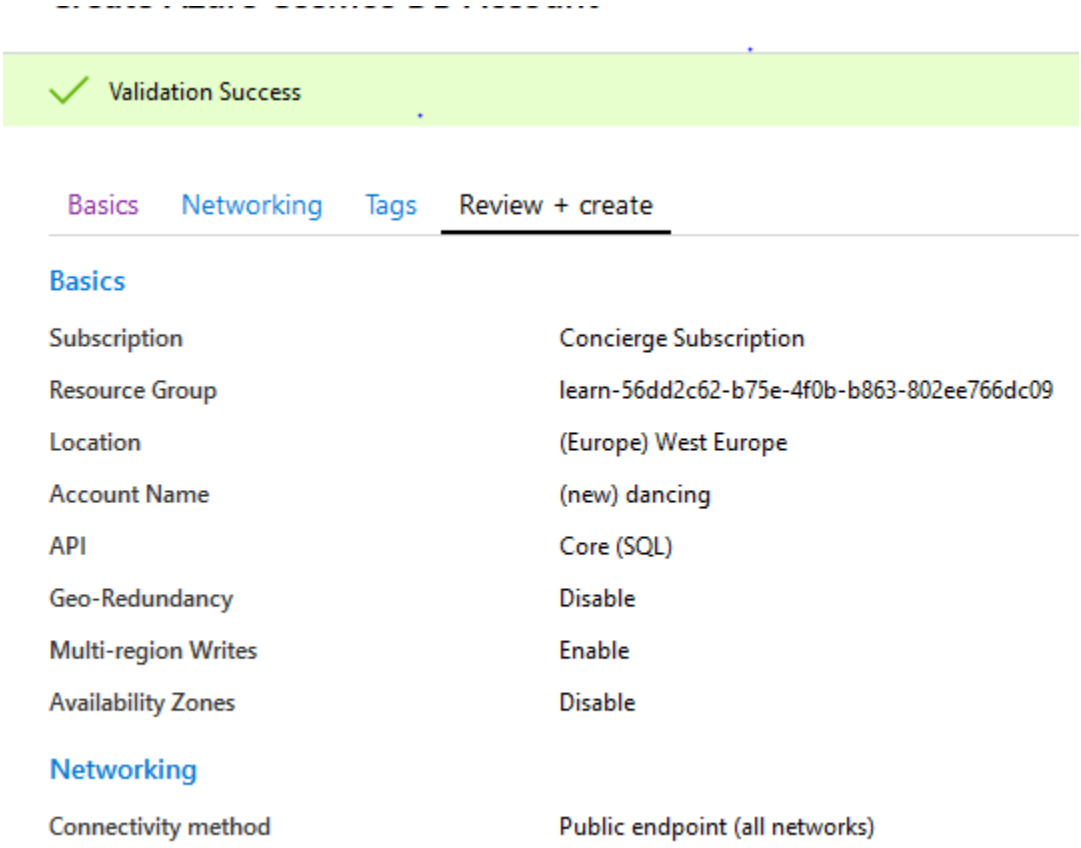
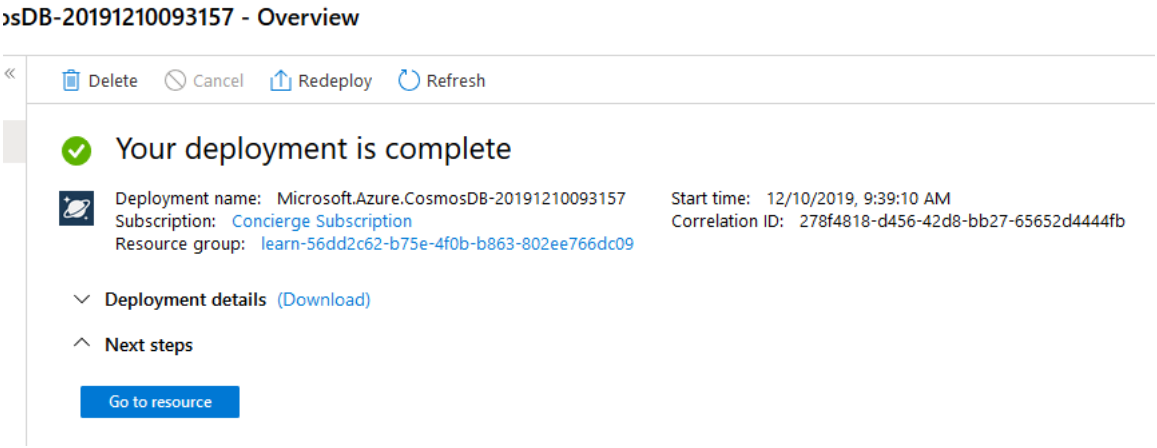
Laboratory hadoop ecosystem

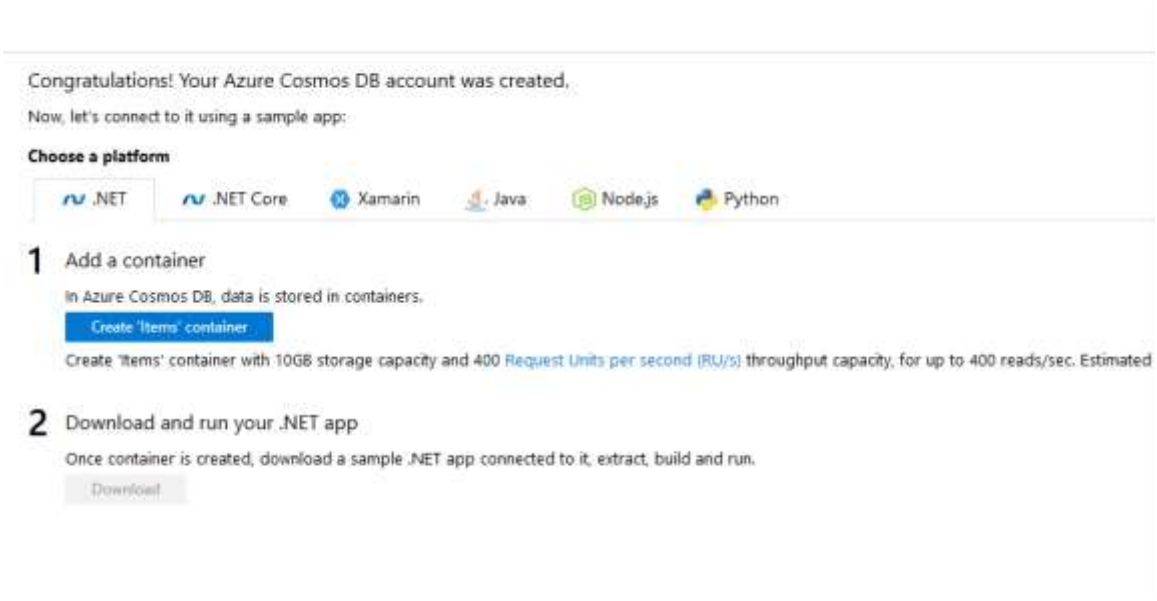
Section – Ankit Rathi , Uwacu Jean Remy

1. create Azure cosmos DB



Deploying and creating account





2. Setup your azure cosmos DB database and container

   a. create a new Azure Cosmos DB account with your specified name

```
rathi_ankit1998@Azure:~$ az cosmosdb create --name $NAME \
>     --kind GlobalDocumentDB \
>     --resource-group learn-e853bcce-eaa4-4e7b-aee0-e894bb4c770c

{
  "capabilities": [],
  "connectorOffer": null,
  "consistencyPolicy": {
    "defaultConsistencyLevel": "Session",
    "maxIntervalInSeconds": 5,
    "maxStalenessPrefix": 100
  },
  "databaseAccountOfferType": "Standard",
  "documentEndpoint": "https://cosmos28314.documents.azure.com:443/",
  "enableAutomaticFailover": false,
  "enableCassandraConnector": null,
  "enableMultipleWriteLocations": false,
  "failoverPolicies": [
    {
      "failoverPriority": 0,
      "id": "cosmos28314-westus",
      "locationName": "West US"
    }
  ],
  "id": "/subscriptions/b55cfb2d-2d7c-4db7-92d4-b98523213590/resourceGroups/learn-e853bcce-eaa4-4
0-e894bb4c770c/providers/Microsoft.DocumentDB/databaseAccounts/cosmos28314",
  "ipRangeFilter": "",
  "isVirtualNetworkFilterEnabled": false,
```

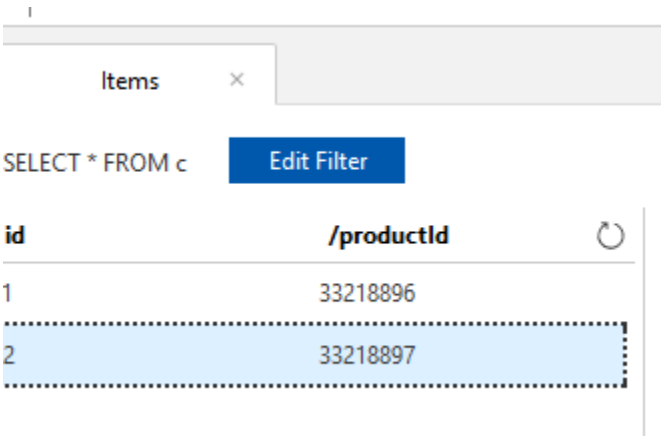
b. Create the Products database in the account using the cosmosdb database create command

```
rathi_ankit1998@Azure:~$ az cosmosdb database create --name $NAME \
>     --db-name "Products" \
>     --resource-group learn-e853bcce-eaa4-4e7b-aee0-e894bb4c770c
This command has been deprecated and will be removed in a future release
smosdb mongodb database, cosmosdb cassandra keyspace or cosmosdb gremlin
This command is implicitly deprecated because command group 'cosmosdb da
e removed in a future release. Use 'cosmosdb sql database, cosmosdb mong
 keyspace or cosmosdb gremlin database' instead.
{
  "_colls": "colls/",
  "_etag": "\"0000a303-0000-0700-0000-5def68ee0000\"",
  "_rid": "bH1eAA==",
  "_self": "dbs/bH1eAA==/",
```

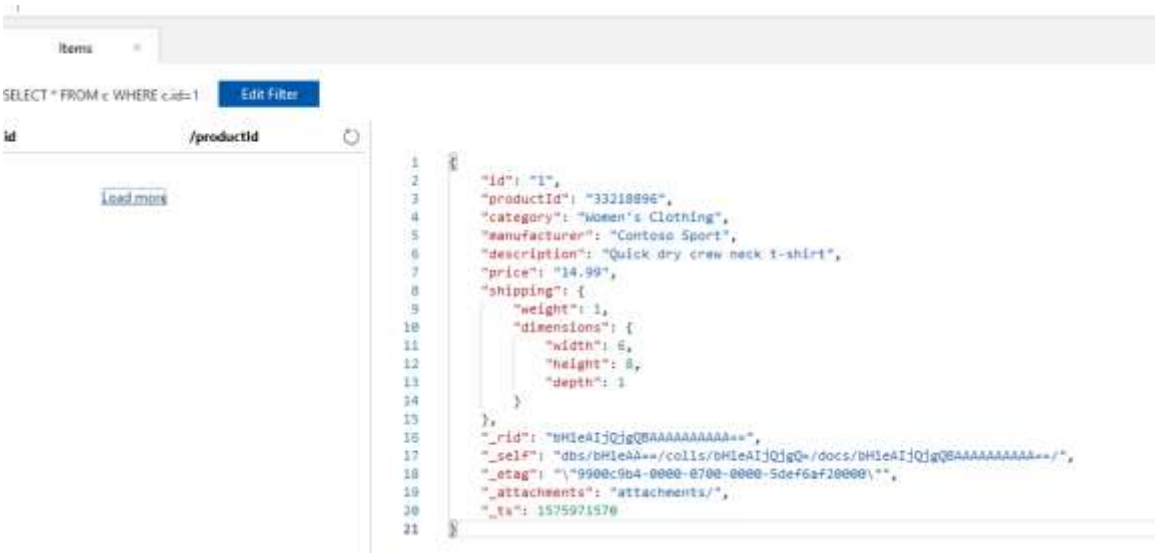
c. create the Clothing container with the cosmosdb collection create command in the Cloud Shell

```
>      --name $NAME \
>      --db-name "Products" \
>      --collection-name "Clothing" \
>      --partition-key-path "/productId" \
>      --throughput 1000 \
>      --resource-group learn-e853bcce-eaa4-4e7b-aee0-e894bb4c
This command has been deprecated and will be removed in a fut
osmosdb mongodb collection, cosmosdb cassandra table, cosmosc
This command is implicitly deprecated because command group '
 be removed in a future release. Use 'cosmosdb sql container,
andra table, cosmosdb gremlin graph or cosmosdb table' instea
{
  "collection": {
    "_conflicts": "conflicts/",
    "_docs": "docs/",
    "_etag": "\"0000a503-0000-0700-0000-5def693f0000\"",
    "_rid": "bH1eAIjQjgQ=",
    "_self": "dbs/bH1eAA==/colls/bH1eAIjQjgQ=/",
    "_sprocs": "sprocs/",
    "_triggers": "triggers/",
    "_ts": 1575971135,
```

3. Add data using data explorer

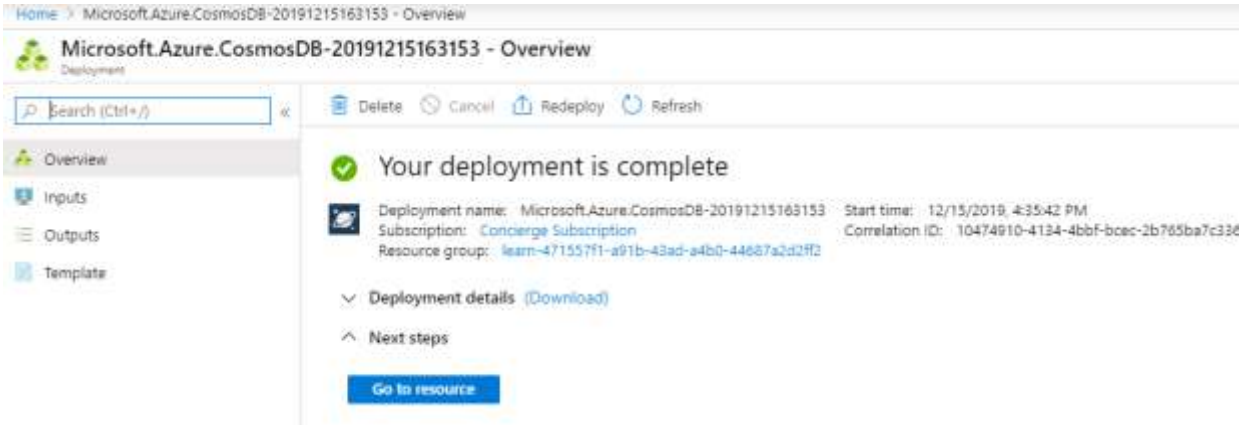


Explore SQL query types

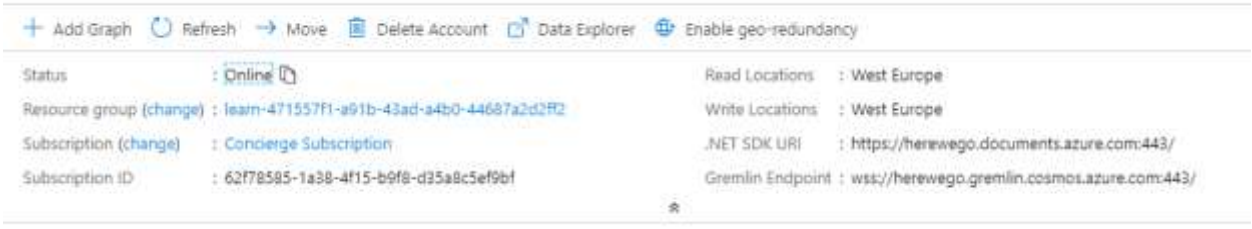


4. Storing and accessing graph data in Azure cosmos db using the graph API

a. As the initial step we will create Azure cosmos db account –



b. Add graph –



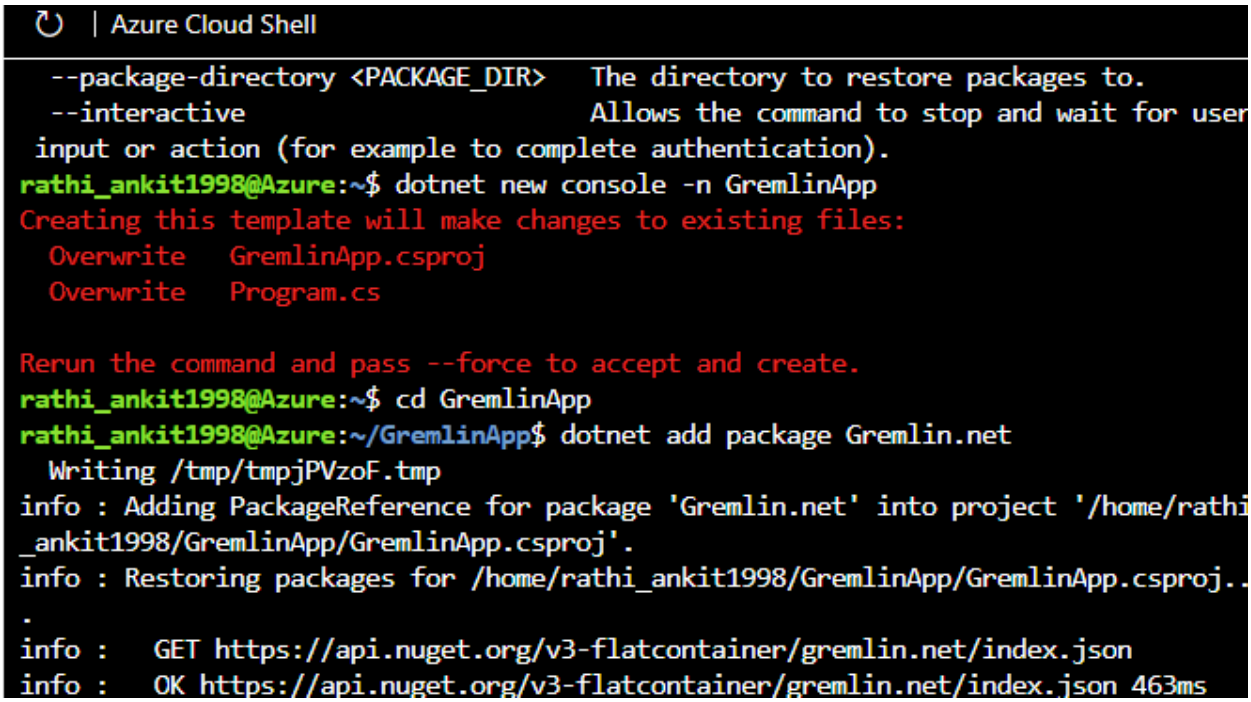
Germlin endpoint -- wss://herewego.gremlin.cosmos.azure.com:443/

Database id—sample-database

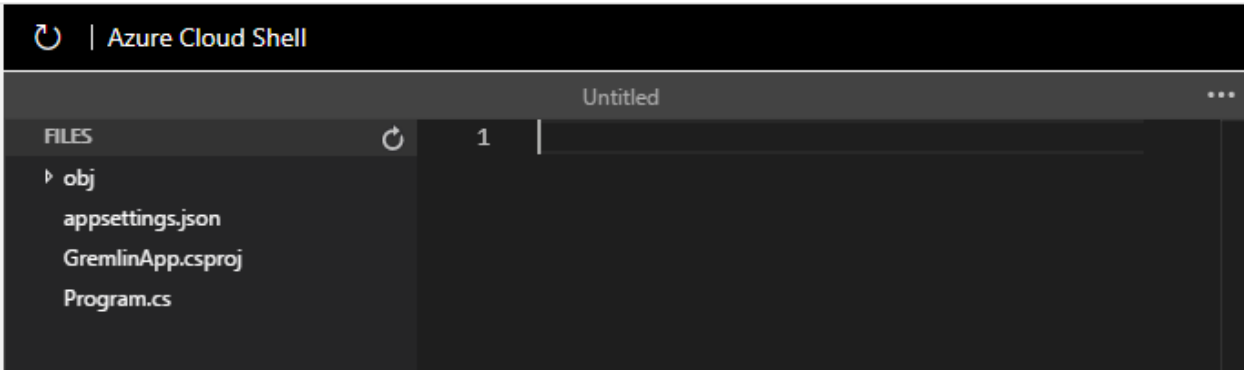
Graph id—sample-graph

Primary key --
ABjJNQC5NLCvr5TDwFRrFWzlTNSz4WeoymHZ32KImIVXEKizWI7BKB3mfH82LHVYsiyfgHiOBgah
NssbH8wQtQ==

5. Create a .net core app



a. Open app in online editor –



Run query with the app –

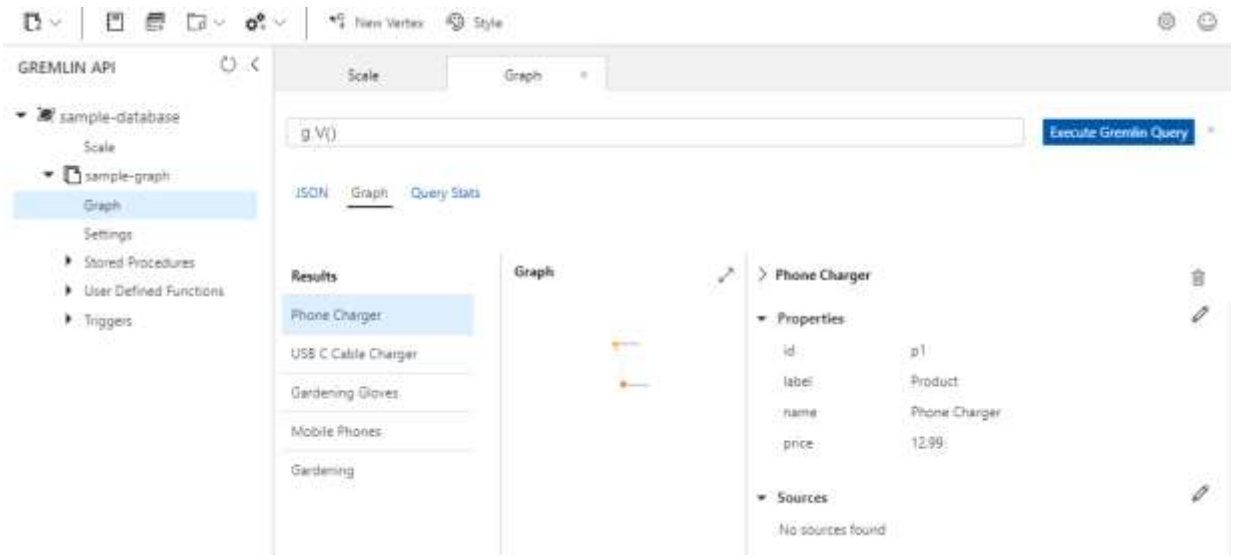
```
rathi_ankit1998@Azure:~/GremlinApp$ touch appsettings.json
rathi_ankit1998@Azure:~/GremlinApp$ code .
rathi_ankit1998@Azure:~/GremlinApp$ dotnet run
Please enter a Gremlin/Graph Query.
rathi_ankit1998@Azure:~/GremlinApp$ dotnet run "g.V().drop()"

{"Returned": "0"}
```

Add product nodes to the database and as result app indicates that we have 1 vertices was returned.

```
rathi_ankit1998@Azure:~/GremlinApp$ dotnet run "g.addV('Product').property('id
p1').property('name', 'Phone Charger').property('price', 12.99)"


{"Returned": "1"}
rathi_ankit1998@Azure:~/GremlinApp$ dotnet run "g.addV('Product').property('id
p2').property('name', 'USB C Cable Charger').property('price', 8.99)"

{"Returned": "1"}
```

```
rathi_ankit1998@Azure:~/GremlinApp$ dotnet run "g.V()"

{"Returned": "5"}
rathi_ankit1998@Azure:~/GremlinApp$
```

```
rathi_ankit1998@Azure:~/GremlinApp$ dotnet run "g.E()"

{"Returned": "3"}
rathi_ankit1998@Azure:~/GremlinApp$
```

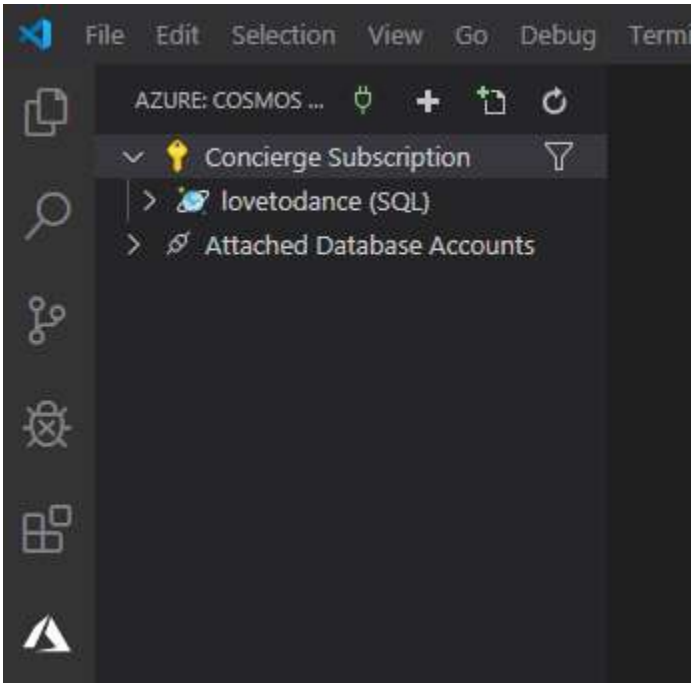


Run detailed query with the app –

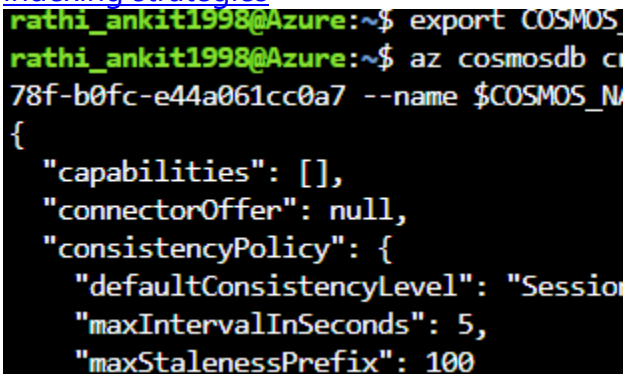
```
rathi_ankit1998@Azure:~/GremlinApp$ dotnet run "g.V()"

{"Returned": "5"}
{"id":"p1","label":"Product","type":"vertex","properties":{"name":[{"id":"p1|name"
,"value":"Phone Charger"}],"price":[{"id":"96950f04-2a9d-475b-936e-b779976ddcc7","
value":12.99}]}}
{"id":"p2","label":"Product","type":"vertex","properties":{"name":[{"id":"p2|name"
,"value":"USB C Cable Charger"}],"price":[{"id":"6064d384-0973-4709-927f-486a644c6
0b6","value":8.99}]}}
{"id":"p3","label":"Product","type":"vertex","properties":{"name":[{"id":"p3|name"
,"value":"Gardening Gloves"}],"price":[{"id":"06e6a2f1-bfaa-4005-85be-27ddd1f034a2
","value":2.99}]}}
{"id":"c1","label":"Category","type":"vertex","properties":{"name":[{"id":"c1|name
","value":"Mobile Phones"}]}}
{"id":"c2","label":"Category","type":"vertex","properties":{"name":[{"id":"c2|name
```

```
rathi_ankit1998@Azure:~/GremlinApp$ dotnet run "g.V().hasLabel('Cate
'name')"

{"Returned": "2"}
"Mobile Phones"
"Gardening"
rathi_ankit1998@Azure:~/GremlinApp$ dotnet run "g.V().hasLabel('Prod
name','price')"

{"Returned": "6"}
"Phone Charger"
12.99
"USB C Cable Charger"
8.99
"Gardening Gloves"
2.99
rathi_ankit1998@Azure:~/GremlinApp$
```

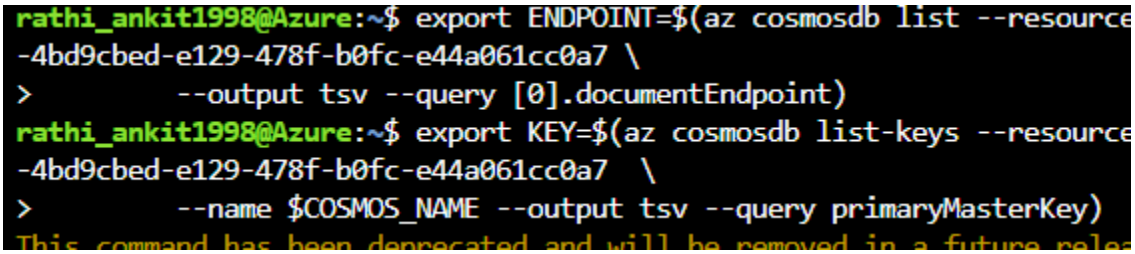
Visual studio sign in --- http://127.0.0.1:51748/



- Optimize the performance of Azure Cosmos DB by using partitioning and indexing strategies



Set environment variables for endpoint and keys



Populate your collections –

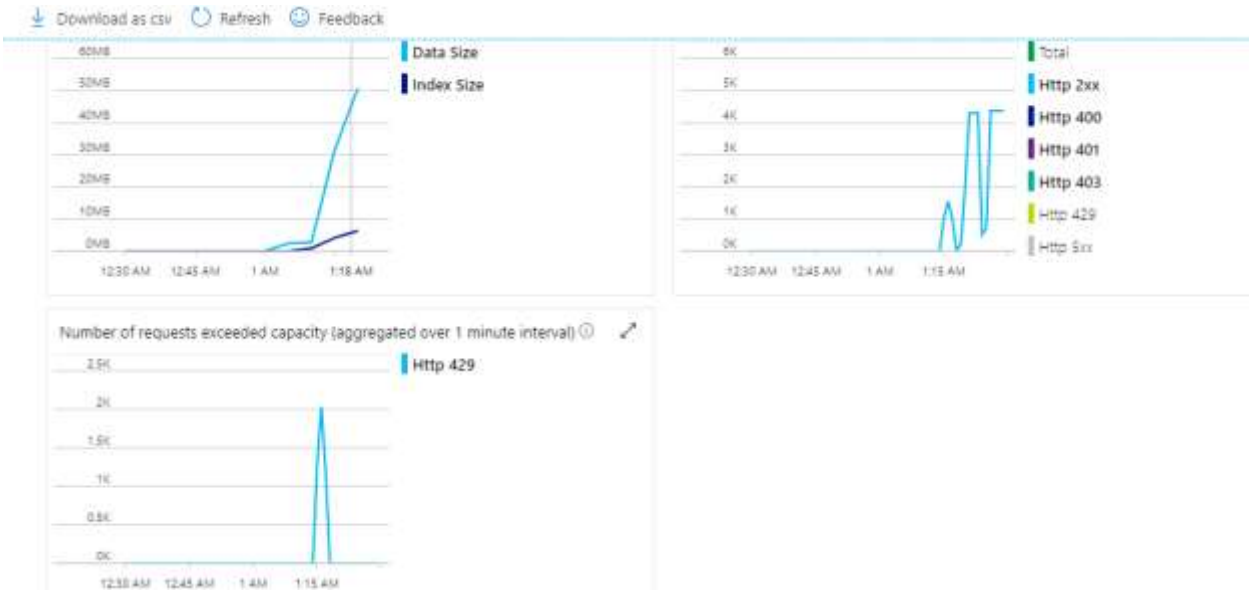
```
rathi_ankit1998@Azure:~$ git clone https://github.com/MicrosoftDocs/mslearn-m
r-azure-cosmos-db
Cloning into 'mslearn-monitor-azure-cosmos-db'...
remote: Enumerating objects: 67, done.
remote: Total 67 (delta 0), reused 0 (delta 0), pack-reused 67
Unpacking objects: 100% (67/67), done.
Checking connectivity... done.
rathi_ankit1998@Azure:~$ cd mslearn-monitor-azure-cosmos-db/ExerciseCosmosDB
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseCosmosDB$
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseCosmosDB$ exp
OSMOS_NAME=$(az cosmosdb list --output tsv --query [0].name)
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseCosmosDB$ exp
NDPOINT=$(az cosmosdb list --resource-group learn-4bd9cbed-e129-478f-b0fc-e44
c0a7 \
>         --output tsv --query [0].documentEndpoint)
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseCosmosDB$ exp
EY=$(az cosmosdb list-keys --resource-group learn-4bd9cbed-e129-478f-b0fc-e44
c0a7  \
>         --name $COSMOS_NAME --output tsv --query primaryMasterKey)
```

```
Performed 3852 Write operations @ 26 operations/s, 411.1 RU/s)
Performed 3879 Write operations @ 26 operations/s, 411.2 RU/s)
Performed 3908 Write operations @ 26 operations/s, 411.5 RU/s)
Performed 3936 Write operations @ 26 operations/s, 411.7 RU/s)
Performed 3960 Write operations @ 26 operations/s, 411.5 RU/s)
Performed 3983 Write operations @ 26 operations/s, 411.2 RU/s)
Performed 3994 Write operations @ 26 operations/s, 409.7 RU/s)
Performed 3997 Write operations @ 26 operations/s, 407.3 RU/s)
```

```
Performed 19809 Write operations @ 71 operations/s, 1130 RU/s)
Performed 19879 Write operations @ 71 operations/s, 1129.9 RU/s)
Performed 19952 Write operations @ 71 operations/s, 1130 RU/s)
Performed 19999 Write operations @ 71 operations/s, 1128.6 RU/s)
Performed 20000 Write operations @ 71 operations/s, 1124.7 RU/s)

---------------------------------------------------------------
Performed 20000 Write operations @ 71 operations/s, 1124.7 RU/s)
Total (consumed 316201.3 RUs in 281 seconds)
---------------------------------------------------------------
```

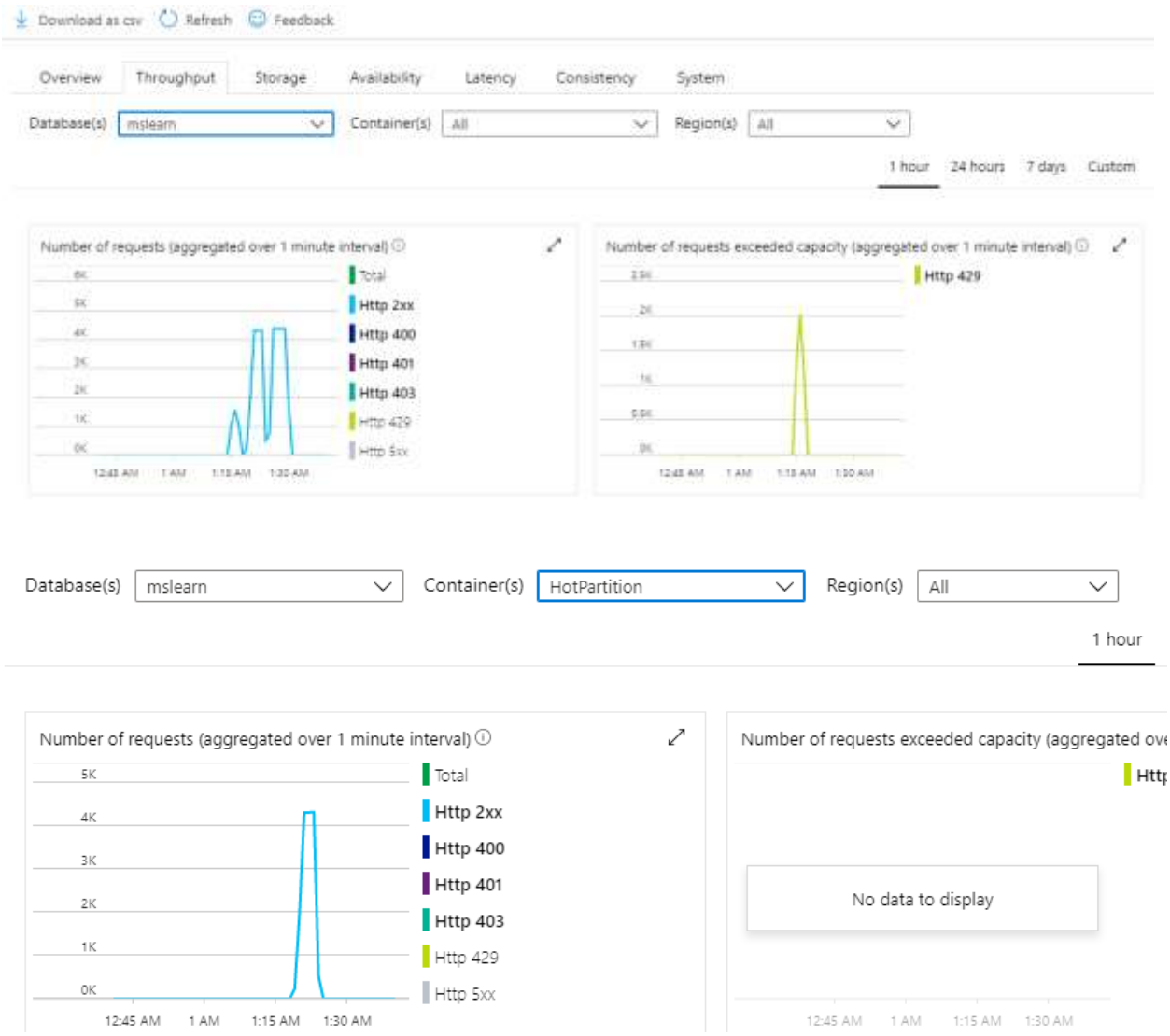
```
Performed 19801 Write operations @ 72 operations/s, 1145.5 RU/s)
Performed 19878 Write operations @ 72 operations/s, 1145.8 RU/s)
Performed 19950 Write operations @ 72 operations/s, 1145.8 RU/s)
Performed 20000 Write operations @ 72 operations/s, 1144.5 RU/s)

---------------------------------------------------------------
Performed 20000 Write operations @ 72 operations/s, 1144.5 RU/s)
Total (consumed 316200 RUs in 276 seconds)
```

Measuring throughputs in azure cosmos DB



```
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseC
NDPOINT=$(az cosmosdb list --resource-group learn-4bd9cbed-e129-4
c0a7 \
>         --output tsv --query [0].documentEndpoint)
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseC
EY=$(az cosmosdb list-keys --resource-group learn-4bd9cbed-e129-4
c0a7  \
>         --name $COSMOS_NAME --output tsv --query primaryMasterK
This command has been deprecated and will be removed in a future
mosdb keys list' instead.
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseC
rn-monitor-azure-cosmos-db/ExerciseCosmosDB
bash: cd: mslearn-monitor-azure-cosmos-db/ExerciseCosmosDB: No su
ory
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseC
un -- -c Orders -o InsertDocument -n 1
Setting up experiment...
Starting experiment with 1 tasks @ 12/18/19 12:41:00 AM
```

```
","Data":"5YFvPRdWc4zsxg==","_rid":"doBFAKxb1QABAAAAAAAAAA==","_self":"dbs/doBFAA
=/colls/doBFAKxb1QA=/docs/doBFAKxb1QABAAAAAAAAAA==/","_etag":"\"4f00e1a6-0000-000
-0000-5c2d266a0000\"","_attachments":"attachments/","_ts":1546462826}
bash: Order: command not found
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseCosmosDB$     Per
ormed 1 Query operations @ 1 operations/s, 4.6 RU/s (0B max monthly 1KB reads)
bash: syntax error near unexpected token `('
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseCosmosDB$ dotnet
un -- -o ReadDocument -l dbs/mslearn/colls/Orders/docs/<Document id value> -k <It
m id value> -n 1
bash: Document: No such file or directory
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseCosmosDB$ Perform
d 1 Query operations @ 1 operations/s, 1 RU/s (0B max monthly 1KB reads)
bash: syntax error near unexpected token `('
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseCosmosDB$
```



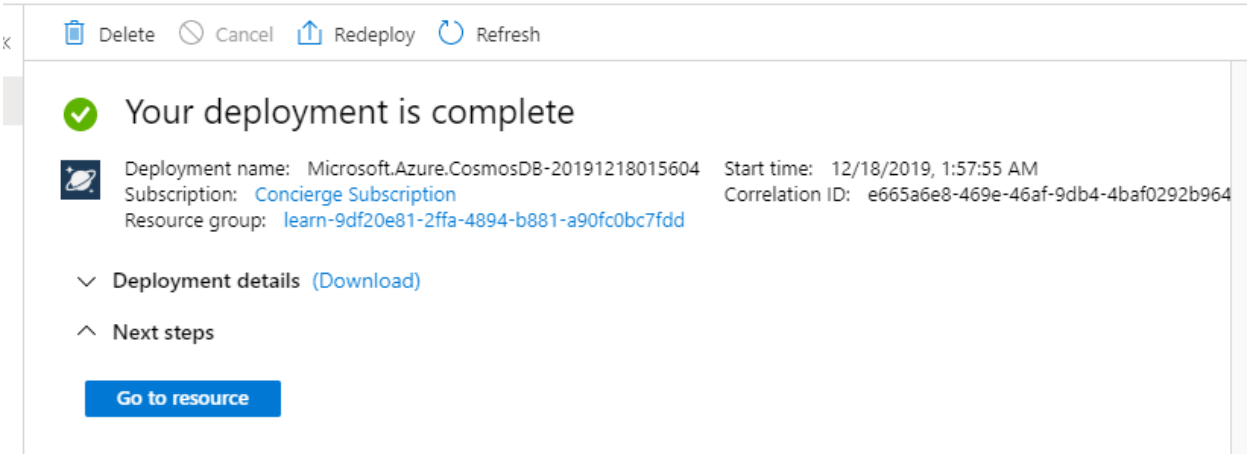
Measure the effect of index policy change—

```
: UZABIZKtF5V/XA== }
Performed 1 Write operations @ 0 operations/s, 2.5 RU/s)

---------------------------------------------------------------
Performed 1 Write operations @ 0 operations/s, 2.5 RU/s)
Total (consumed 5 RUs in 2 seconds)
---------------------------------------------------------------
CosmosDB experiment complete
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseCosmosDB$ dotne
un -- -c Orders -o QueryCollection -q "SELECT TOP 1 * FROM c WHERE c.Item.id='<
m id value>'"
Starting experiment with 1 tasks @ 12/18/19 12:49:14 AM
Performed 1 Query operations @ 1 operations/s, 256.3 RU/s)

---------------------------------------------------------------
Performed 1 Query operations @ 1 operations/s, 255.5 RU/s)
Total (consumed 256.9 RUs in 1 seconds)
```

Measure RUs for a partial index

```
Performed 1 Query operations @ 0 operations/s, 2 RU/s)

---------------------------------------------------------------
Performed 1 Query operations @ 0 operations/s, 2 RU/s)
Total (consumed 5.9 RUs in 3 seconds)
---------------------------------------------------------------
CosmosDB experiment complete
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseCosmosDB$ dotnet
un -- -c Orders -o QueryCollection -q "SELECT TOP 1 * FROM c WHERE c.Customer.Sta
e = 'WA'"
Starting experiment with 1 tasks @ 12/18/19 12:50:57 AM
Performed 0 Query operations @ 0 operations/s, 0 RU/s)
Performed 0 Query operations @ 0 operations/s, 0 RU/s)
Performed 1 Query operations @ 0 operations/s, 3 RU/s)

---------------------------------------------------------------
Performed 1 Query operations @ 0 operations/s, 3 RU/s)
Total (consumed 9.1 RUs in 3 seconds)
---------------------------------------------------------------
CosmosDB experiment complete
rathi_ankit1998@Azure:~/mslearn-monitor-azure-cosmos-db/ExerciseCosmosDB$
```

Distribute data globally with azure cosmos db –



Replicate data in multiple region –