

## P3: Implementation

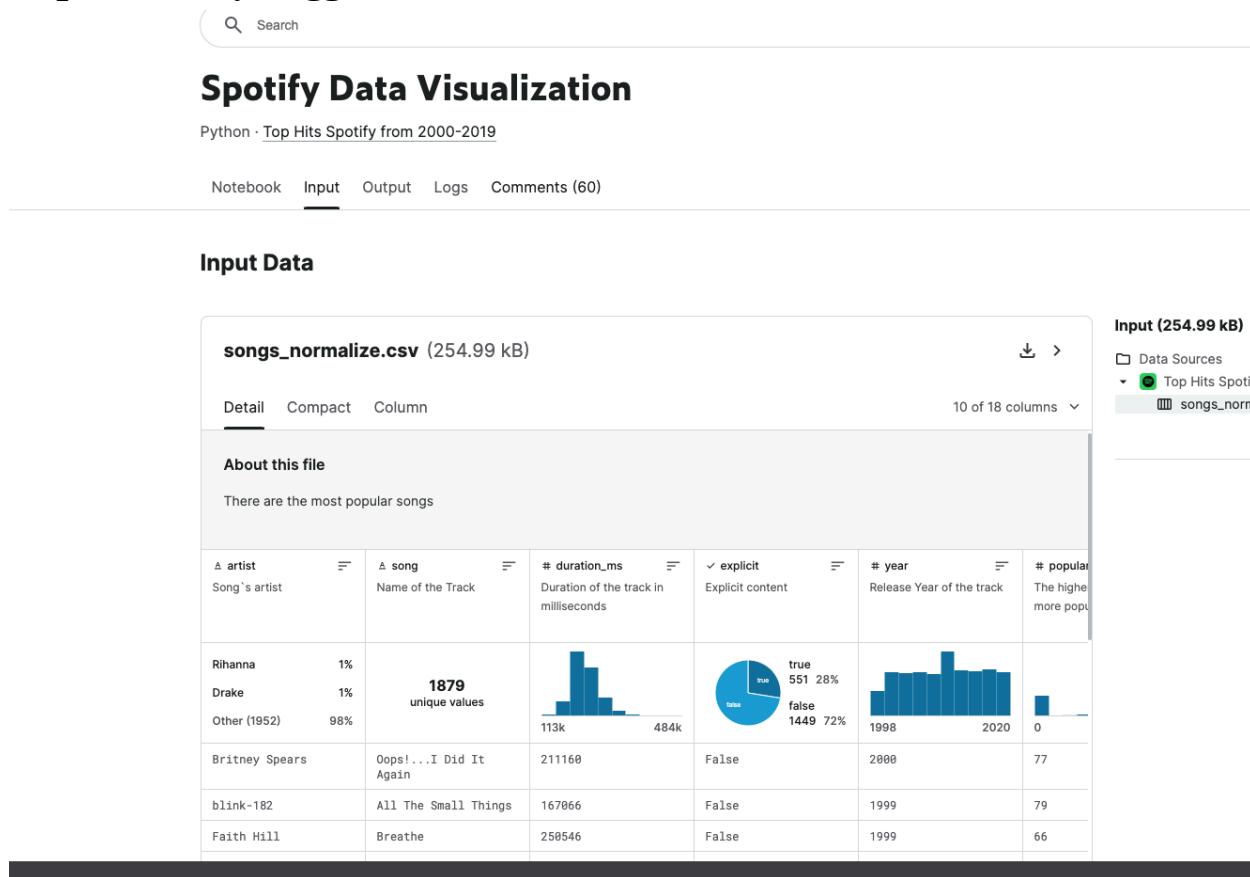
**Description of Implementation Process:** Our project began with the selection of a comprehensive dataset from Kaggle, containing information on artists, songs, and their audio features. We utilized Python to preprocess this data, ensuring its quality by removing null values and duplicates.

Following data preprocessing, we set up an Azure environment by creating a resource group and an Azure Blob Storage account. Within this storage account, we established a container to securely store all our CSV files.

Next, we linked Azure Data Factory and Azure SQL Database to our storage account. Using Azure Data Factory, we created data pipelines to efficiently transfer the CSV files into our destination database.

Finally, we utilized Azure Data Studio to work on our graph and document databases, enabling advanced data visualization and management capabilities. This comprehensive approach ensured a seamless integration and robust data handling process.

### Step 1: Identify Kaggle Dataset



The screenshot shows a Kaggle dataset visualization for 'songs\_normalize.csv'. The main title is 'Spotify Data Visualization' with a subtitle 'Python · Top Hits Spotify from 2000-2019'. Below the title are tabs for Notebook, Input (selected), Output, Logs, and Comments (60). The 'Input' tab displays the file 'songs\_normalize.csv' (254.99 kB) with 1879 unique values. It includes a summary table and several data visualizations: a histogram of duration\_ms, a pie chart of explicit content (true: 28%, false: 72%), and a bar chart of release year (1998, 2000, 2020). The right sidebar shows the input file details and a data sources section for 'Top Hits Spotify'.

### Step 2: Python Preprocessing

```
In [13]: import pandas as pd
```

```
In [14]: df = pd.read_csv(r"C:\Users\kumar\User.csv")
```

```
In [15]: df
```

```
Out[15]:
```

	UserId	UserName	Email	Gender	ListeningDevice	Region	Followers	SubscriptionId	PaymentId
0	1	Kimbell Blesdill	kblesdill0@shareasale.com	Male	Windows PC	MV-01	125525988.0	5	1
1	2	Leo Yarrall	lyarrall1@jiathis.com	Male	Streaming Stick	CA-NT	129700749.0	5	4
2	3	Elisbeth Brickwood	ebrickwood2@ebay.com	Female	macOS Computers	BR-AM	121228683.0	5	5
3	4	Bambi Hollyman	bhollyman3@imageshack.us	Female	Smart TV	CG-12	11650272.0	4	2
4	5	Amos Beagrie	abeagrie4@ycombinator.com	Male	In-car infotainment system	BR-GO	48650848.0	1	1
...	...	...	...	...	...	...	...	...	...
995	996	Walker McRamsey	wmramseyrn@hc360.com	Male	Gaming Console	AU-WA	43342214.0	2	1
996	997	Adore Maggorini	amaggoriniro@ed.gov	Female	Tablet	PH-PLW	62727322.0	5	6
997	998	Jacki Kydde	jkydderp@time.com	Female	macOS Computers	DZ-11	44775312.0	5	4
998	999	Cassie Shatliffe	cshatlifferq@bloomberg.com	Female	Amazon Echo (Alexa)	AR-R	13299615.0	5	2
999	1000	Marmaduke Letterick	mletterickrr@surveymonkey.com	Male	Smartphone	US-CA	34009230.0	3	2

1000 rows × 9 columns

```
In [16]: newdf = df.dropna()
```

```
In [17]: newdf
```

```
Out[17]:
```

	UserId	UserName	Email	Gender	ListeningDevice	Region	Followers	SubscriptionId	PaymentId
0	1	Kimbell Blesdill	kblesdill0@shareasale.com	Male	Windows PC	MV-01	125525988.0	5	1
1	2	Leo Yarrall	lyarrall1@jiathis.com	Male	Streaming Stick	CA-NT	129700749.0	5	4
2	3	Elisbeth Brickwood	ebrickwood2@ebay.com	Female	macOS Computers	BR-AM	121228683.0	5	5
3	4	Bambi Hollyman	bhollyman3@imageshack.us	Female	Smart TV	CG-12	11650272.0	4	2
4	5	Amos Beagrie	abeagrie4@ycombinator.com	Male	In-car infotainment system	BR-GO	48650848.0	1	1
...	...	...	...	...	...	...	...	...	...
995	996	Walker McRamsey	wmramseyrn@hc360.com	Male	Gaming Console	AU-WA	43342214.0	2	1
996	997	Adore Maggorini	amaggoriniro@ed.gov	Female	Tablet	PH-PLW	62727322.0	5	6
997	998	Jacki Kydde	jkydderp@time.com	Female	macOS Computers	DZ-11	44775312.0	5	4
998	999	Cassie Shatliffe	cshatlifferq@bloomberg.com	Female	Amazon Echo (Alexa)	AR-R	13299615.0	5	2
999	1000	Marmaduke Letterick	mletterickrr@surveymonkey.com	Male	Smartphone	US-CA	34009230.0	3	2

997 rows × 9 columns

```
In [19]: newdf.drop_duplicates()
```

```
Out[19]:
```

	UserId	UserName	Email	Gender	ListeningDevice	Region	Followers	SubscriptionId	PaymentId
0	1	Kimbell Blesdill	kblesdill0@shareasale.com	Male	Windows PC	MV-01	125525988.0	5	1
1	2	Leo Yarrall	lyarrall1@jiathis.com	Male	Streaming Stick	CA-NT	129700749.0	5	4
2	3	Elisbeth Brickwood	ebrickwood2@ebay.com	Female	macOS Computers	BR-AM	121228683.0	5	5
3	4	Bambi Hollyman	bhollyman3@imageshack.us	Female	Smart TV	CG-12	11650272.0	4	2
4	5	Amos Beagrie	abeagrie4@ycombinator.com	Male	In-car infotainment system	BR-GO	48650848.0	1	1
...	...	...	...	...	...	...	...	...	...
995	996	Walker McRamsey	wmramseyrn@hc360.com	Male	Gaming Console	AU-WA	43342214.0	2	1
996	997	Adore Maggorini	amaggoriniro@ed.gov	Female	Tablet	PH-PLW	62727322.0	5	6
997	998	Jacki Kydde	jkydderp@time.com	Female	macOS Computers	DZ-11	44775312.0	5	4
998	999	Cassie Shatliffe	cshatlifferq@bloomberg.com	Female	Amazon Echo (Alexa)	AR-R	13299615.0	5	2
999	1000	Marmaduke Letterick	mletterickrr@surveymonkey.com	Male	Smartphone	US-CA	34009230.0	3	2

997 rows × 9 columns

## Step 3: Create Dummy Data

The screenshot shows the Mockaroo schema configuration interface for a 'User' table. The table has nine fields:

- UserId**: Row Number, blank: 0 %, Σ
- UserName**: Full Name, blank: 0 %, Σ
- Email**: Email Address, blank: 0 %, Σ
- Gender**: Gender, blank: 0 %, Σ
- ListeningDevice**: Custom List, options: Smartphone, Tablet, Windows PC, macOS Computers, Amazon Echo (Alexa), Google Home (Google Assistant). Type dropdown: random, blank: 0 %, Σ
- Region**: Airport Region Code, blank: 0 %, Σ
- Followers**: Number, min: 1, max: 133000000, decimals: 0, blank: 0 %, Σ
- SubscriptionId**: Dataset Column, SubscriptionPlan dropdown, SubscriptionId dropdown, random dropdown, blank: 0 %, Σ
- PaymentId**: Dataset Column, random dropdown, blank: 0 %, Σ

Buttons at the bottom include '+ ADD ANOTHER FIELD' and 'GENERATE FIELDS USING AI...'

## Step 4: Create Resource Group

The screenshot shows the Microsoft Azure portal interface. The user is viewing the 'adbms\_rg' resource group details. The left sidebar shows other resource groups like 'Resource1'. The main pane displays the 'Overview' tab for 'adbms\_rg', which contains a table of resources:

Name	Type	Location
adbmsdata	Data factory (V2)	East US

Other tabs visible include 'Activity log', 'Access control (IAM)', 'Tags', 'Resource visualizer', and 'Events'. The bottom of the screen shows the Windows taskbar with various open applications.

## Step 5: Create Azure Blob Storage Account

The screenshot shows the Microsoft Azure portal interface. At the top, the URL is [portal.azure.com/#view/HubsExtension/DeploymentDetailsBlade/~/overview/id/%2Fsubscriptions%2F251d3e46-bc1c-48e8-b1e0-1af25bbc4068%2FresourceGroups%2Fadbms\\_rg%2FprovisioningStatus](https://portal.azure.com/#view/HubsExtension/DeploymentDetailsBlade/~/overview/id/%2Fsubscriptions%2F251d3e46-bc1c-48e8-b1e0-1af25bbc4068%2FresourceGroups%2Fadbms_rg%2FprovisioningStatus). The page title is "storageadbms\_1716425390810 | Overview". The main content area displays a green checkmark icon and the message "Your deployment is complete". Below this, it shows the deployment name "storageadbms\_1716425390810", subscription "Azure subscription 1", and resource group "adbms\_rg". It also shows the start time as "5/22/2024, 5:50:45 PM" and correlation ID "b7cd065d-0b55-4287-bcfb-910b017414de". On the right side, there are promotional cards for "Cost Management", "Microsoft Defender for Cloud", "Free Microsoft tutorials", and "Work with an expert". The bottom of the screen shows the Windows taskbar with various pinned icons like File Explorer, Downloads, Google Chrome, and Microsoft Word.

## Step 6: Create Database Server

The screenshot shows the Microsoft Azure portal interface. The URL is [portal.azure.com/#view/SqAzureExtension/SharedCreateServer/isNewServer~/true/subscriptionId/251d3e46-bc1c-48e8-b1e0-1af25bbc4068/resourceGroupName/adbms\\_rg/info](https://portal.azure.com/#view/SqAzureExtension/SharedCreateServer/isNewServer~/true/subscriptionId/251d3e46-bc1c-48e8-b1e0-1af25bbc4068/resourceGroupName/adbms_rg/info). The page title is "Create SQL Database Server". The main content area shows "Server details" with fields for "Server name" (set to "projectadbms") and "Location" (set to "(US) East US"). A note indicates that Azure Active Directory (Azure AD) is now Microsoft Entra ID. The "Authentication" section allows selecting preferred authentication methods. At the bottom, there is an "OK" button. The bottom of the screen shows the Windows taskbar with various pinned icons like File Explorer, Downloads, Google Chrome, and Microsoft Word.

## Step 7: Loaded the CSV Files into the Container

**Authentication method:** Access key ([Switch to Microsoft Entra user account](#))  
**Location:** spotifycontainer

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
Artist.csv	5/22/2024, 6:01:59 PM	Hot (inferred)		Block blob	563.2 KIB	Available
AudioFeatures.csv	5/22/2024, 6:02:00 PM	Hot (inferred)		Block blob	7.25 MiB	Available
PaymentMethod.csv	5/22/2024, 6:01:58 PM	Hot (inferred)		Block blob	146 B	Available
Playlist.csv	5/22/2024, 6:02:10 PM	Hot (inferred)		Block blob	51.42 MiB	Available
Song.csv	5/22/2024, 6:01:59 PM	Hot (inferred)		Block blob	9.5 MiB	Available
SubscriptionPlan.csv	5/22/2024, 6:01:58 PM	Hot (inferred)		Block blob	162 B	Available
User.csv	5/22/2024, 6:01:59 PM	Hot (inferred)		Block blob	83.42 KIB	Available

## Step 8: Created Azure Data Factory

**Your deployment is complete**

Deployment name : Microsoft.DataFactory-20240522173503  
Subscription : Azure subscription 1  
Resource group : adbms\_rg

Start time : 5/22/2024, 5:36:58 PM  
Correlation ID : 0f64dbf3-fb4e-423d-b9b6-b9a1e55efb9e

**Deployment details**

**Next steps**

[Go to resource](#)

**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 >](#)

## Step 9: Created a link to the Azure Data Factory

The screenshot shows the Microsoft Azure Data Factory interface. At the top, there are two informational banners: one about Microsoft Fabric and another about Azure Data Factory's Git repository support. Below these, the main dashboard for the 'adbmsdata' factory is displayed. It features a large central image of a factory building with pipes and containers, surrounded by four boxes: 'Ingest' (Copy data at scale once or on a schedule), 'Orchestrate' (Code-free data pipelines), 'Transform data' (Transform your data using data flows), and 'Configure SSIS' (Manage & run your SSIS packages in the cloud). To the left, there's a sidebar with icons for Home, New, and Recent resources. The taskbar at the bottom shows various open windows and system status.

## Step 10: Created a link to the Azure Data SQL

The screenshot shows the Microsoft Azure Data Factory interface, specifically the 'Factory Resources' section for the 'Artist' dataset. On the left, the 'Datasets' section is expanded, showing 'Artist' and 'AudioFeatures'. In the center, the 'Connection' tab is selected for the 'Artist' dataset. The 'Linked service' dropdown is set to 'adbms8S'. The 'File path' field contains 'spotifycontainer'. A 'Browse' window is open on the right, showing a list of CSV files in a folder structure: Root folder > spotifycontainer. The file 'PaymentMethod.csv' is currently selected. The taskbar at the bottom shows various open windows and system status.

## Step 11: Created a Data Pipeline

The screenshot shows the Microsoft Azure Data Factory Data Pipeline interface. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (1), 'DataPipeline' (selected), 'Datasets' (7), 'Artist', 'AudioFeatures', 'PaymentMethod', 'Playlist', 'Song', 'SubscriptionPlan', 'User', 'Data flows' (1), 'dataflow1', and 'Power Query'. The main workspace displays a complex data flow with several 'Copy data' activities. One activity copies from 'ArtistTable' to 'SongTable'. Another activity copies from 'SongTable' to 'ArtistTable'. Other activities include 'Copy data' from 'SongTable' to 'SubscriptionPlanTable', 'Copy data' from 'SongTable' to 'PaymentMethodTable', and 'Copy data' from 'SongTable' to 'UserTable'. The 'Properties' pane on the right shows the pipeline is named 'DataPipeline' with no description. The 'Output' section shows the following table:

Activity name	Activity status	Activity type	Run start
PlaylistTable	Succeeded	Copy data	5/22/2024, 6:41:21
SongTable	Succeeded	Copy data	5/22/2024, 6:41:21
ArtistTable	Succeeded	Copy data	5/22/2024, 6:41:21
SubscriptionPlanTable	Succeeded	Copy data	5/22/2024, 6:41:21
PaymentMethodTable	Succeeded	Copy data	5/22/2024, 6:41:21

## Step 12: Created and Inserted data in Document Database in Azure Data Studio

The screenshot shows Microsoft SQL Server Management Studio (SSMS) with a query window titled 'SQLQuery\_4.sql - projectadbms.database.windows.net.SpotifyDB (spotify (52))'. The query is a complex T-SQL script using FOR JSON PATH to insert data into a document database. The script joins multiple tables (Artist, Song, AudioFeatures, PaymentMethod, User) and inserts the results into a document database table. The 'Results' tab at the bottom shows the execution status: 'Query executed successfully.' and '1 rows'.

```

SELECT
    s.track_name AS [Song],
    (
        SELECT
            ar.artistNames AS [ArtistName]
        FROM artist ar
        JOIN doc.tlSongSong ars ON ar.artistid = ars.artistid
        WHERE ars.songid = s.songid
        FOR JSON PATH
    ) AS [Artist],
    (
        SELECT
            af.danceability,
            af.energy,
            af.key,
            af.liveness,
            af.mode,
            af.speechiness,
            af.acousticness,
            af.instrumentalness,
            af.valence,
            af.tempo
        FROM doc.tlSongAudioFeatures af
        JOIN doc.tlSongAudioFeature saf ON af.audio_id = saf.audio_id
        WHERE saf.songid = s.songid
        FOR JSON PATH
    ) AS [Audiofeature]
    FROM doc.tlSong s
    FOR JSON PATH, ROOT('Songs');

```

C:\Users\prach\OneDrive\Desktop\UW\SpringCourses\S63\document.json - Notepad++

```

1  {
2     "Songs": [
3         {
4             "Song": "Ghost - Acoustic",
5             "Artist": [
6                 {
7                     "ArtistName": "Third Eye Blind"
8                 }
9             ],
10            "AudioFeature": [
11                {
12                    "danceability": 6.17000000000000e-001,
13                    "energy": 3.19000000000000e-001,
14                    "loudness": -8.44500000000000e+000,
15                    "mode": 1,
16                    "speechiness": 2.90000000000000e-002,
17                    "acousticness": 6.09000000000000e-001,
18                    "instrumentalness": 0.00000000000000e+000,
19                    "liveness": 1.22000000000000e-001,
20                    "valence": 3.24000000000000e-001,
21                    "tempo": 1.33821000000000e+002
22                }
23            ],
24        },
25        {
26            "Song": "To Begin Again",
27            "Artist": [
28                {
29                    "ArtistName": "A Great Big World"
30                }
31            ],
32            "AudioFeature": [
33                {
34                    "danceability": 6.78000000000001e-001,
35                    "energy": 3.51000000000000e-001,
36                    "loudness": -8.65400000000000e+000,
37                    "mode": 1,
38                    "speechiness": 2.66000000000000e-002,
39                    "acousticness": 7.47000000000000e-001,
40                    "instrumentalness": 0.00000000000000e+000,
41                    "liveness": 3.55000000000000e-001,
42                    "valence": 5.69000000000000e-001,
43                    "tempo": 9.00320000000000e+001
44                }
45            ],
46        },
47        {
48            "Song": "Can't Help Falling In Love",
49            "AudioFeature": [
50
51        ]
52    ]
53}

```

JSON file length: 804,390 lines: 20,548 Ln: 20,548 Col: 2 Pos: 804,391 Windows (CR LF) UTF-8 INS

Document.sql - projectadbsms.database.windows.net.SpotifyDB (spotify (69)) - Microsoft SQL Server Management Studio

```

1 | CREATE TABLE tb1DocArtist (
2 |     Artists NVARCHAR(MAX)
3 | );
4 |
5 | CREATE TABLE tb1DocSong (
6 |     Songs NVARCHAR(MAX)
7 | );
8 |
9 | CREATE TABLE tb1DocAudiofeatures (
10 |     AudioFeatures NVARCHAR(MAX)
11 | );
12 |
13 |
14 Insert Into tb1DocArtist (Artists)
15 Values(N'"Artist_ID_1","ArtistNames":"Bon Iver"'),
16 (N'"Artist_ID_2","ArtistNames":"Ben Howard"'),
17 (N'"Artist_ID_3","ArtistNames":"Ingrid Michaelson"'),
18 (N'"Artist_ID_4","ArtistNames":"ZAYN"'),
19 (N'"Artist_ID_5","ArtistNames":"Kina Grannis"'),
20 (N'"Artist_ID_6","ArtistNames":"Lana Del Rey"'),
21 (N'"Artist_ID_7","ArtistNames":"Troye Sivan"'),
22 (N'"Artist_ID_8","ArtistNames":"A Great Big World"'),
23 (N'"Artist_ID_9","ArtistNames":"Christina Aguilera"'),
24 (N'"Artist_ID_10","ArtistNames":"Jessie J"'),
25 (N'"Artist_ID_11","ArtistNames":"Kanye West"'),
26 (N'"Artist_ID_12","ArtistNames":"Bazzi"'),
27 (N'"Artist_ID_13","ArtistNames":"Ross Copperman"'),
28 (N'"Artist_ID_14","ArtistNames":"Zack Tabubilo"'),
29 (N'"Artist_ID_15","ArtistNames":"Dan Berk"'),
30 (N'"Artist_ID_16","ArtistNames":"Hozier"'),
31 (N'"Artist_ID_17","ArtistNames":"Deepend")'),
32 (N'"Artist_ID_18","ArtistNames":Landon Pigg"'),
33 (N'"Artist_ID_19","ArtistNames":Andrew Foy"'),
34 (N'"Artist_ID_20","ArtistNames":Liam James"'),
35 (N'"Artist_ID_21,"ArtistNames":Royce Aviary"'),
36 (N'"Artist_ID_22,"ArtistNames":Jennel Garcia"'),
37 (N'"Artist_ID_23,"ArtistNames":Khalid"'),
38 (N'"Artist_ID_24,"ArtistNames":Sam Smith"'),
39 (N'"Artist_ID_25,"ArtistNames":KT Tunstall"'),
40 (N'"Artist_ID_26,"ArtistNames":Eddie Vedder"'),
41 (N'"Artist_ID_27,"ArtistNames":Lucius"'),
42 (N'"Artist_ID_28,"ArtistNames":Lil Peep"'),
43 (N'"Artist_ID_29,"ArtistNames":Tomohiro Nata"'),
44 (N'"Artist_ID_30,"ArtistNames":Andrew Belle"'),
45 (N'"Artist_ID_31,"ArtistNames":Bon Pope"'),
46 (N'"Artist_ID_32,"ArtistNames":Cassadee Pope"'),
47 (N'"Artist_ID_33,"ArtistNames":Aron Wright"'),
48 (N'"Artist_ID_34,"ArtistNames":Sara Bareilles"'),
49 (N'"Artist_ID_35,"ArtistNames":Kurt Cobain"'),
50 (N'"Artist_ID_36,"ArtistNames":Tim Heidecker"'),
51 (N'"Artist_ID_37,"ArtistNames":The City"'),
52 (N'"Artist_ID_38,"ArtistNames":Aaron Speer"'),
53 (N'"Artist_ID_39,"ArtistNames":Tyler Ward"'),
54 (N'"Artist_ID_40,"ArtistNames":Five For Fighting"')

```

Connected. (1/1) projectadbsms.database.windows.net SpotifyDB 00:00:00 0 rows

Ready 67° Search Ln 1 Col 1 Ch 1 INS 3:06 PM 5/27/2024

Document.sql - projectadbsm.database.windows.net.SpotifyDB (spotify (69)) - Microsoft SQL Server Management Studio

```

File Edit View Query Project Tools Window Help
Connect Object Explorer
projectadbsm.database.windows.net (SQL Server 12)
  Databases System Databases DemoGraph SpotifyDB Security Integration Services Catalogs
SpotifyDB Execute
  New Query Find Replace Copy All Paste All Undo Redo Cut Copy Insert Delete Select All Select Previous Select Next Select All Lines
  from
  Document.sql - proj...ipyDB (spotify (69)) × | SQLQuery_4.sql - pr...ipyDB (spotify (52))*
3011 ('N'["audio_id":991,"danceability":0.385,"energy":-12.619,"loudness":0,"mode":0.0327,"speechiness":0.942,"acousticness":0.0437,"instrumentalness":0.111,"liveness":0.618,"valence":184.933
3012 ('N'["audio_id":992,"danceability":0.693,"energy":-16.395,"loudness":1,"mode":0.0273,"speechiness":0.000315,"acousticness":0,"instrumentalness":0.3,"liveness":0.452,"valence":119.992,"tempo":121.275
3013 ('N'["audio_id":993,"danceability":0.61,"energy":-13.09,"loudness":0,"mode":0.0707,"speechiness":0.0278,"acousticness":0.335,"instrumentalness":0.0369,"liveness":0.669,"valence":121.275
3014 ('N'["audio_id":994,"danceability":0.61,"energy":-13.09,"loudness":0,"mode":0.0707,"speechiness":0.0278,"acousticness":0.335,"instrumentalness":0.0369,"liveness":0.669,"valence":121.275
3015 ('N'["audio_id":995,"danceability":0.621,"energy":-5.855,"loudness":1,"mode":0.037,"speechiness":0.721,"acousticness":0,"instrumentalness":0.0881,"liveness":0.867,"valence":113.896,"tempo":121.275
3016 ('N'["audio_id":996,"danceability":0.632,"energy":-6.84,"loudness":1,"mode":0.0863,"speechiness":0.0274,"acousticness":7.576_06,"instrumentalness":0.0957,"liveness":0.838,"valence":130.91
3017 ('N'["audio_id":997,"danceability":0.5947,"energy":-13.468,"loudness":1,"mode":0.11,"speechiness":0.558,"acousticness":2.546_05,"instrumentalness":0.0946,"liveness":0.7878,"valence":78.599
3018 ('N'["audio_id":998,"danceability":0.389,"energy":-8.275,"loudness":1,"mode":0.0927,"speechiness":0.0277,"acousticness":9.788_05,"instrumentalness":0.11,"liveness":0.8378,"valence":86.36
3019 ('N'["audio_id":999,"danceability":0.608,"energy":-6.89,"loudness":1,"mode":0.0325,"speechiness":0.406,"acousticness":0.18,"instrumentalness":0.178,"liveness":0.412,"valence":161.463,"tempo":121.275
3020 ('N'["audio_id":1000,"danceability":0.881,"energy":-6.48,"loudness":1,"mode":0.0934,"speechiness":0.00885,"acousticness":0.11,"instrumentalness":0.173,"liveness":0.173,"valence":96.9,"tempo":121.275
3021
3022
3023 select * from tblDocArtist
3024
3025 select * from tblDocSong
3026
3027 select * from tblDocAudioFeatures
3028
3029
3030
3031
3032
3033
3034
3035
3036
3037
3038
3039
3040
3041
3042
3043
3044
3045
3046
3047
3048
3049
3050
3051
3052
3053
3054
3055
3056
3057
3058
3059
3060
3061
3062
3063
3064
3065
3066
3067
3068
3069
3070
3071
3072
3073
3074
3075
3076
3077
3078
3079
3080
3081
3082
3083
3084
3085
3086
3087
3088
3089
3090
3091
3092
3093
3094
3095
3096
3097
3098
3099
3100
3101
3102
3103
3104
3105
3106
3107
3108
3109
3110
3111
3112
3113
3114
3115
3116
3117
3118
3119
3120
3121
3122
3123
3124
3125
3126
3127
3128
3129
3130
3131
3132
3133
3134
3135
3136
3137
3138
3139
3140
3141
3142
3143
3144
3145
3146
3147
3148
3149
3150
3151
3152
3153
3154
3155
3156
3157
3158
3159
3160
3161
3162
3163
3164
3165
3166
3167
3168
3169
3170
3171
3172
3173
3174
3175
3176
3177
3178
3179
3180
3181
3182
3183
3184
3185
3186
3187
3188
3189
3190
3191
3192
3193
3194
3195
3196
3197
3198
3199
3200
3201
3202
3203
3204
3205
3206
3207
3208
3209
3210
3211
3212
3213
3214
3215
3216
3217
3218
3219
3220
3221
3222
3223
3224
3225
3226
3227
3228
3229
3230
3231
3232
3233
3234
3235
3236
3237
3238
3239
3240
3241
3242
3243
3244
3245
3246
3247
3248
3249
3250
3251
3252
3253
3254
3255
3256
3257
3258
3259
3260
3261
3262
3263
3264
3265
3266
3267
3268
3269
3270
3271
3272
3273
3274
3275
3276
3277
3278
3279
3280
3281
3282
3283
3284
3285
3286
3287
3288
3289
3290
3291
3292
3293
3294
3295
3296
3297
3298
3299
3300
3301
3302
3303
3304
3305
3306
3307
3308
3309
3310
3311
3312
3313
3314
3315
3316
3317
3318
3319
3320
3321
3322
3323
3324
3325
3326
3327
3328
3329
3330
3331
3332
3333
3334
3335
3336
3337
3338
3339
3340
3341
3342
3343
3344
3345
3346
3347
3348
3349
3350
3351
3352
3353
3354
3355
3356
3357
3358
3359
3360
3361
3362
3363
3364
3365
3366
3367
3368
3369
3370
3371
3372
3373
3374
3375
3376
3377
3378
3379
3380
3381
3382
3383
3384
3385
3386
3387
3388
3389
3390
3391
3392
3393
3394
3395
3396
3397
3398
3399
3400
3401
3402
3403
3404
3405
3406
3407
3408
3409
3410
3411
3412
3413
3414
3415
3416
3417
3418
3419
3420
3421
3422
3423
3424
3425
3426
3427
3428
3429
3430
3431
3432
3433
3434
3435
3436
3437
3438
3439
3440
3441
3442
3443
3444
3445
3446
3447
3448
3449
3450
3451
3452
3453
3454
3455
3456
3457
3458
3459
3459
3460
3461
3462
3463
3464
3465
3466
3467
3468
3469
3470
3471
3472
3473
3474
3475
3476
3477
3478
3479
3480
3481
3482
3483
3484
3485
3486
3487
3488
3489
3490
3491
3492
3493
3494
3495
3496
3497
3498
3499
3500
3501
3502
3503
3504
3505
3506
3507
3508
3509
3509
3510
3511
3512
3513
3514
3515
3516
3517
3518
3519
3519
3520
3521
3522
3523
3524
3525
3526
3527
3528
3529
3529
3530
3531
3532
3533
3534
3535
3536
3537
3538
3539
3539
3540
3541
3542
3543
3544
3545
3546
3547
3548
3549
3549
3550
3551
3552
3553
3554
3555
3556
3557
3558
3559
3559
3560
3561
3562
3563
3564
3565
3566
3567
3568
3569
3569
3570
3571
3572
3573
3574
3575
3576
3577
3578
3579
3579
3580
3581
3582
3583
3584
3585
3586
3587
3588
3589
3589
3590
3591
3592
3593
3594
3595
3596
3597
3598
3599
3599
3600
3601
3602
3603
3604
3605
3606
3607
3608
3609
3609
3610
3611
3612
3613
3614
3615
3616
3617
3618
3619
3619
3620
3621
3622
3623
3624
3625
3626
3627
3628
3629
3629
3630
3631
3632
3633
3634
3635
3636
3637
3638
3639
3639
3640
3641
3642
3643
3644
3645
3646
3647
3648
3649
3649
3650
3651
3652
3653
3654
3655
3656
3657
3658
3659
3659
3660
3661
3662
3663
3664
3665
3666
3667
3668
3669
3669
3670
3671
3672
3673
3674
3675
3676
3677
3678
3679
3679
3680
3681
3682
3683
3684
3685
3686
3687
3688
3689
3689
3690
3691
3692
3693
3694
3695
3696
3697
3698
3699
3699
3700
3701
3702
3703
3704
3705
3706
3707
3708
3709
3709
3710
3711
3712
3713
3714
3715
3716
3717
3718
3719
3719
3720
3721
3722
3723
3724
3725
3726
3727
3728
3729
3729
3730
3731
3732
3733
3734
3735
3736
3737
3738
3739
3739
3740
3741
3742
3743
3744
3745
3746
3747
3748
3749
3749
3750
3751
3752
3753
3754
3755
3756
3757
3758
3759
3759
3760
3761
3762
3763
3764
3765
3766
3767
3768
3769
3769
3770
3771
3772
3773
3774
3775
3776
3777
3778
3779
3779
3780
3781
3782
3783
3784
3785
3786
3787
3788
3789
3789
3790
3791
3792
3793
3794
3795
3796
3797
3798
3799
3799
3800
3801
3802
3803
3804
3805
3806
3807
3808
3809
3809
3810
3811
3812
3813
3814
3815
3816
3817
3818
3819
3820
3821
3822
3823
3824
3825
3826
3827
3828
3829
3829
3830
3831
3832
3833
3834
3835
3836
3837
3838
3839
3839
3840
3841
3842
3843
3844
3845
3846
3847
3848
3849
3849
3850
3851
3852
3853
3854
3855
3856
3857
3858
3859
3859
3860
3861
3862
3863
3864
3865
3866
3867
3868
3869
3869
3870
3871
3872
3873
3874
3875
3876
3877
3878
3879
3879
3880
3881
3882
3883
3884
3885
3886
3887
3888
3889
3889
3890
3891
3892
3893
3894
3895
3896
3897
3898
3899
3899
3900
3901
3902
3903
3904
3905
3906
3907
3908
3909
3909
3910
3911
3912
3913
3914
3915
3916
3917
3918
3919
3919
3920
3921
3922
3923
3924
3925
3926
3927
3928
3929
3929
3930
3931
3932
3933
3934
3935
3936
3937
3938
3939
3939
3940
3941
3942
3943
3944
3945
3946
3947
3948
3949
3949
3950
3951
3952
3953
3954
3955
3956
3957
3958
3959
3959
3960
3961
3962
3963
3964
3965
3966
3967
3968
3969
3969
3970
3971
3972
3973
3974
3975
3976
3977
3978
3979
3979
3980
3981
3982
3983
3984
3985
3986
3987
3988
3989
3989
3990
3991
3992
3993
3994
3995
3996
3997
3998
3999
3999
4000
4001
4002
4003
4004
4005
4006
4007
4008
4009
4009
4010
4011
4012
4013
4014
4015
4016
4017
4018
4019
4019
4020
4021
4022
4023
4024
4025
4026
4027
4028
4029
4029
4030
4031
4032
4033
4034
4035
4036
4037
4038
4039
4039
4040
4041
4042
4043
4044
4045
4046
4047
4048
4049
4049
4050
4051
4052
4053
4054
4055
4056
4057
4058
4059
4059
4060
4061
4062
4063
4064
4065
4066
4067
4068
4069
4069
4070
4071
4072
4073
4074
4075
4076
4077
4078
4079
4079
4080
4081
4082
4083
4084
4085
4086
4087
4088
4089
4089
4090
4091
4092
4093
4094
4095
4096
4097
4098
4099
4099
4100
4101
4102
4103
4104
4105
4106
4107
4108
4109
4109
4110
4111
4112
4113
4114
4115
4116
4117
4118
4119
4119
4120
4121
4122
4123
4124
4125
4126
4127
4128
4129
4129
4130
4131
4132
4133
4134
4135
4136
4137
4138
4139
4139
4140
4141
4142
4143
4144
4145
4146
4147
4148
4149
4149
4150
4151
4152
4153
4154
4155
4156
4157
4158
4159
4159
4160
4161
4162
4163
4164
4165
4166
4167
4168
4169
4169
4170
4171
4172
4173
4174
4175
4176
4177
4178
4179
4179
4180
4181
4182
4183
4184
4185
4186
4187
4188
4189
4189
4190
4191
4192
4193
4194
4195
4196
4197
4198
4199
4199
4200
4201
4202
4203
4204
4205
4206
4207
4208
4209
4209
4210
4211
4212
4213
4214
4215
4216
4217
4218
4219
4219
4220
4221
4222
4223
4224
4225
4226
4227
4228
4229
4229
4230
4231
4232
4233
4234
4235
4236
4237
4238
4239
4239
4240
4241
4242
4243
4244
4245
4246
4247
4248
4249
4249
4250
4251
4252
4253
4254
4255
4256
4257
4258
4259
4259
4260
4261
4262
4263
4264
4265
4266
4267
4268
4269
4269
4270
4271
4272
4273
4274
4275
4276
4277
4278
4279
4279
4280
4281
4282
4283
4284
4285
4286
4287
4288
4289
4289
4290
4291
4292
4293
4294
4295
4296
4297
4298
4299
4299
4300
4301
4302
4303
4304
4305
4306
4307
4308
4309
4309
4310
4311
4312
4313
4314
4315
4316
4317
4318
4319
4319
4320
4321
4322
4323
4324
4325
4326
4327
4328
4329
4329
4330
4331
4332
4333
4334
4335
4336
4337
4338
4339
4339
4340
4341
4342
4343
4344
4345
4346
4347
4348
4349
4349
4350
4351
4352
4353
4354
4355
4356
4357
4358
4359
4359
4360
4361
4362
4363
4364
4365
4366
4367
4368
4369
4369
4370
4371
4372
4373
4374
4375
4376
4377
4378
4379
4379
4380
4381
4382
4383
4384
4385
4386
4387
4388
4389
4389
4390
4391
4392
4393
4394
4395
4396
4397
4398
4399
4399
4400
4401
4402
4403
4404
4405
4406
4407
4408
4409
4409
4410
4411
4412
4413
4414
4415
4416
4417
4418
4419
4419
4420
4421
4422
4423
4424
4425
4426
4427
4428
4429
4429
4430
4431
4432
4433
4434
4435
4436
4437
4438
4439
4439
4440
4441
4442
4443
4444
4445
4446
4447
4448
4449
4449
4450
4451
4452
4453
4454
4455
4456
4457
4458
4459
4459
4460
4461
4462
4463
4464
4465
4466
4467
4468
4469
4469
4470
4471
4472
4473
4474
4475
4476
4477
4478
4479
4479
4480
4481
4482
4483
4484
4485
4486
4487
4488
4489
4489
4490
4491
4492
4493
4494
4495
4496
4497
4498
4499
4499
4500
4501
4502
4503
4504
4505
4506
4507
4508
4509
4509
4510
4511
4512
4513
4514
4515
4516
4517
4518
4519
4519
4520
4521
4522
4523
4524
4525
4526
4527
4528
4529
4529
4530
4531
4532
4533
4534
4535
4536
4537
4538
4539
4539
4540
4541
4542
4543
4544
4545
4546
4547
4548
4549
4549
4550
4551
4552
4553
4554
4555
4556
4557
4558
4559
4559
4560
4561
4562
4563
4564
4565
4566
4567
4568
4569
4569
4570
4571
4572
4573
4574
4575
4576
4577
4578
4579
4579
4580
4581
4582
4583
4584
4585
4586
4587
4588
4589
4589
4590
4591
4592
4593
4594
4595
4596
4597
4598
4599
4599
4600
4601
4602
4603
4604
4605
4606
4607
4608
4609
4609
4610
4611
4612
4613
4614
4615
4616
4617
4618
4619
4619
4620
4621
4622
4623
4624
4625
4626
4627
4628
4629
4629
4630
4631
4632
4633
4634
4635
4636
4637
4638
4639
4639
4640
4641
4642
4643
4644
4645
4646
4647
4648
4649
4649
4650
4651
4652
4653
4654
4655
4656
4657
4658
4659
4659
4660
4661
4662
4663
4664
4665
4666
4667
4668
4669
4669
4670
4671
4672
4673
4674
4675
4676
4677
4678
4679
4679
4680
4681
4682
4683
4684
4685
4686
4687
4688
4689
4689
4690
4691
4692
4693
4694
4695
4696
4697
4698
4699
4699
4700
4701
4702
4703
4704
4705
4706
4707
4708
4709
4709
4710
4711
4712
4713
4714
4715
4716
4717
4718
4719
4719
4720
4721
4722
4723
4724
4725
4726
4727
4728
4729
4729
4730
4731
4732
4733
4734
4735
4736
4737
4738
4739
4739
4740
4741
4742
4743
4744
4745
4746
4747
4748
4749
4749
4750
4751
4752
4753
4754
4755
4756
4757
4758
4759
4759
4760
4761
4762
4763
4764
4765
4766
4767
4768
4769
4769
4770
4771
4772
4773
4774
4775
4776
4777
4778
4779
4779
4780
4781
4782
4783
4784
4785
4786
4787
4788
4789
4789
4790
4791
4792
4793
4794
4795
4796
4797
4798
4799
4799
4800
4801
4802
4803
4804
4805
4806
4807
4808
4809
4809
4810
4811
4812
4813
4814
4815
4816
4817
4818
4819
4819
4820
4821
4822
4823
4824
4825
4826
4827
4828
4829
4829
4830
4831
4832
4833
4834
4835
4836
4837
4838
4839
4839
4840
4841
4842
4843
4844
4845
4846
4847
4848
4849
4849
4850
4851
4852
4853
4854
4855
4856
4857
4858
4859
4859
4860
4861
4862
4863
4864
4865
4866
4867
4868
4869
4869
4870
4871
4872
4873
4874
4875
4876
4877
4878
4879
4879
4880
4881
4882
4883
4884
4885
4886
4887
4888
4889
4889
4890
4891
4892
4893
4894
4895
4896
4897
4898
4899
4899
4900
4901
4902
4903
4904
4905
4906
4907
4908
4909
4909
4910
4911
4912
4913
4914
4915
4916
4917
4918
4
```

The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar indicates the connection is to 'projectadbsms.database.windows.net.SpotifyDB (spotify (69)) - Microsoft SQL Server Management Studio'. The main window has 'Object Explorer' on the left showing the database structure, and 'SQLQuery\_4.sql' is open in the center. The query retrieves data from three tables: 'tblDocArtist', 'tblDocSong', and 'tblDocAudioFeatures'. The results pane shows the output of the query, which includes columns like 'audio\_id', 'danceability', 'energy', 'loudness', 'mode', 'speechiness', 'instrumentalness', 'valence', 'liveness', 'tempo', and 'acousticness'. The results are sorted by 'audio\_id'. A status bar at the bottom shows 'Query executed successfully.' and other system information.

## **Step 13: Created and Inserted data in Graph Database in Azure Data Studio**

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The top bar includes File, Edit, View, Help, and a search bar. The left sidebar displays the 'CONNECTIONS' node, which is expanded to show 'Servers' (containing 'projectedbms.database.windows.net...'), 'Databases' (containing 'SpotifyDB'), 'System Databases', 'DemoGraph', and 'Tables'. Under 'Tables', there are many entries starting with 'dbo.' such as 'adds', 'creates', 'listens', 'PlaylistGraph', 'SongGraph', 'tblArtist', 'tblArtistSong', 'tblAudioFeatures', 'tblDocArtist', 'tblDocAudioFeatures', 'tblDocSong', 'tblPaymentMethod', 'tblPlaylist', 'tblSong', 'tblSongAudioFeature', 'tblSubscriptionPlan', 'tblUser', 'tblUserAudit', 'tblUserPlaylist', 'UserGraph', and 'Dropped Ledger Tables'. Other nodes like 'Views', 'Synonyms', 'Programmability', 'External Resources', 'Storage', 'Security', and 'Azure' are also visible. The main area shows a query window titled 'SQLQuery\_1 - (69 p.../spotify)' with the following T-SQL code:

```
1 SELECT TOP (1000) [edge_id_4F2B33A002334731B8B0E8B1CC4B5AB1]
2      ,[#from_id_5C938A732BF041D4AC7CD017FE18433]
3      ,[$to_id_2A82BD944FD74EE98F6EF98833A60DA6]
4  FROM [dbo].[adds]
```

The results grid below the code shows 19 rows of data, each representing an edge from the 'adds' table. Each row contains three columns: '\$edge\_id\_4F2B33A002334731B8B0E8B1CC4B5AB1', '\$from\_id\_5C938A732BF041D4AC7CD017FE18433', and '\$to\_id\_2A82BD944FD74EE98F6EF98833A60DA6'. The data consists of JSON objects representing edges between nodes in the SpotifyDB schema.

CONNECTIONS ... SQLQuery\_1 - (69) p...otify SQLQuery\_2 - (86) p...otify

Run Cancel Disconnect Change Database: SpotifyDB Estimated Plan Enable Actual Plan Parse

```

1 SELECT TOP (1000) [node_id_f57d4a960f594082907eb48f2cfa118]
2   ,[UserID]
3   ,[Username]
4   ,[Email]
5   ,[Gender]
6   ,[ListeningDevice]
7   ,[Region]
8   ,[Followers]
9   ,[SubscriptionID]
10  ,[PaymentID]
11 FROM [dbo].[UserGraph]

```

Results Messages

\$node_id_f57d4a960f594082907eb48f2cfa118	UserID	UserName	Email	Gender	ListeningDevice	Region
1 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 0}	1	Kimbell Bledhill	kbledhill@shareseale.com	Male	Windows PC	MW-01
2 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 1}	2	Leo Yarrall	lyarrall@fiathis.com	Male	Streaming Stick	CA-NT
3 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 2}	3	Eisabeth Brickwood	ebrickwood2@ebay.com	Female	macOS Computers	BR-AH
4 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 3}	4	Bambi Hollyman	bhollyman3@imageshack.us	Female	Smart TV	CG-12
5 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 4}	5	Amos Beagrie	abeagrie4@ycombinator.com	Male	In-car infotainment system	BR-GO
6 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 5}	6	Johanna Stout	jstout@spotify.com	Female	Android Phone	US-NY
7 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 6}	7	Curtice Seggison	cseggison@outlook.com	Male	iPhone	UK-LN
8 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 7}	8	Belva Saylor	bsaylor7@yahoo.com	Female	Smart Speaker	DE-BE
9 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 8}	9	Brendon Blaxter	bbaxter8@google.com	Male	Windows PC	AU-NSW
10 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 9}	10	Clementius Bethel	cbethel9@bing.com	Male	Smart TV	CA-BC
11 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 10}	11	Daphne Stanley	dstanley@ymail.com	Female	MacBook	US-CA
12 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 11}	12	Emery Couper	ecouper@icloud.com	Male	iPad	IN-DL
13 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 12}	13	Fay Glasspool	fglasspool@instagram.com	Female	Linux PC	FR-IDF
14 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 13}	14	Gavin McKean	gmckean@live.com	Male	Android Tablet	US-TX
15 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 14}	15	Hillary Lutwyche	hilutwyche@webmd.com	Female	iPhone	US-FL
16 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 15}	16	Irving Gregson	igregson@earthlink.net	Male	Smart Watch	CA-ON
17 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 16}	17	Jade Bonnet	jbonnet8@tiktok.com	Female	Windows Laptop	AU-VIC
18 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 17}	18	Kurt Brigham	kbrigham@twitter.com	Male	MacBook	DE-HH
19 {"type": "node", "schema": "dbo", "table": "UserGraph", "id": 18}	19	Lacey Burdon	lburdon1@netflix.com	Female	Smart TV	IN-MH

SERVERS ... projectadms.database.windows.net ...

Databases System Databases DemoGraph SpotifyDB

Tables dbo.adds dbo.creates dbo.listens dbo.PlaylistGraph dbo.SongGraph dbo.tbArtist dbo.tbArtistSong dbo.tbAudioFeatures dbo.tbDocArtist dbo.tbDocAudioFeatures dbo.tbDocSong dbo.tbPaymentMethod dbo.tbPlaylist dbo.tbSong dbo.tbSongAudioFeature dbo.tbSubscriptionPlan dbo.tbUser dbo.tbUserAudit dbo.tbUserPlaylist

dbo.UserGraph Columns Keys Constraints Triggers Indexes Statistics Dropped Ledger Tables Views

AZURE