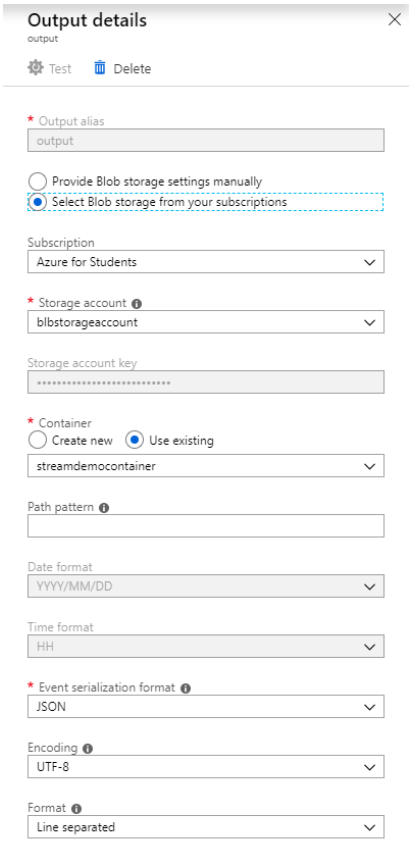
Stream demo creation



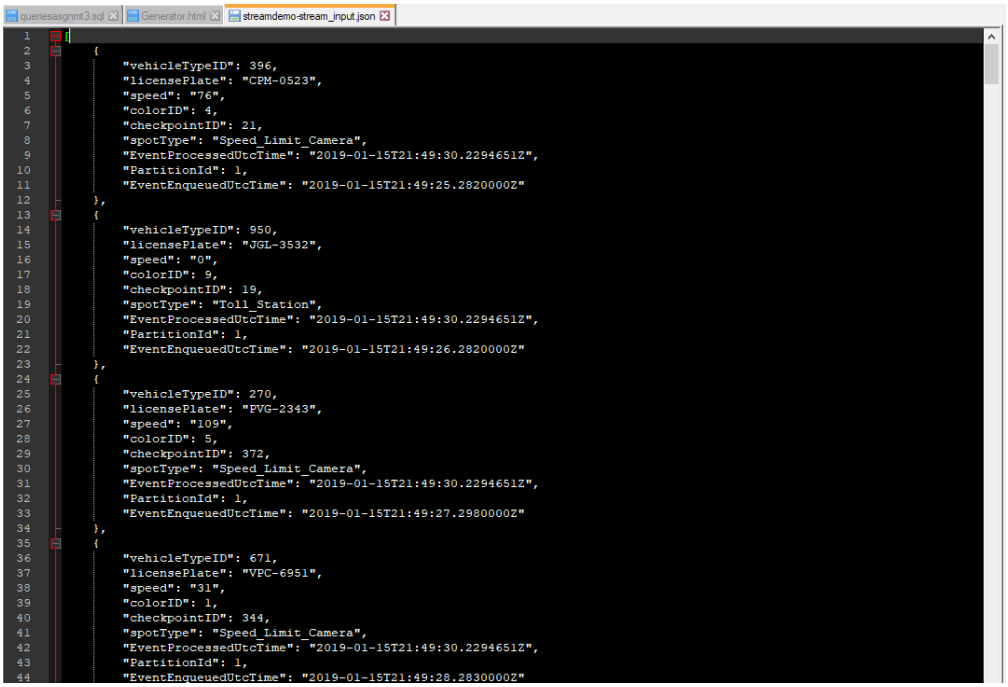
Stream Demo Inputs Outputs and Settings





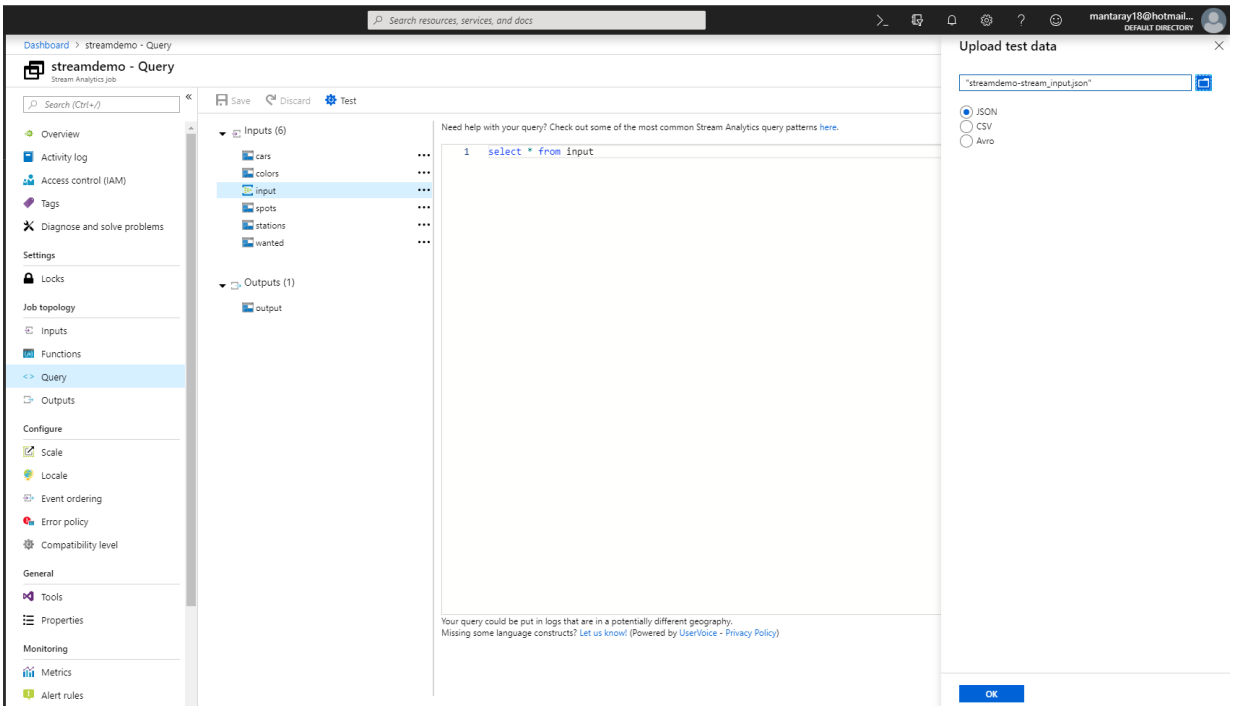


Input sample data for stream demo

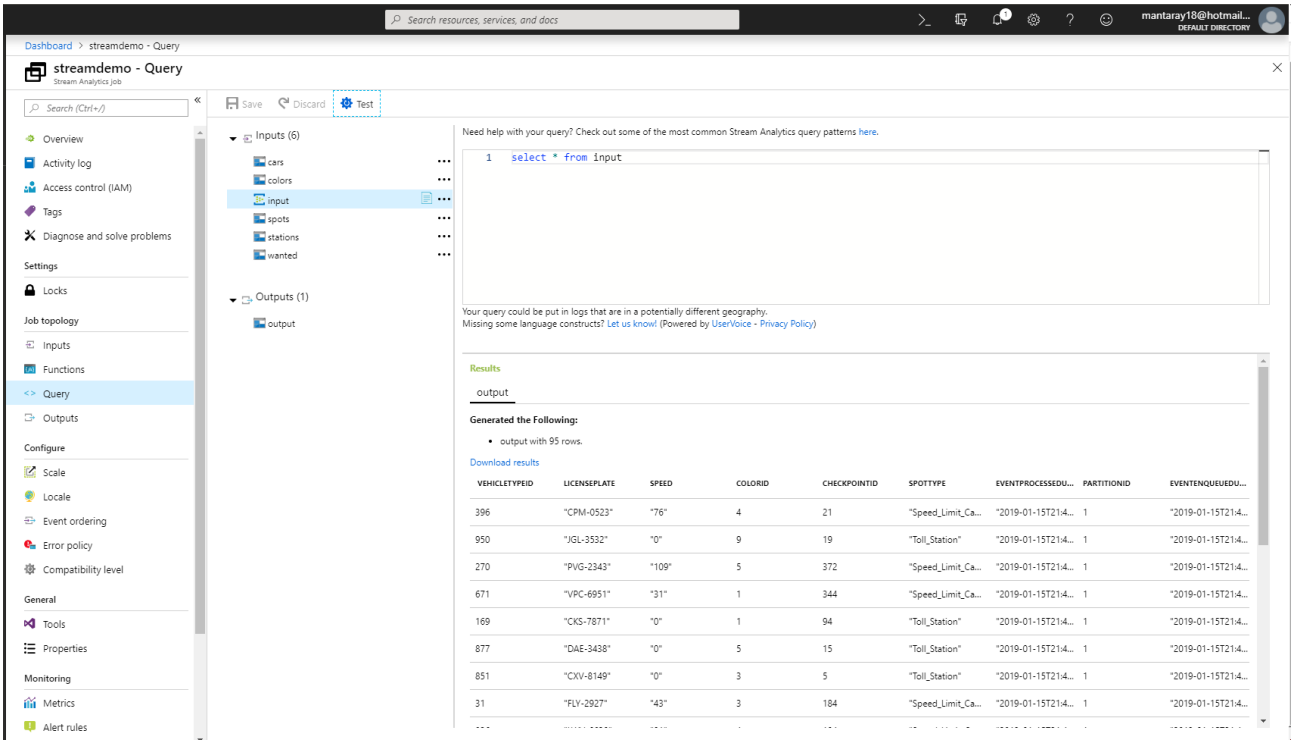


```
[
    {
        "vehicleTypeID": 396,
        "licensePlate": "CPM-0523",
        "speed": "76",
        "colorID": 4,
        "checkpointID": 21,
        "spotType": "Speed_Limit_Camera",
        "EventProcessedUtcTime": "2019-01-15T21:49:30.2294651Z",
        "PartitionId": 1,
        "EventEnqueuedUtcTime": "2019-01-15T21:49:25.2820000Z"
    },
    {
        "vehicleTypeID": 950,
        "licensePlate": "JGL-3532",
        "speed": "0",
        "colorID": 9,
        "checkpointID": 19,
        "spotType": "Toll_Station",
        "EventProcessedUtcTime": "2019-01-15T21:49:30.2294651Z",
        "PartitionId": 1,
        "EventEnqueuedUtcTime": "2019-01-15T21:49:26.2820000Z"
    },
    {
        "vehicleTypeID": 270,
        "licensePlate": "PVG-2343",
        "speed": "109",
        "colorID": 5,
        "checkpointID": 372,
        "spotType": "Speed_Limit_Camera",
        "EventProcessedUtcTime": "2019-01-15T21:49:30.2294651Z",
        "PartitionId": 1,
        "EventEnqueuedUtcTime": "2019-01-15T21:49:27.2980000Z"
    },
    {
        "vehicleTypeID": 671,
        "licensePlate": "VPC-6951",
        "speed": "31",
        "colorID": 1,
        "checkpointID": 344,
        "spotType": "Speed_Limit_Camera",
        "EventProcessedUtcTime": "2019-01-15T21:49:30.2294651Z",
        "PartitionId": 1,
        "EventEnqueuedUtcTime": "2019-01-15T21:49:28.2830000Z"
```

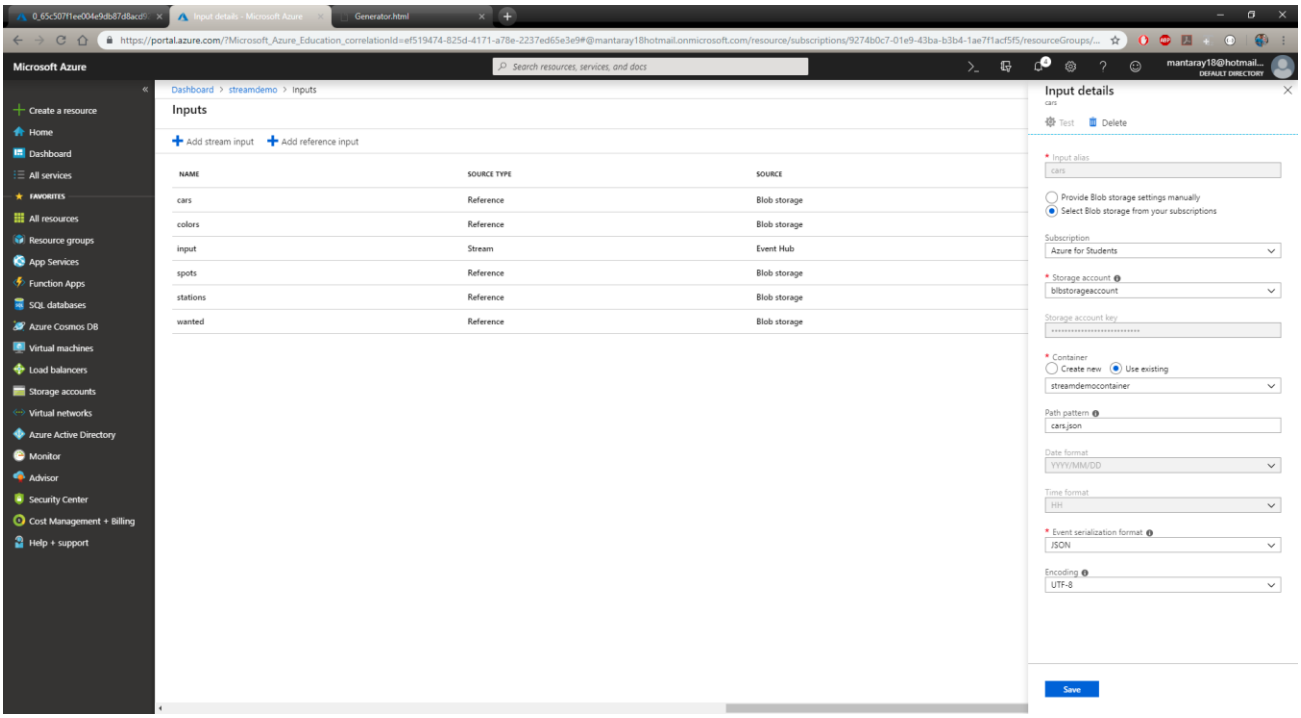
Upload input file for stream demo



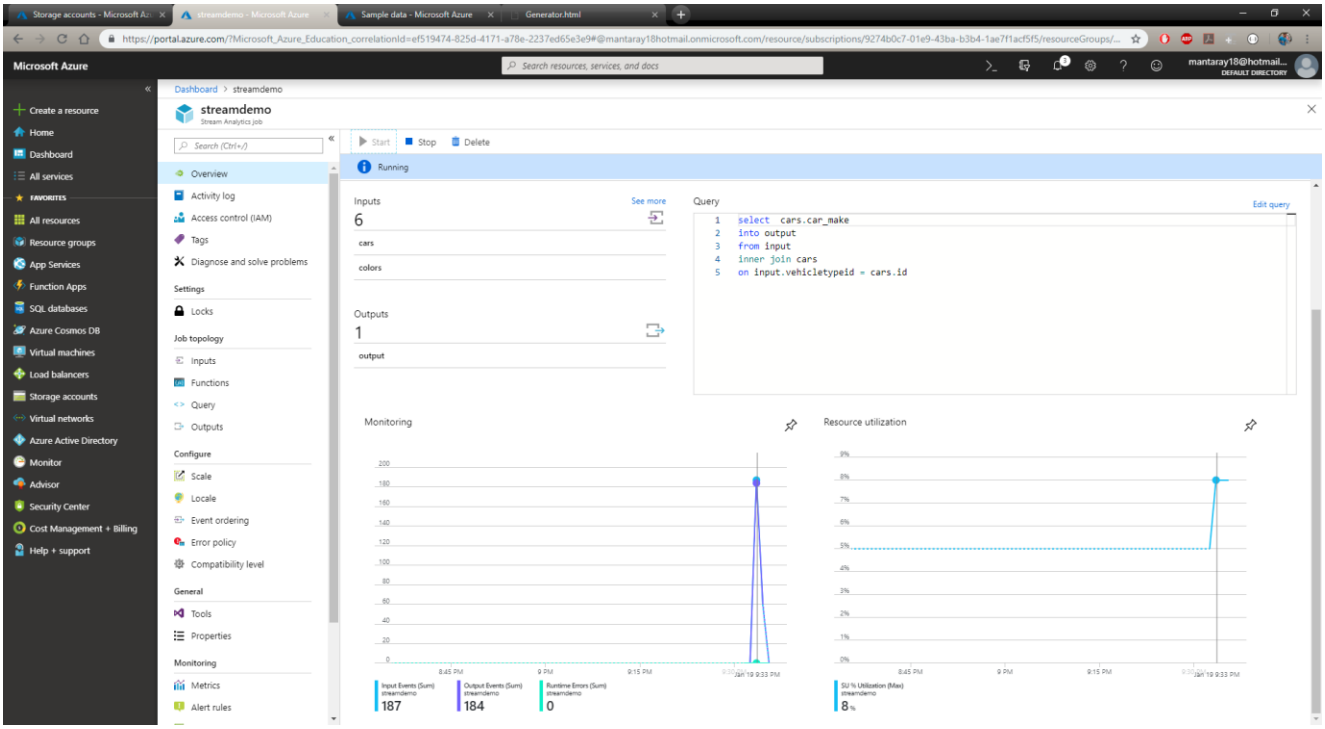
Test Query



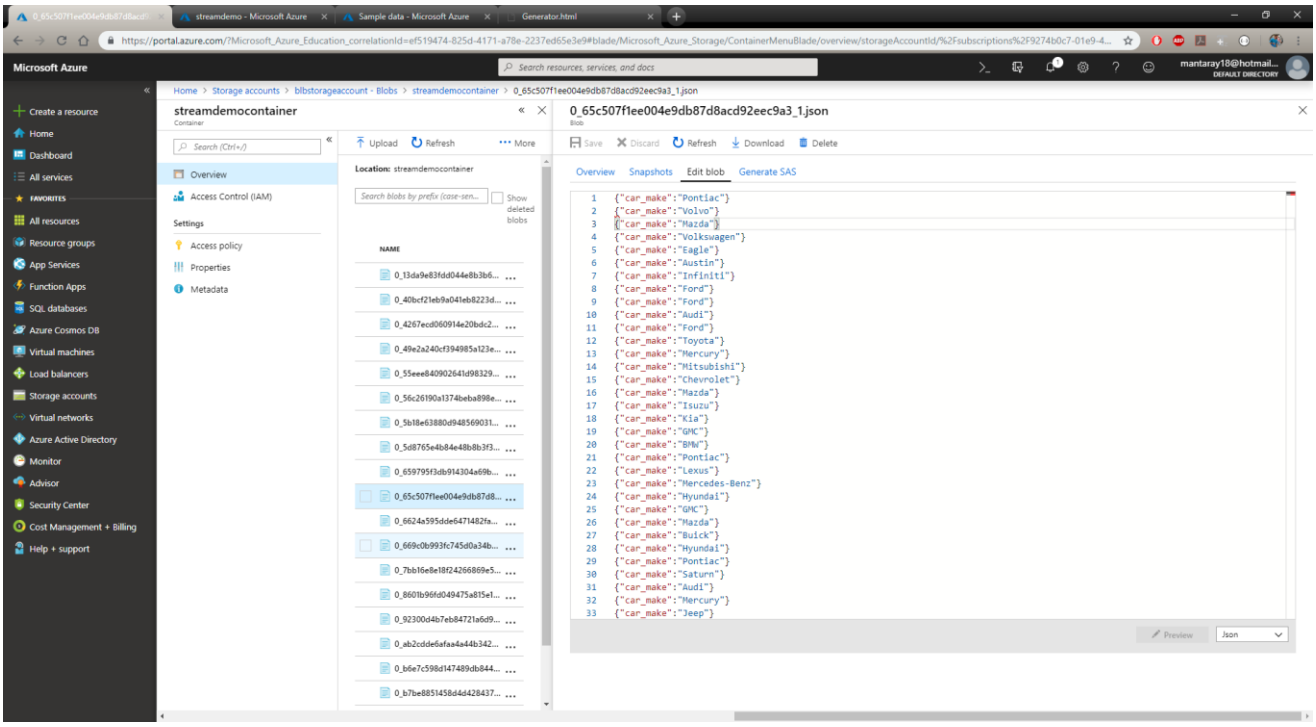
Reference Data Settings



Stream Output



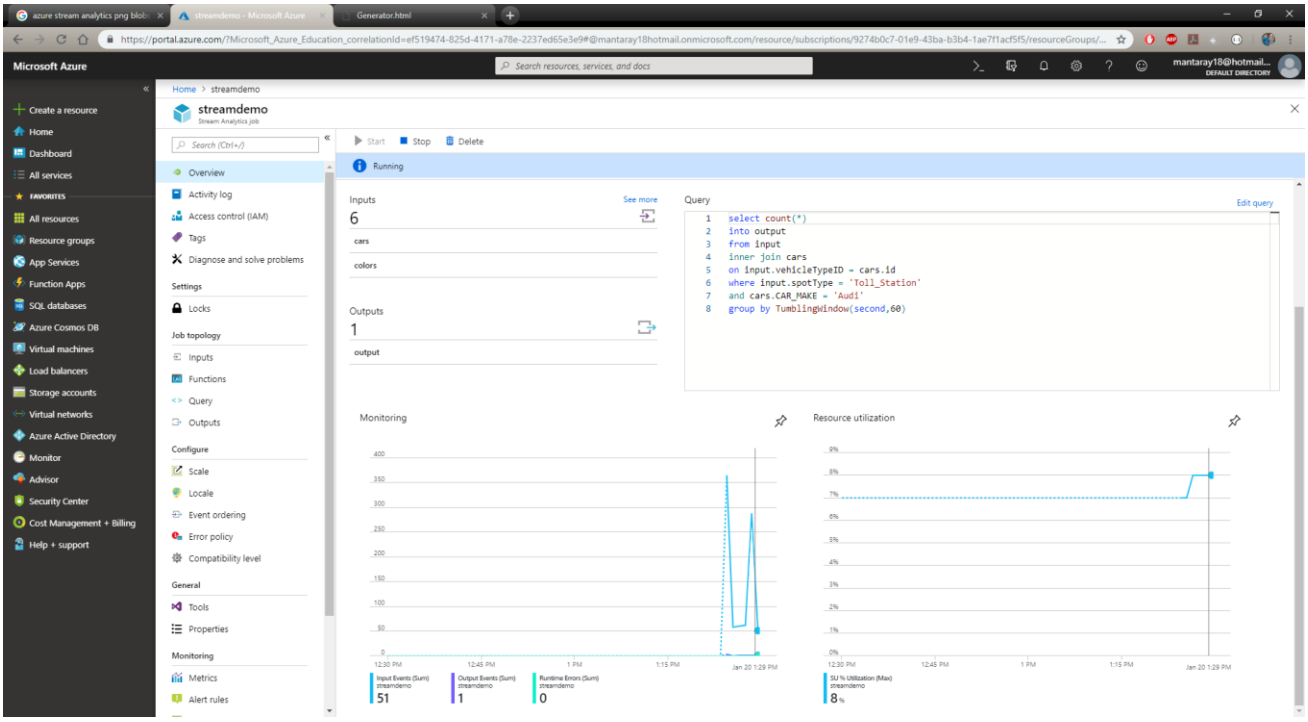
Generated Blob



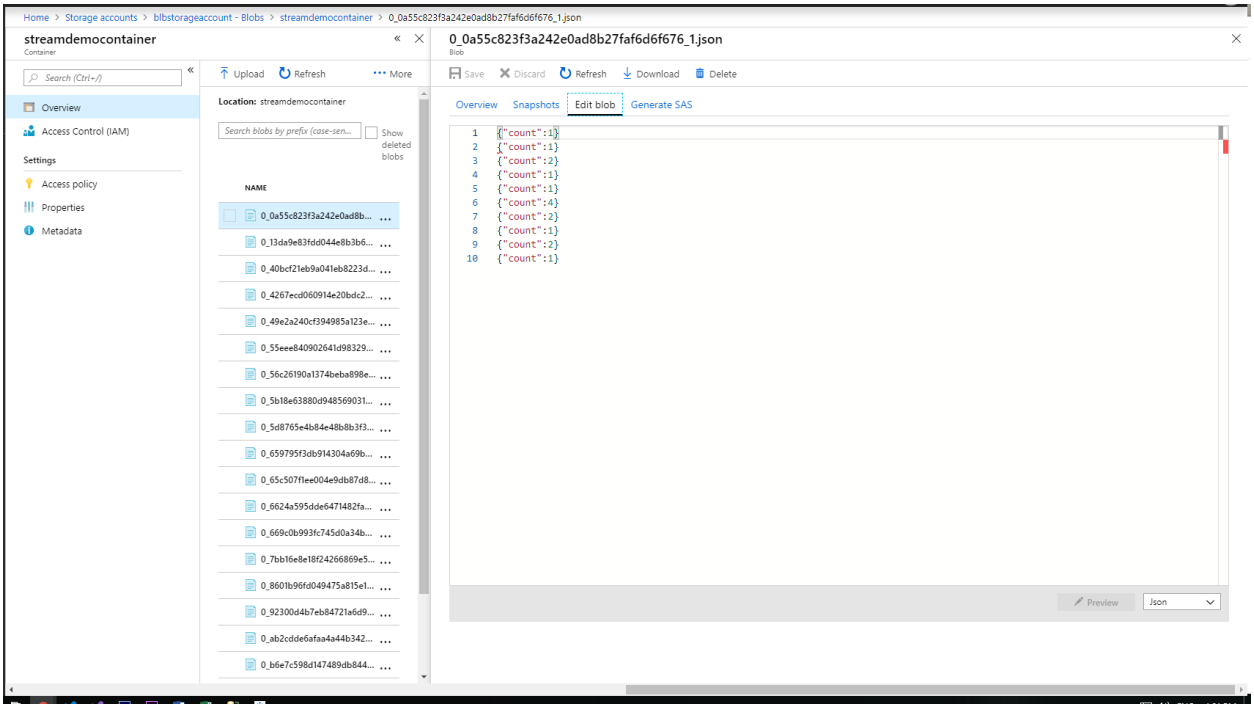
Queries

Query 1

```
select count(*)
into output
from input
inner join cars
on input.vehicleTypeID = cars.id
where input.spotType = 'Toll_Station'
and cars.CAR_MAKE = 'Audi'
group by TumblingWindow(second,60)
```

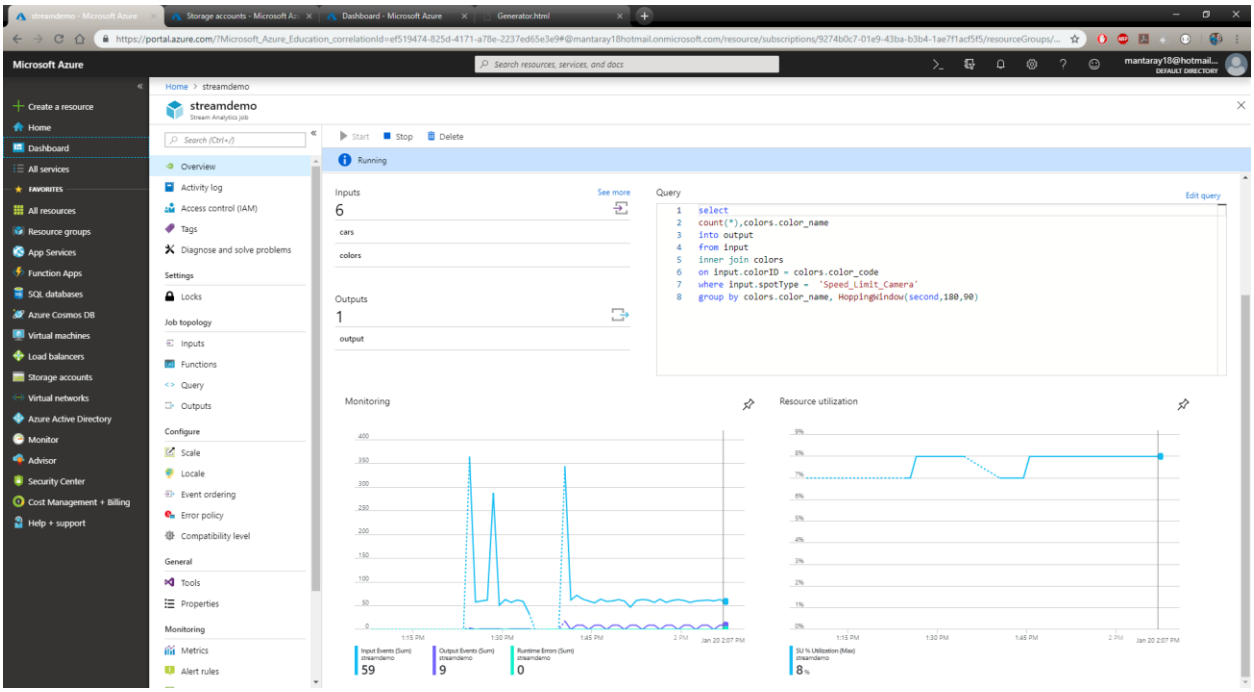


Blob output of query 1

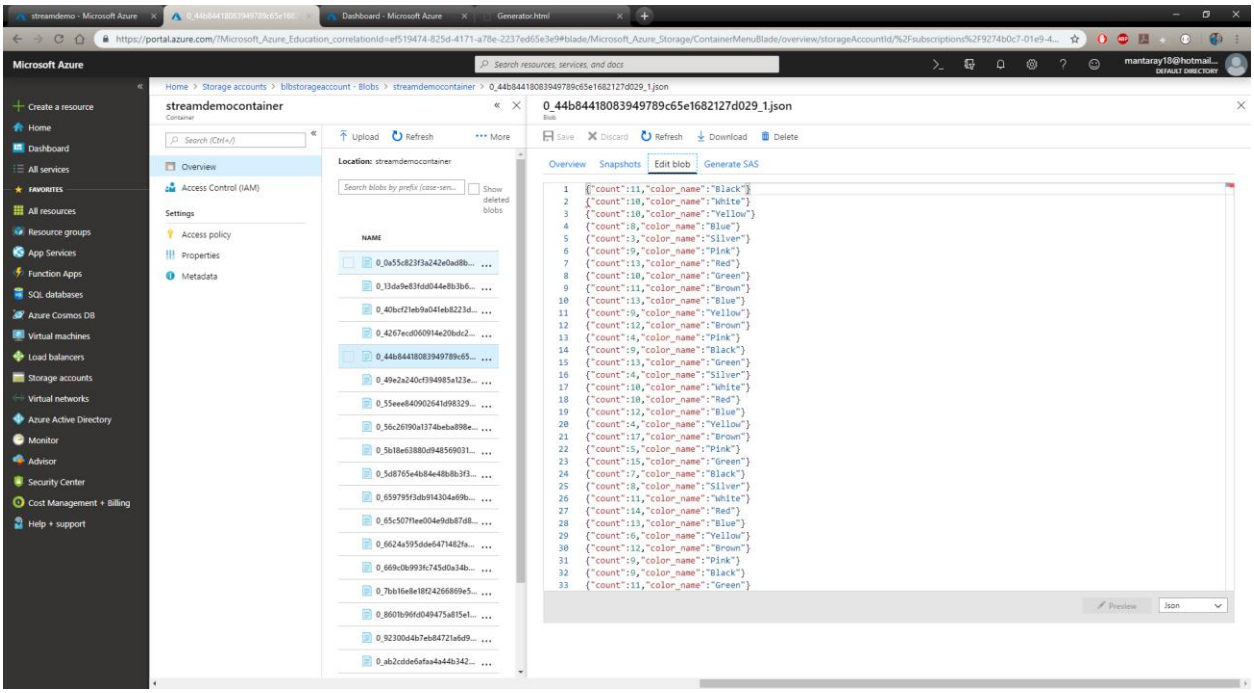


Query 2

```
select

count(*),colors.color_name

into output

from input

inner join colors

on input.colorID = colors.color_code

where input.spotType =  'Speed_Limit_Camera'

group by colors.color_name, HoppingWindow(second,180,90)
```

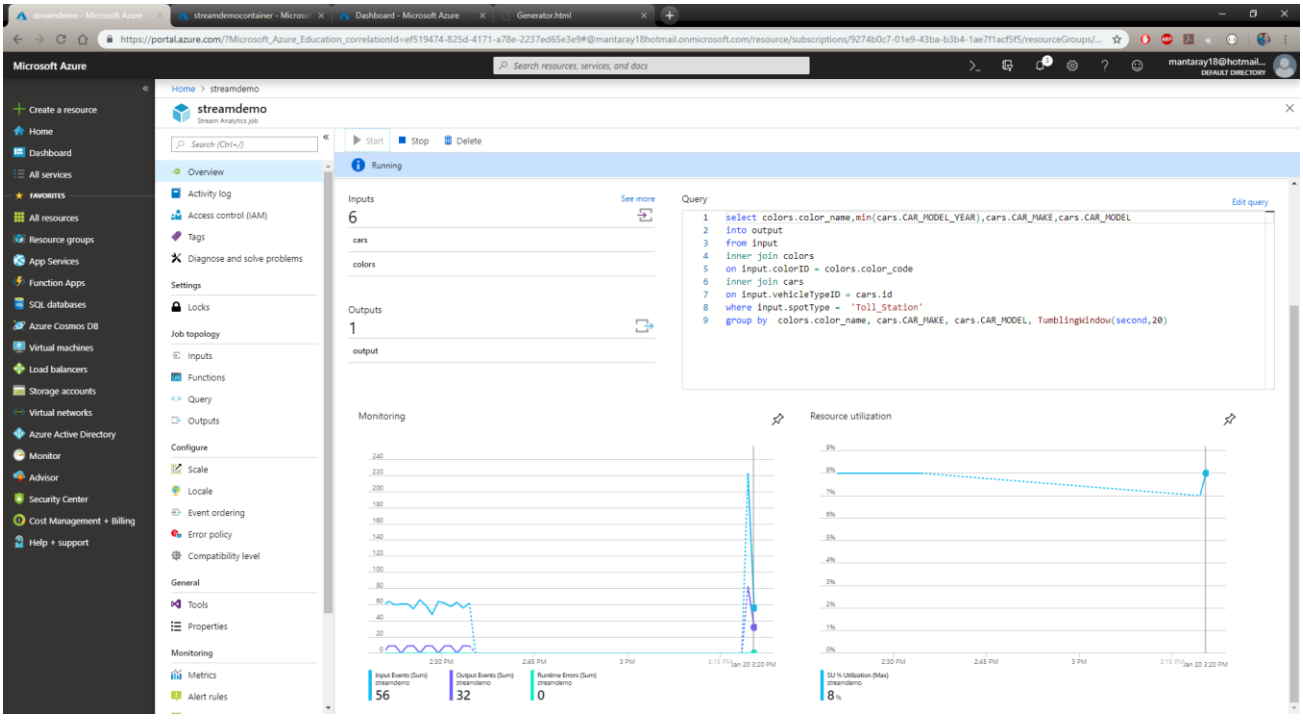


Blob output query 2

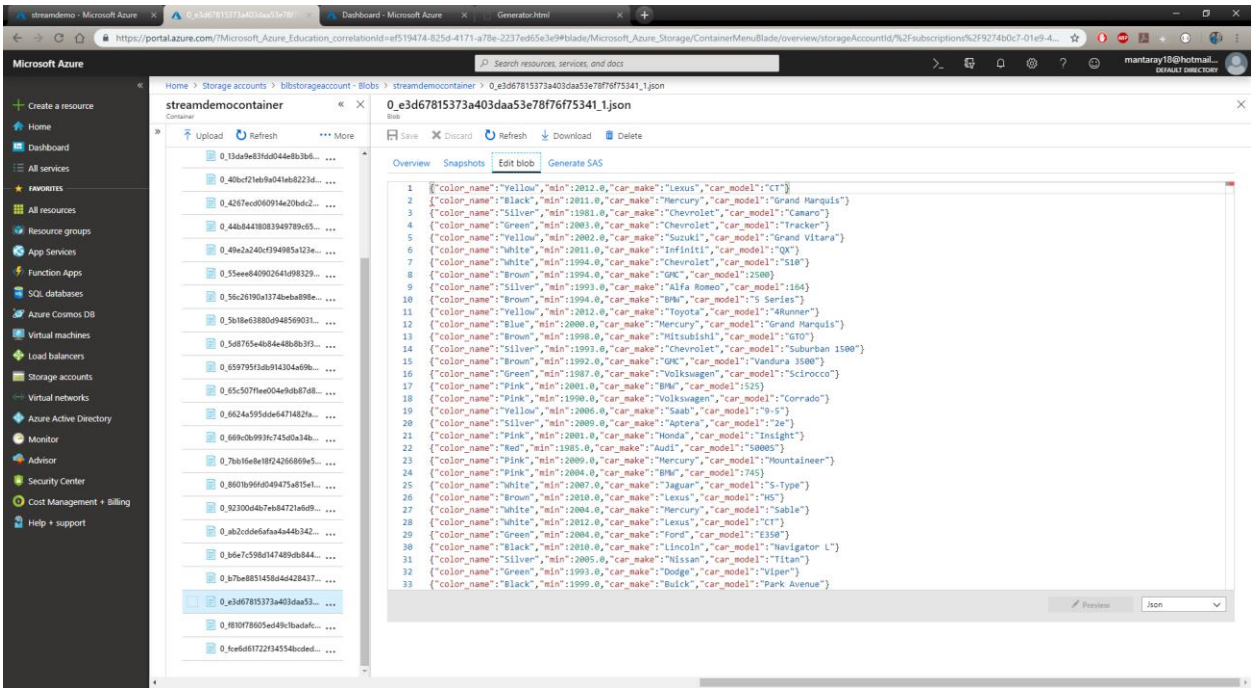


Query 3

```
select colors.color_name,min(cars.CAR_MODEL_YEAR),cars.CAR_MAKE,cars.CAR_MODEL
into output
from input
inner join colors
on input.colorID = colors.color_code
inner join cars
on input.vehicleTypeID = cars.id
where input.spotType =  'Toll_Station'
group by  colors.color_name, cars.CAR_MAKE, cars.CAR_MODEL, TumblingWindow(second,20)
```

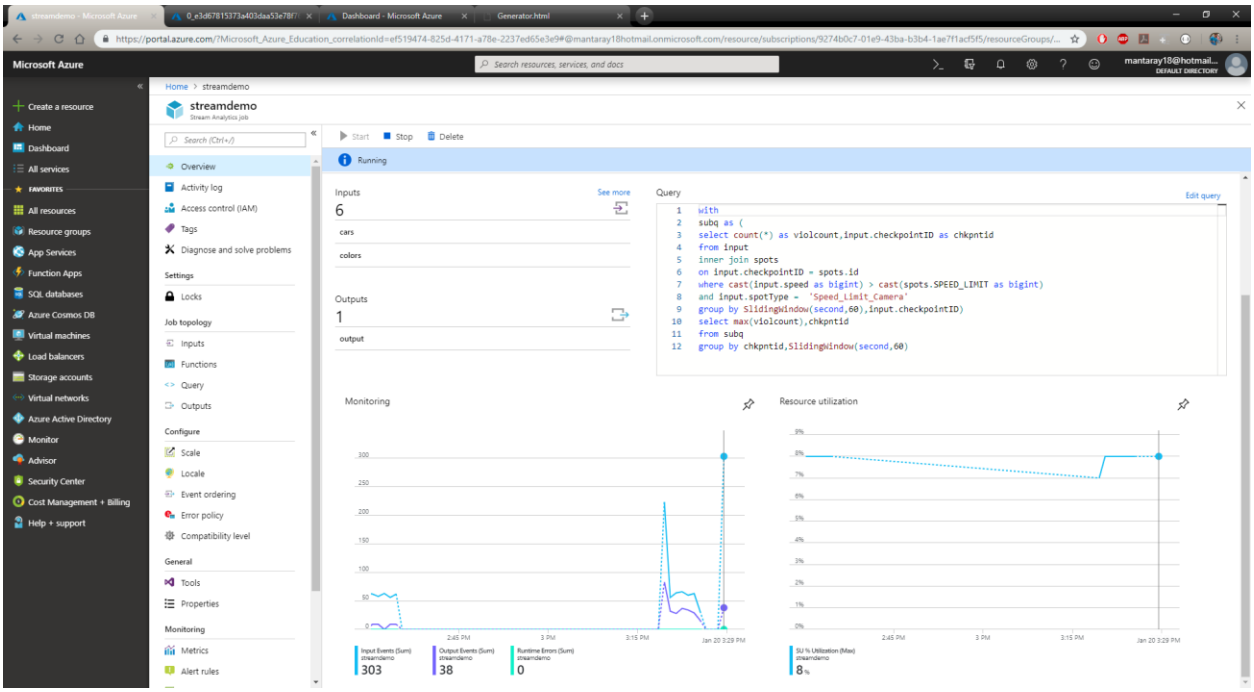


Blob output query 3

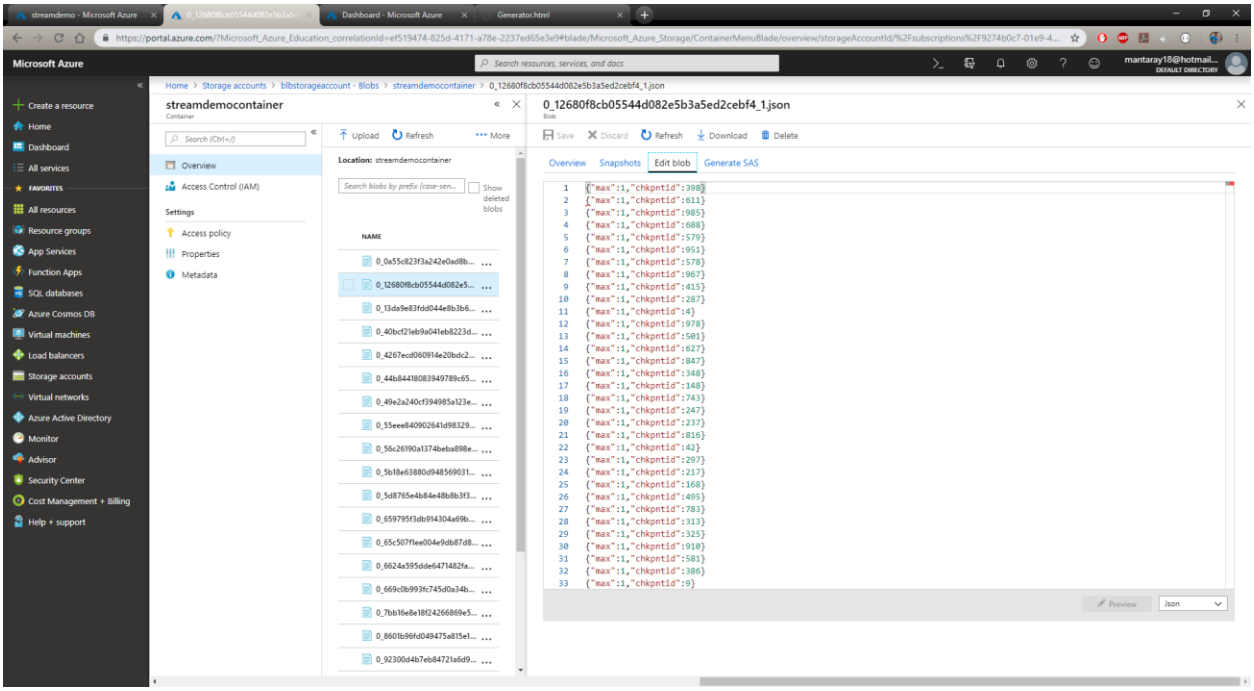


Query 4

```
with

subq as (

select count(*) as violcount,input.checkpointID as chkpntid

from input

inner join spots

on input.checkpointID = spots.id

where cast(input.speed as bigint) > cast(spots.SPEED_LIMIT as bigint)

and input.spotType =  'Speed_Limit_Camera'

group by SlidingWindow(second,60),input.checkpointID)

select max(violcount),chkpntid

from subq

group by chkpntid,SlidingWindow(second,60)
```

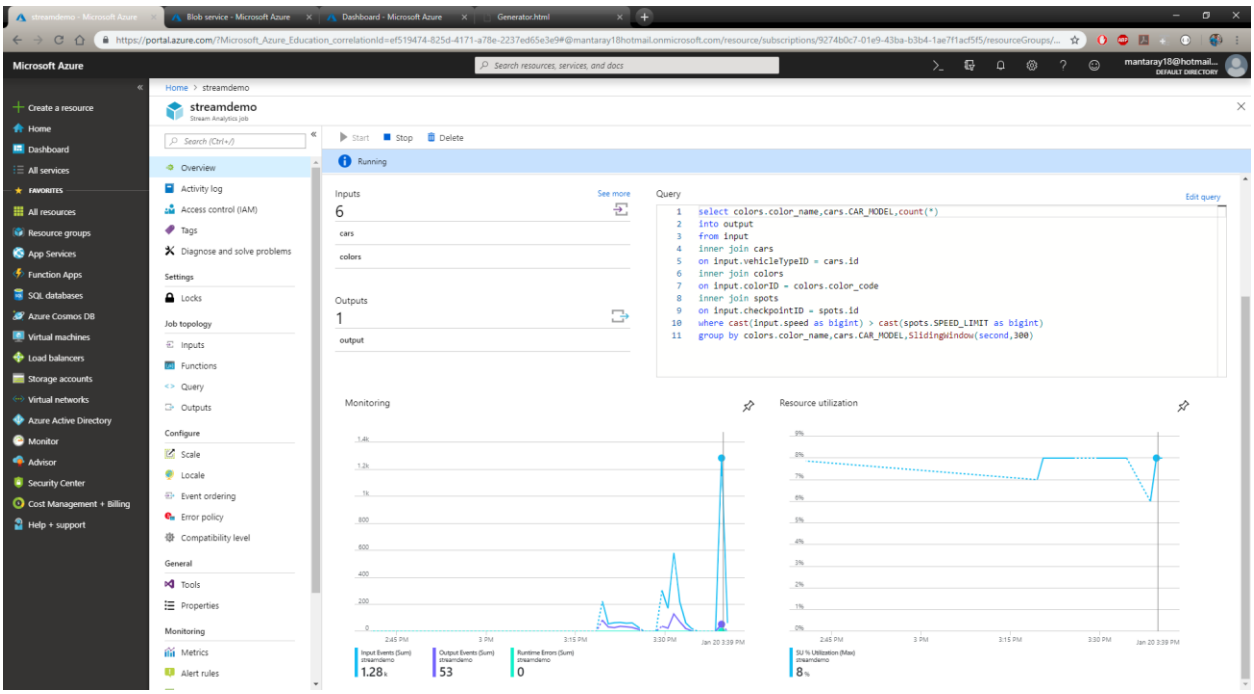


blob output query 4

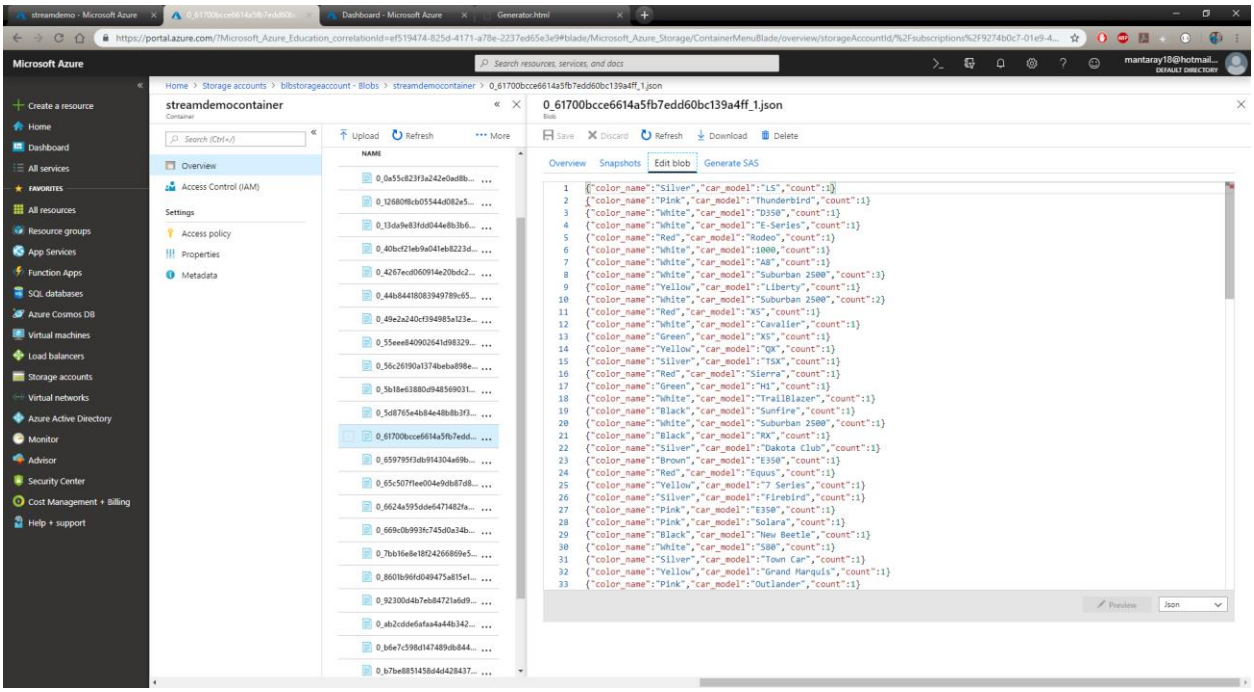


Query 5

```
select colors.color_name,cars.CAR_MODEL,count(*)
into output
from input
inner join cars
on input.vehicleTypeID = cars.id
inner join colors
on input.colorID = colors.color_code
inner join spots
on input.checkpointID = spots.id
where cast(input.speed as bigint) > cast(spots.SPEED_LIMIT as bigint)
group by colors.color_name,cars.CAR_MODEL,SlidingWindow(second,300)
```

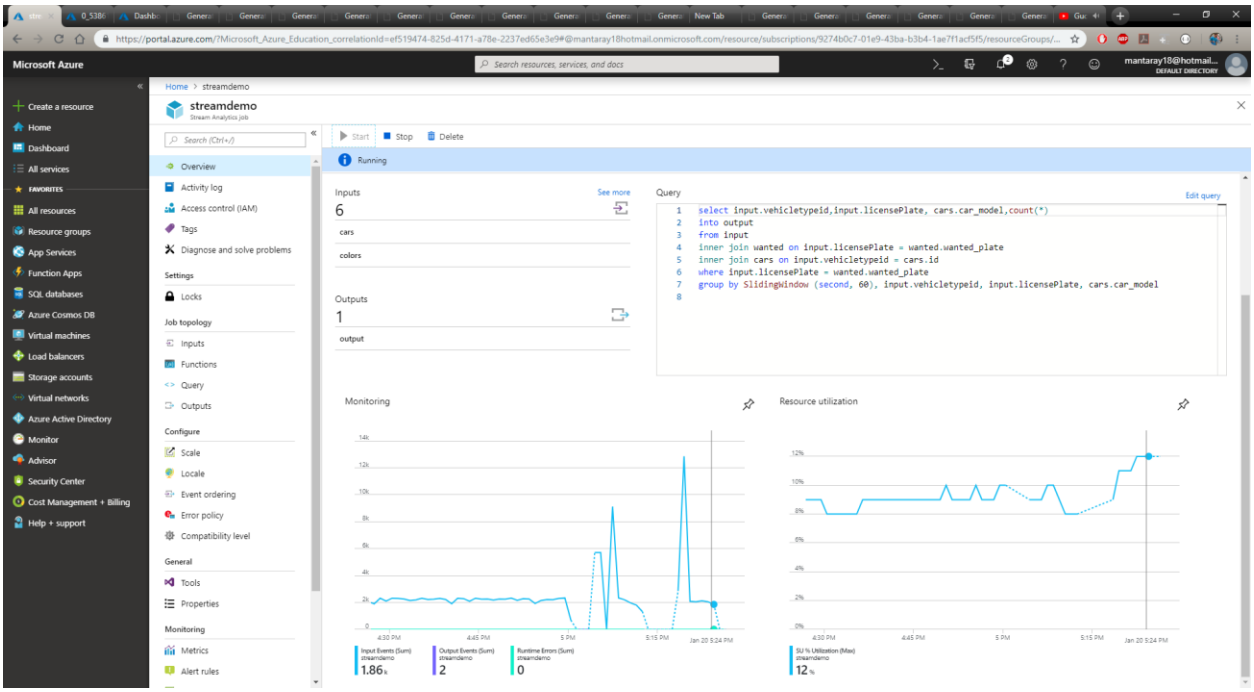


Blob output query 5



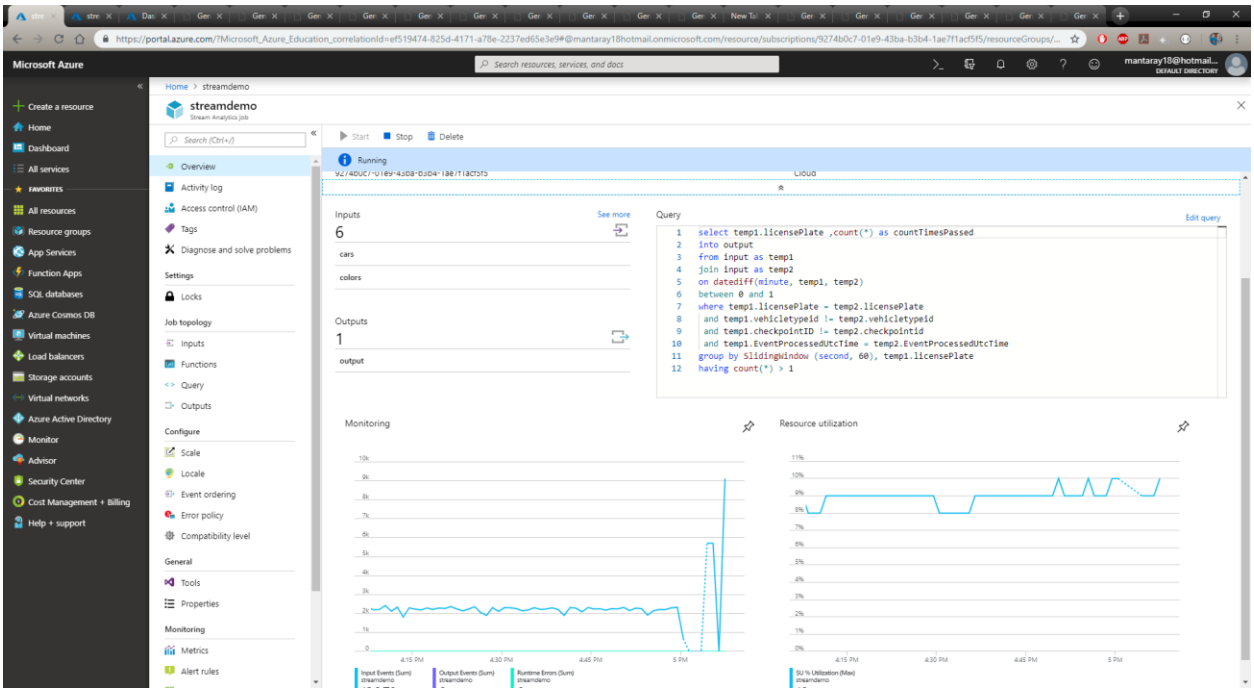
Query 6

```
select input.vehicletypeid,input.licensePlate, cars.car_model,count(*)

into output

from input

inner join wanted on input.licensePlate = wanted.wanted_plate

inner join cars on input.vehicletypeid = cars.id

where input.licensePlate = wanted.wanted_plate

group by SlidingWindow (second, 60), input.vehicletypeid, input.licensePlate, cars.car_model
```



Query 7

```
select temp1.licensePlate ,count(*) as countTimesPassed
into output
from input as temp1
join input as temp2
on datediff(minute, temp1, temp2)
between 0 and 1
where temp1.licensePlate = temp2.licensePlate
and temp1.vehicletypeid != temp2.vehicletypeid
and temp1.checkpointID != temp2.checkpointid
and temp1.EventProcessedUtcTime = temp2.EventProcessedUtcTime
group by SlidingWindow (second, 60), temp1.licensePlate
having count(*) > 1
```



Query 8

with

wantedcars as

(select count(*) as bmwcount

from input

inner join cars on input.vehicletypeid = cars.id

where cars.car_make = 'BMW' and input.spotType = 'Speed_Limit_Camera'

group by TumblingWindow(second,120)),

abovelimit as

(select count(*)as bmwviolations

from

input

inner join cars on input.vehicletypeid = cars.id

inner join spots on input.checkpointID = spots.id

where cars.car_make = 'BMW' and input.spotType = 'Speed_Limit_Camera'

and cast(input.speed as bigint) > cast(spots.speed_limit as bigint)

group by TumblingWindow(second,120))

select case when abovelimit.bmwviolations is null then 0 else abovelimit.bmwviolations end as bmwviolations,

wantedcars.bmwcount,

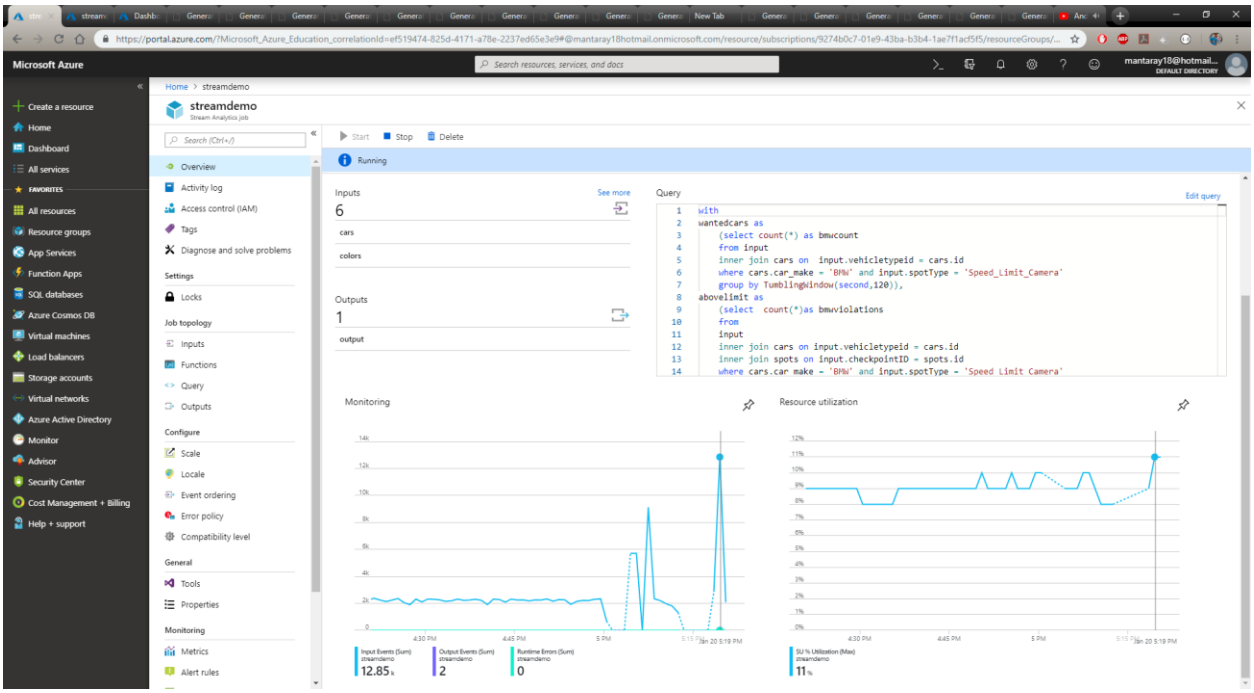
case when ((abovelimit.bmwviolations)*100)/(wantedcars.bmwcount) is null then 0 else ((abovelimit.bmwviolations)*100)/(wantedcars.bmwcount) end as percentage

from wantedcars

left join abovelimit

on datediff(minute, wantedcars, abovelimit)

between 0 AND 2



Blob output query 8