(1)

(2)

Perfomance:
-Standard
Redundancy:
-Locally redundand starage(LRS)
DLSGen2:
-Enable hoerarchical namespace

Upload

(10)

create new container(input)

(9)

(3)

Manage: (11)
-Linked services-
+New-
ADLSGen2(adlsLS):
Authentication type:
Account key

Manage: (13)
-Linked services-
+New-
ASQLD(asqldLS)
Autentication type:
-SQL autentication

Author-Datasets-New Data   (14)
AZDLSGen2-CSV-
Linked service:
adlsLS-File path: input

Datasets-New Dataset-   (15)
ASQLD-Linked service:
Linked service:
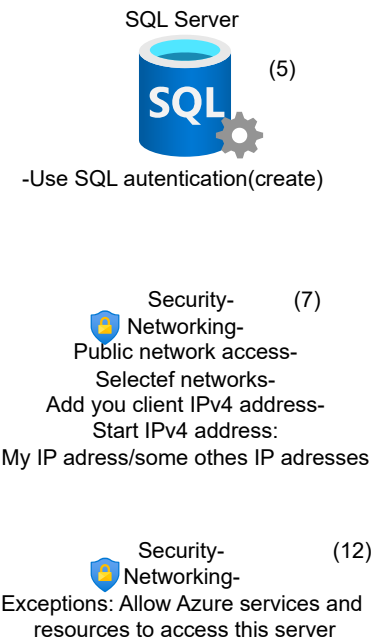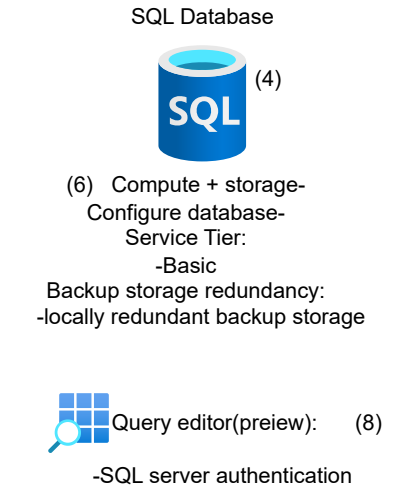asqldLS
Connection-Table:
dbo   movies
Publish all

Author- Data flows-New data flow:
Name:(Transform)

Source settings:
-Output stream name:(source1)
-Dataset:+New
ADLSGen2-CSV
Name:(Source)
Linked service:adlsLS
File path: input

-Data flow debug:on
-Test connection
Data preview:

SQL Server

(5)

-Use SQL autentication(create)

Security-   (7)
Networking-
Public network access-
Selectef networks-
Add you client IPv4 address-
Start IPv4 address:
My IP adress/some othes IP adresses

Security-   (12)
Networking-
Exceptions: Allow Azure services and
resources to access this server

+
filter
Filter settings:
Filter on-Open expression builder:
toInteger(year)>=1910 &&
toInteger(year)<2000 &&
rlike(genres, 'Comedy')
Data preview:

SQL Database

(4)

(6)   Compute + storage-
Configure database-
Service Tier:
-Basic
Backup storage redundancy:
-locally redundant backup storage

Query editor(preiew):   (8)
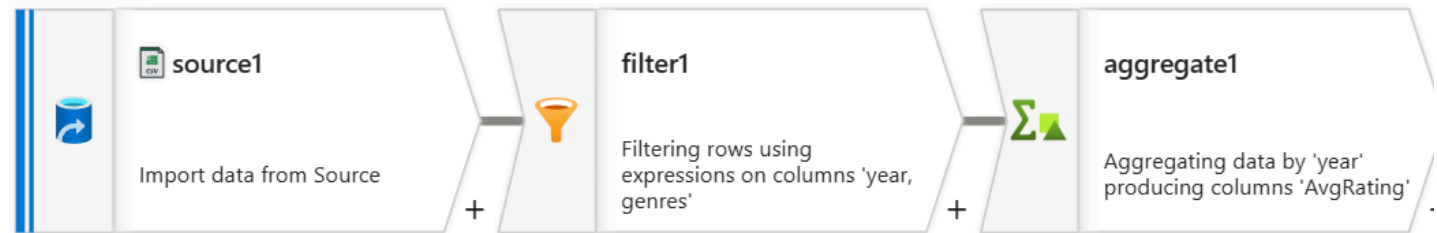
-SQL server authentication

+aggregate
Aggregate settings:
-Group by:year
-Column:year
-Aggregates:
-Column(AvgRating)
-Expression-Open expression builder:
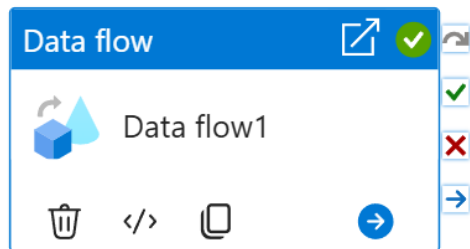avg(toInteger(Rating))

+sink
-New-ASQLD:
Name:(Sink)
Linked service:asqldLS

Data preview.

source1	filter1	aggregate1
📄 source1	🔽 filter1	Σ aggregate1
Import data from Source	Filtering rows using expressions on columns 'year, genres'	Aggregating data by 'year' producing columns 'AvgRating'

Pipelines-new pipeline
Activities:Dataflow
Debug

Data flow

Data flow1

🗑 </> 📋 ➡

From Dataflow:
Add trigger-New
Type:Starage Event
Blob path ends with:(.csv)
Event-Blob created
Create-Continue-OK-Publish

Monitoring-Trigger runs-Refresh

In Azure-input-Upload-its start if its csv-file

Trigger runs-Refresh
Pipeline runs

Azure-resource group-SQL database-Query editor
Tables:dbj.movies
Select Top 1000 Rows

```
DROP TABLE [dbo].[movies]
            +
"delete movieDB.csv from input in blob"
```

sink1

Export data to Sink