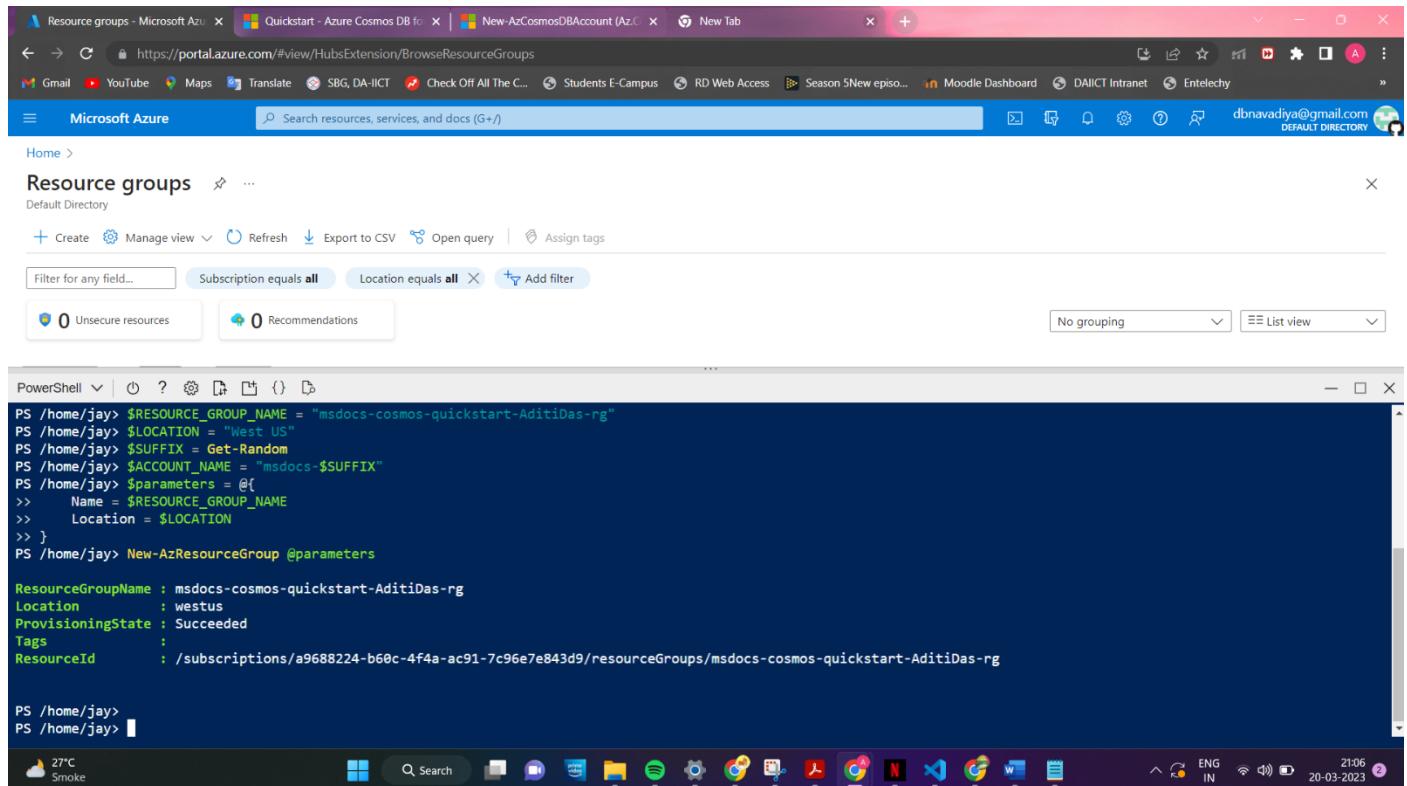


1. Azure Cosmos DB for Table for .NET.

i. Creating resource group (msdocs-cosmos-quickstart-aditi-rg):

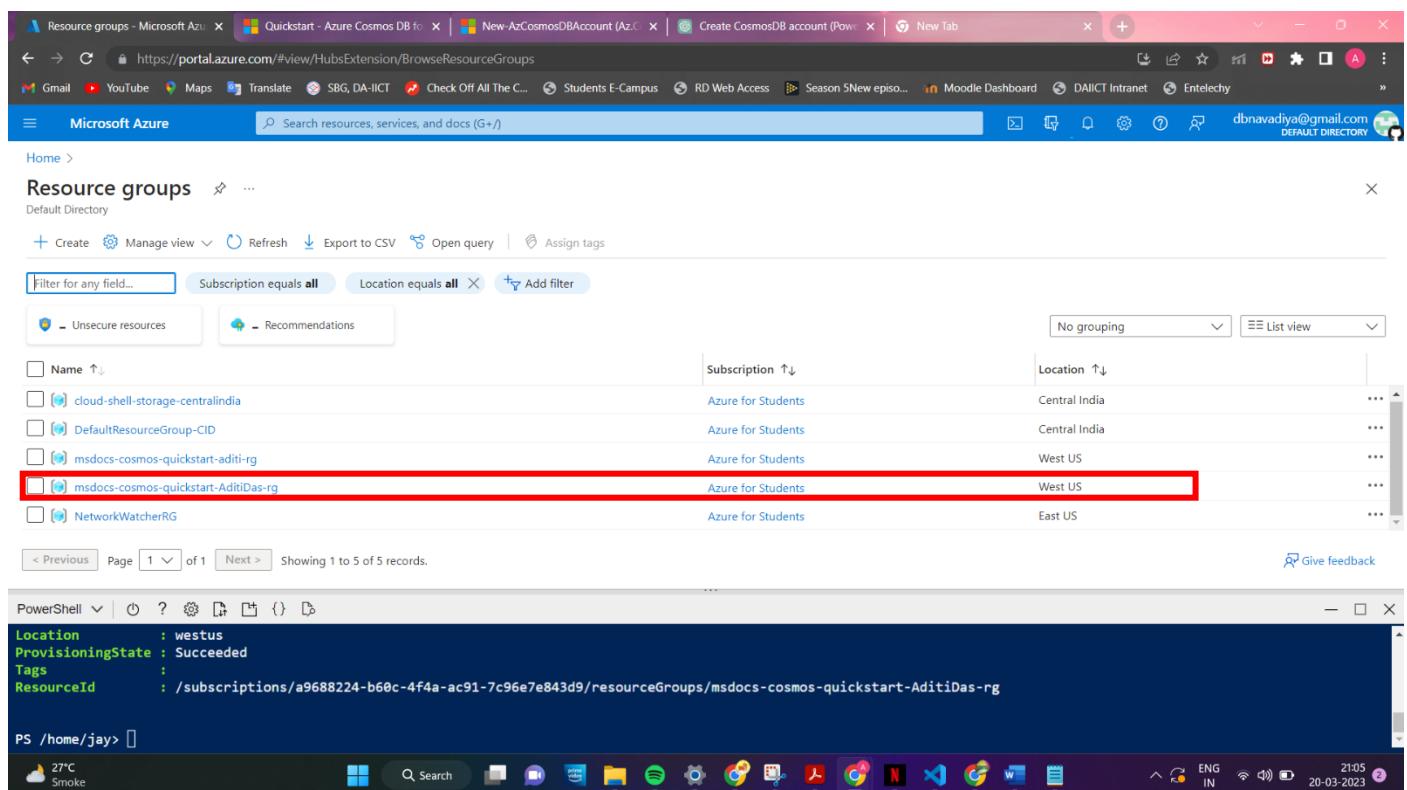


The screenshot shows the Microsoft Azure portal's Resource Groups page. A PowerShell window is open below the browser, displaying the command to create a new resource group:

```
PS /home/jay> $RESOURCE_GROUP_NAME = "msdocs-cosmos-quickstart-AditiDas-rg"
PS /home/jay> $LOCATION = "West US"
PS /home/jay> $SUFFIX = Get-Random
PS /home/jay> $ACCOUNT_NAME = "msdocs-$SUFFIX"
PS /home/jay> $parameters = @{
>>   Name = $RESOURCE_GROUP_NAME
>>   Location = $LOCATION
>> }
PS /home/jay> New-AzResourceGroup @parameters

ResourceGroupName : msdocs-cosmos-quickstart-AditiDas-rg
Location          : westus
ProvisioningState : Succeeded
Tags              :
ResourceId        : /subscriptions/a9688224-b60c-4f4a-ac91-7c96e7e843d9/resourceGroups/msdocs-cosmos-quickstart-AditiDas-rg

PS /home/jay>
PS /home/jay>
```



The screenshot shows the Microsoft Azure portal's Resource Groups page after the resource group has been created. The newly created resource group, 'msdocs-cosmos-quickstart-AditiDas-rg', is highlighted with a red border in the list.

Name	Subscription	Location
cloud-shell-storage-centralindia	Azure for Students	Central India
DefaultResourceGroup-CID	Azure for Students	Central India
msdocs-cosmos-quickstart-aditi-rg	Azure for Students	West US
msdocs-cosmos-quickstart-AditiDas-rg	Azure for Students	West US
NetworkWatcherRG	Azure for Students	East US

Below the list, a PowerShell window shows the details of the newly created resource group:

```
PS /home/jay> msdocs-cosmos-quickstart-AditiDas-rg
Location          : westus
ProvisioningState : Succeeded
Tags              :
ResourceId        : /subscriptions/a9688224-b60c-4f4a-ac91-7c96e7e843d9/resourceGroups/msdocs-cosmos-quickstart-AditiDas-rg

PS /home/jay>
```

ii. Creating Cosmos DB account:

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar titled 'Resource groups' with a list of existing resource groups: 'cloud-shell-storage-centralindia', 'DefaultResourceGroup-CID', 'msdocs-cosmos-quickstart-AditiDas-rg' (which is selected), and 'NetworkWatcherRG'. The main content area is titled 'msdocs-cosmos-quickstart-AditiDas-rg' and shows a table of resources. A single row is highlighted with a red box: 'msdocs-527297' (Type: Azure Cosmos DB account, Location: West US). Below the portal, a PowerShell window is open, displaying the command used to create the account: PS /home/jay> New-AzCosmosDBAccount @parameters. The system tray at the bottom shows the date as 20-03-2023 and the time as 21:11.

iii. Creating Table in CosmosDB:

The screenshot shows the Microsoft Azure portal interface. The main content area is titled 'msdocs-527297' and displays a PowerShell session. The session shows the creation of a new table named 'WeatherData' under the account 'msdocs-527297'. The output of the command 'New-AzCosmosDBTable' is shown, including the table's name, account name, and resource group. The system tray at the bottom shows the date as 20-03-2023 and the time as 22:36.

The screenshot shows the Microsoft Azure portal interface. At the top, there are two tabs: 'msdocs 527297 - Microsoft Azure' and 'Quickstart: API for Table with Pyt...'. The main content area is titled 'msdocs-527297' and describes it as an 'Azure Cosmos DB account'. On the left, a sidebar lists various settings like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Cost Management, Quick start, Notifications, Data Explorer, and Settings. The 'Tables' section is highlighted with a red box. It contains a table with three columns: ID, Database, and Throughput (RU/s). There is one entry: 'WeatherData' in 'TablesDB' with a throughput of 400. Below the table, there are sections for Monitoring and Requests.

iv. Getting Connection Strings:

The screenshot shows a Windows PowerShell window with a blue background. The command entered is:

```
PS /home/jay> $res = $(Get-AzCosmosDBAccountKey -ResourceGroupName $resourceGroupName -Name $cosmosAccountName -Type "ConnectionStrings").Primary Table Connection String
```

. The output is a very long connection string starting with:

```
DefaultEndpointsProtocol=https;AccountName=msdocs-527297;AccountKey=Pc6tJfv8lsm0UKWjuk8iBjY7uDgxlrBa2VQxHWrLYM1zZ6iWduYxyAmXq9DjOTSQihMxmFjKinD6ACDbkkme7Q==;TableEndpoint=https://msdocs-527297.table.cosmos.azure.com:443/
```

. The PowerShell window has a dark blue header bar with the title 'msdocs-527297 - Microsoft Azure' and a tab for 'Quickstart: API for Table with Pyt...'. The taskbar at the bottom shows various icons for apps like File Explorer, Edge, and File Explorer.

Connection String:

```
DefaultEndpointsProtocol=https;AccountName=msdocs-527297;AccountKey=Pc6tJfv8lsm0UKWjuk8iBjY7uDgxlrBa2VQxHWrLYM1zZ6iWduYxyAmXq9DjOTSQihMxmFjKinD6ACDbkkme7Q==;TableEndpoint=https://msdocs-527297.table.cosmos.azure.com:443/;
```

v. Running the Web App:

a. Installing requirements.text

```
2-completed-app
C:\WINDOWS\system32\cmd

Microsoft Windows [Version 10.0.22621.1413]
(c) Microsoft Corporation. All rights reserved.

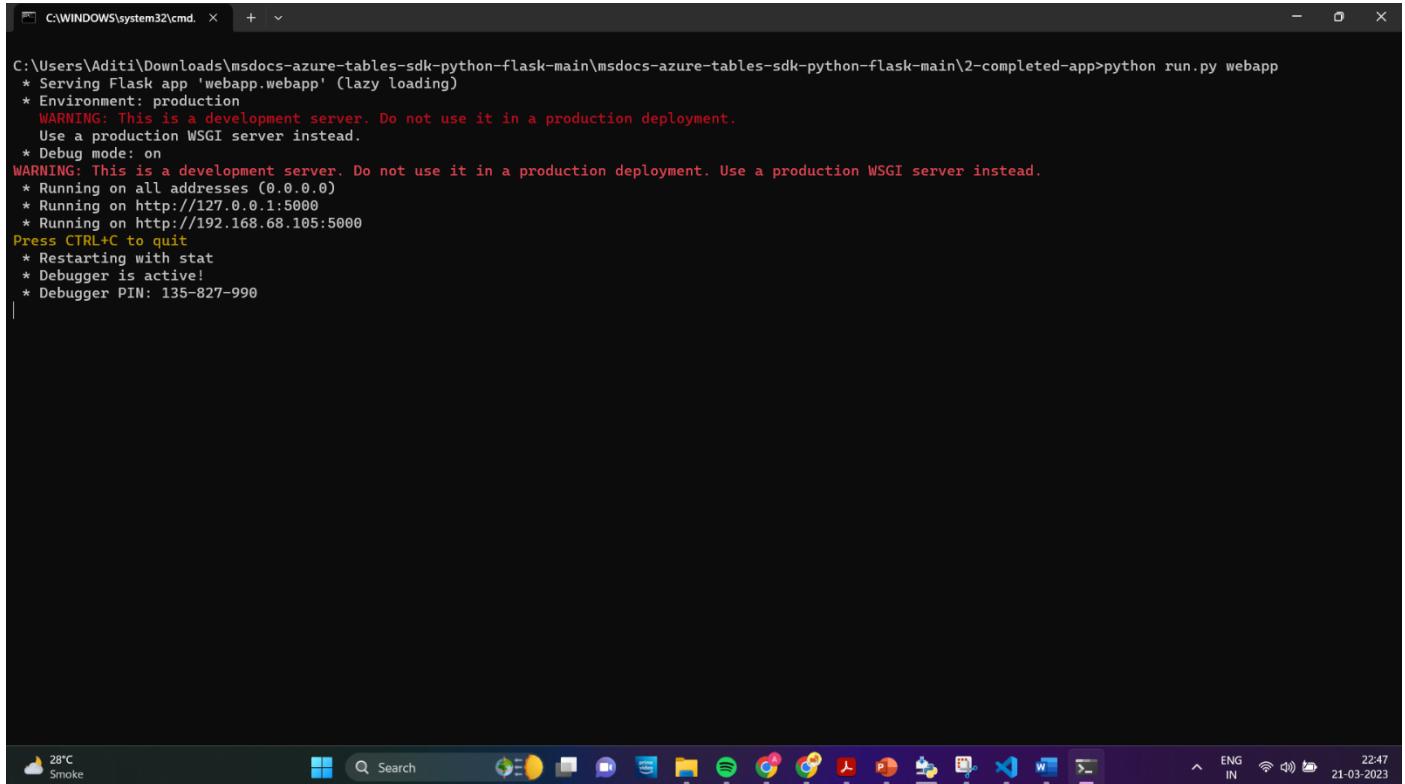
C:\Users\Aditi\Downloads\msdocs-azure-tables-sdk-python-flask-main\msdocs-azure-tables-sdk-python-flask-main\2-completed
-app>pip3 install -r requirements.txt
Collecting azure-data-tables==12.2.0
  Downloading azure_data_tables-12.2.0-py2.py3-none-any.whl (108 kB)
    108.1/108.1 kB 6.1 MB/s eta 0:00:00
Collecting Flask==2.0.3
  Downloading Flask-2.0.3-py3-none-any.whl (95 kB)
    95.6/95.6 kB ? eta 0:00:00
Collecting python-dotenv==0.19.2
  Downloading python_dotenv-0.19.2-py2.py3-none-any.whl (17 kB)
Requirement already satisfied: azure-core<2.0.0,>=1.14.0 in c:\users\aditi\appdata\local\programs\python\python310\lib\s
ite-packages (from azure-data-tables==12.2.0->r requirements.txt (line 1)) (1.26.3)
Collecting msrest>=0.6.21
  Downloading msrest-0.7.1-py3-none-any.whl (85 kB)
    85.4/85.4 kB ? eta 0:00:00
Collecting click>=7.1.2
  Downloading click-8.1.3-py3-none-any.whl (96 kB)
    96.6/96.6 kB ? eta 0:00:00
Collecting Jinja2>=3.0
  Downloading Jinja2-3.1.2-py3-none-any.whl (133 kB)
    133.1/133.1 kB 2.6 MB/s eta 0:00:00
Collecting itsdangerous>=2.0
  Downloading itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Collecting Werkzeug>=2.0
  Downloading Werkzeug-2.2.3-py3-none-any.whl (233 kB)
    233.6/233.6 kB 14.0 MB/s eta 0:00:00
Requirement already satisfied: requests>=2.18.4 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packs
a
New Volume (D)
Network
7 items |
```

```
C:\WINDOWS\system32\cmd. + - X

line 3)) (0.19.2)
Requirement already satisfied: msrest<=0.6.21 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from azure-data-tables==12.2.0->-r requirements.txt (line 1)) (0.7.1)
Requirement already satisfied: azure-core<2.0.0,>=1.14.0 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from azure-data-tables==12.2.0->-r requirements.txt (line 1)) (1.26.3)
Requirement already satisfied: itsdangerous>=2.0 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from Flask==2.0.3->-r requirements.txt (line 2)) (2.1.2)
Requirement already satisfied: Werkzeug>=2.0 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from Flask==2.0.3->-r requirements.txt (line 2)) (2.2.3)
Requirement already satisfied: Jinja2>=3.0 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from Flask==2.0.3->-r requirements.txt (line 2)) (3.1.2)
Requirement already satisfied: click>=7.1.2 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from Flask==2.0.3->-r requirements.txt (line 2)) (8.1.3)
Requirement already satisfied: typing-extensions>=4.0.1 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from azure-core<2.0.0,>=1.14.0->azure-data-tables==12.2.0->-r requirements.txt (line 1)) (4.5.0)
Requirement already satisfied: six>=1.11.0 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from azure-core<2.0.0,>=1.14.0->azure-data-tables==12.2.0->-r requirements.txt (line 1)) (1.16.0)
Requirement already satisfied: requests>=2.18.4 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from azure-core<2.0.0,>=1.14.0->azure-data-tables==12.2.0->-r requirements.txt (line 1)) (2.28.2)
Requirement already satisfied: colorama in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from click>=7.1.2->Flask==2.0.3->-r requirements.txt (line 2)) (0.4.6)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from Jinja2>=3.0->Flask==2.0.3->-r requirements.txt (line 2)) (2.1.2)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from msrest<=0.6.21->azure-data-tables==12.2.0->-r requirements.txt (line 1)) (2022.12.7)
Requirement already satisfied: requests-oauthlib>=0.5.0 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from msrest<=0.6.21->azure-data-tables==12.2.0->-r requirements.txt (line 1)) (3.1.3)
Requirement already satisfied: isodate>=0.6.0 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from msrest<=0.6.21->azure-data-tables==12.2.0->-r requirements.txt (line 1)) (0.6.1)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from requests>=2.18.4->azure-core<2.0.0,>=1.14.0->azure-data-tables==12.2.0->-r requirements.txt (line 1)) (3.1.0)
Requirement already satisfied: urllib3<1.27,>>1.21.1 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from requests>=2.18.4->azure-core<2.0.0,>=1.14.0->azure-data-tables==12.2.0->-r requirements.txt (line 1)) (1.26.15)
Requirement already satisfied: idna<4,>=2.5 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from requests>=2.18.4->azure-core<2.0.0,>=1.14.0->azure-data-tables==12.2.0->-r requirements.txt (line 1)) (3.4)
Requirement already satisfied: oauthlib>=3.0.0 in c:\users\aditi\appdata\local\programs\python\python310\lib\site-packages (from requests-oauthlib>=0.5.0->msrest<=0.6.21->azure-data-tables==12.2.0->-r requirements.txt (line 1)) (3.2.2)

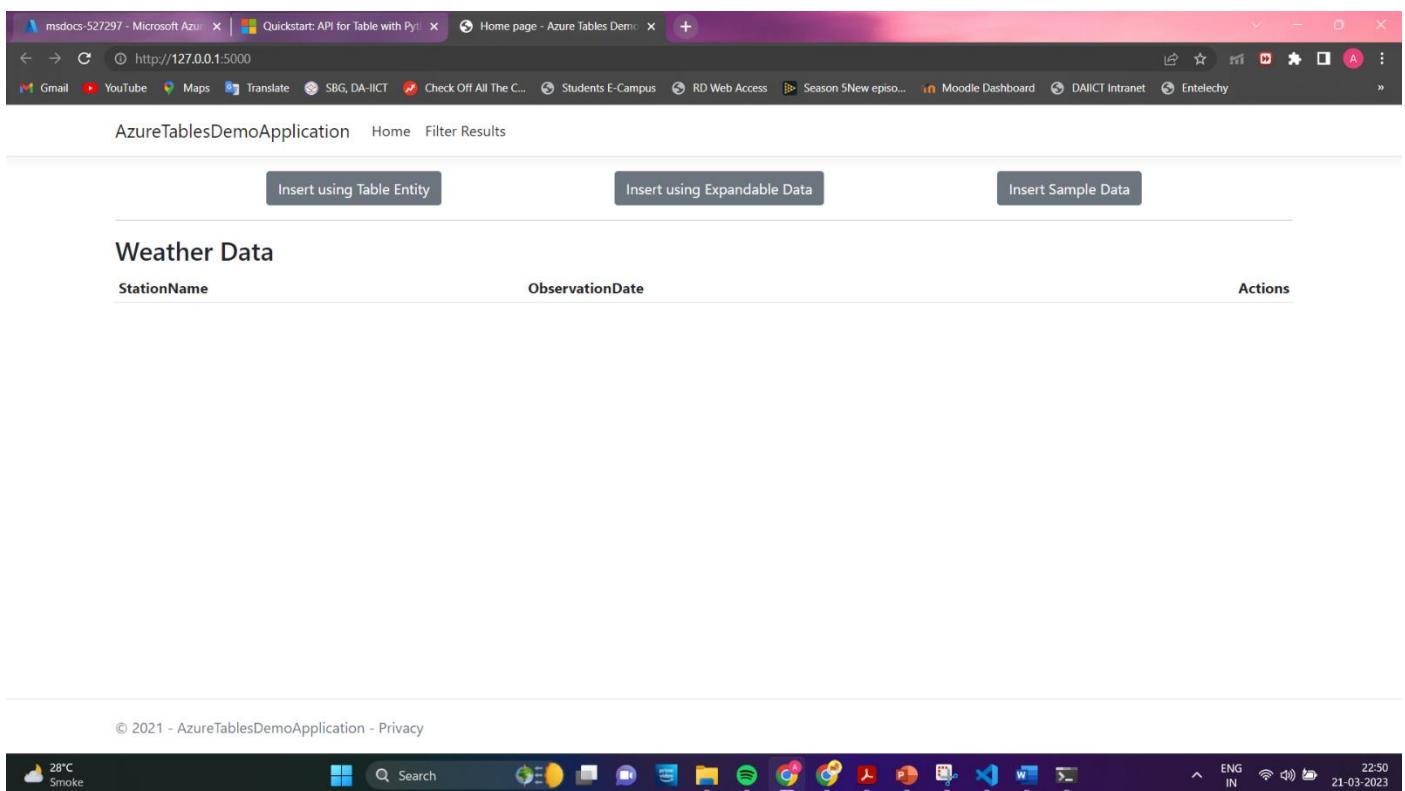
[notice] A new release of pip available: 22.3.1 -> 23.0.1
[notice] To update, run: python.exe -m pip install --upgrade pip
```

b. Running Web App:



```
C:\Users\Aditi\Downloads\msdocs-azure-tables-sdk-python-flask-main\msdocs-azure-tables-sdk-python-flask-main\2-completed-app>python run.py webapp
 * Serving Flask app 'webapp.webapp' (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://192.168.68.105:5000
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 135-827-990
```

Link : <http://127.0.0.1:5000>



msdocs-527297 - Microsoft Azure | Quickstart: API for Table with Python | Home page - Azure Tables Demo

http://127.0.0.1:5000

AzureTablesDemoApplication Home Filter Results

Insert using Table Entity Insert using Expandable Data Insert Sample Data

Weather Data

StationName	ObservationDate	Actions

© 2021 - AzureTablesDemoApplication - Privacy

Here, we can see that the table is empty.

vi. Inserting Data using “Insert Using Table Entity”

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "msdocs-527297 - Microsoft Azure" and displays the URL "http://127.0.0.1:5000". The page content is titled "AzureTablesDemoApplication" and shows a form titled "Insert using TableEntity". The form has fields for "Station Name" (Chicago), "Date/Time" (20-03-2023, 15:00), and various weather parameters: Temperature (91), Humidity (50), Barometer (30.03), Wind Direction (W), Wind Speed (7), and Precipitation (0.00). Below the form are three buttons: "Close", "Insert", and "Upsert". The background shows a "Weather Data" table with one row for Chicago. The status bar at the bottom indicates the date as 21-03-2023 and the time as 22:51.

After Insertion:

The screenshot shows the same web browser window after the data has been inserted. The "Weather Data" table now has two rows. The first row is highlighted with a red border and contains the data: StationName (Chicago), ObservationDate (2023-03-20 15:00), Temperature (91), Humidity (50), Barometer (30.03), WindDirection (W), WindSpeed (7), and Precipitation (0.00). To the right of this row are "Update" and "Delete" buttons. The status bar at the bottom indicates the date as 21-03-2023 and the time as 22:51.

vii. Inserting data using “Insert Using Expandable Data”

Insert using Custom Entity (implementing ITableEntity)

Station Name	Date/Time	Temperature	Humidity	Barometer	Wind Direction	Wind Speed	Precipitation	Cloud Cover	UV Index
Weather station name	dd-mm-yyyy	Current temperature	% Humidity	Atmospheric pressure	CALM	Wind speed	Precipitation	Percent Cloud Cover	UV Index

Add Custom Field

Close Insert Upsert

Weather Data

StationName	Observation
Chicago	2023-03-20

I inserted a field called Maximum Temperature:

Add custom field

Station Name	Custom field name
Chicago	Maximum Temperature

Date/Time	Temperature	Humidity	Barometer	Wind Direction	Wind Speed	Precipitation	Cloud Cover	UV Index
2023-03-20	75	25	25	CALM	6	0.00	40	4

Add Custom Field

Close Insert Upsert

Weather Data

StationName	Observation
Chicago	2023-03-20

AzureTablesDemoApplication Home Filter Results

Insert using Custom Entity (implementing ITableEntity)

Station Name	Observation
Chicago	2023-03-20 15:00

Weather Data

StationName	Observation
Chicago	2023-03-20 15:00

Station Name: Los Angeles

Date/Time: 20-03-2023 16:00

Temperature: 100

Humidity: 75

Barometer: 25

Wind Direction: CALM

Wind Speed: 6

Precipitation: 0.00

Cloud Cover: 40

UV Index: 4

Maximum Temperature: 105

Add Custom Field

Close Insert Upsert

© 2021 - AzureTablesDemoApplication

28°C Smoke Search ENG IN 21-03-2023

AzureTablesDemoApplication Home Filter Results

Insert using Table Entity Insert using Expandable Data Insert Sample Data

Weather Data

StationName	ObservationDate	Temperature	Humidity	Barometer	WindDirection	WindSpeed	Precipitation	CloudCover	UvIndex	MaximumTemperature	Actions
Chicago	2023-03-20 15:00	91	50	30.03	W	7	0.00	None	None	None	<button>Update</button> <button>Delete</button>
Los Angeles	2023-03-20 16:00	100	75	25	CALM	6	0.00	40	4	105	<button>Update</button> <button>Delete</button>

© 2021 - AzureTablesDemoApplication - Privacy

28°C Smoke Search ENG IN 21-03-2023

viii. Inserting data using “Insert Sample Data”

I added data for Phoenix:

The screenshot shows a web browser window with the URL <http://127.0.0.1:5000>. The page title is "AzureTablesDemoApplication". A modal dialog box titled "Insert sample data" is open, showing fields for "Metric" (set to "Phoenix") and "City" (set to "Phoenix"). Below the dialog is a table titled "Weather Data" with columns: StationName, ObservationDate, Temperature, Humidity, Barometer, WindDirection, WindSpeed, Precipitation, CloudCover, UvIndex, MaximumTemperature, and Actions. The table contains two rows: one for Chicago (2023-03-20 15:00, 91, 50) and one for Los Angeles (2023-03-20 16:00, 100, 75). The "Actions" column for each row includes "Update" and "Delete" buttons. At the bottom of the page, there is a footer with the text "© 2021 - AzureTablesDemoApplication - Privacy".

The screenshot shows a web browser window with the URL <http://127.0.0.1:5000>. The page title is "AzureTablesDemoApplication". There are three tabs at the top: "Insert using Table Entity" (selected), "Insert using Expandable Data", and "Insert Sample Data". Below the tabs is a table titled "Weather Data" with columns: StationName, ObservationDate, Temperature, Humidity, Barometer, WindDirection, WindSpeed, Precipitation, CloudCover, UvIndex, MaximumTemperature, and Actions. The table contains eight rows, all for Phoenix, with various dates and time points. The "Actions" column for each row includes "Update" and "Delete" buttons. The bottom of the page has a dark footer bar with various icons and the date "21-03-2023".

ix. Filtering Data:

Here, I used city name “Chicago” as filter to filter out my output:

The screenshot shows a web-based application for filtering data from an Azure Table. At the top, there's a header bar with tabs for "msdocs-527297 - Microsoft Azure", "Quickstart: API for Table with Pyt...", and "Home page - Azure Tables Demo". Below the header, the URL is http://127.0.0.1:5000/?filter=true&partitionKey=Chicago&rowKeyDateStart=&rowKeyTimeStart=&rowKeyDateEnd=&rowKeyTimeEnd=&minTemperature=&maxTemperature=&mi...". The main content area has two sections: "AzureTablesDemoApplication" and "Filter Results".

Filtering Criteria:

Column	Criteria
Station Name (Partition Key)	Chicago
Date (Row Key)	Min date/time: dd-mm-yyyy Max date/time: dd-mm-yyyy
Temperature	Min temperature: Max temperature:
Precipitation	Min precipitation: Max precipitation:

A green "Filter data" button is located below the criteria section.

Weather Data Table:

StationName	ObservationDate	Temperature	Humidity	Barometer	WindDirection	WindSpeed	Precipitation	Actions
Chicago	2023-03-20 15:00	91	50	30.03	W	7	0.00	<button>Update</button> <button>Delete</button>

At the bottom, there's a copyright notice: "© 2021 - AzureTablesDemoApplication - Privacy". The taskbar at the very bottom shows various pinned icons and the date: 21-03-2023.

X. The Table on Azure Portal:

The screenshot shows the Microsoft Azure Data Explorer interface for a Cosmos DB account named "msdocs-527297". The left sidebar includes links for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Cost Management, Quick start, Notifications, and Data Explorer (which is selected). The top navigation bar shows "Microsoft Azure" and the user email "dbnavadiya@gmail.com".

The main area displays the "TABLE API" view for the "TablesDB" table. A search bar at the top allows filtering by "PartitionKey" and "RowKey". The results table shows the following data:

PartitionKey	RowKey	Timestamp	Barometer	CloudCover	Humidity	MaximumTemperature	Precipitation	Temperature	UVIndex	Wind
Chicago	2023-03-20 15:00	Tue, 21 Mar 2023 17:21:51 GMT	30.03		50		0.00	91		W
Los Angeles	2023-03-20 16:00	Tue, 21 Mar 2023 17:32:04 GMT	25	40	75	105	0.00	100	4	CALM
Phoenix	2021-07-01 00:00	Tue, 21 Mar 2023 17:35:43 GMT	28.67		39		0	89		SE
Phoenix	2021-07-01 03:00	Tue, 21 Mar 2023 17:35:43 GMT	28.66		40		0	88		E
Phoenix	2021-07-01 06:00	Tue, 21 Mar 2023 17:35:43 GMT	28.69		38		0	86		E

At the bottom, there's a PowerShell prompt and the taskbar at the very bottom shows the date: 21-03-2023.

Following this, I stopped the execution of the web page using “Ctrl+C” in the command prompt and cleared all the resources.