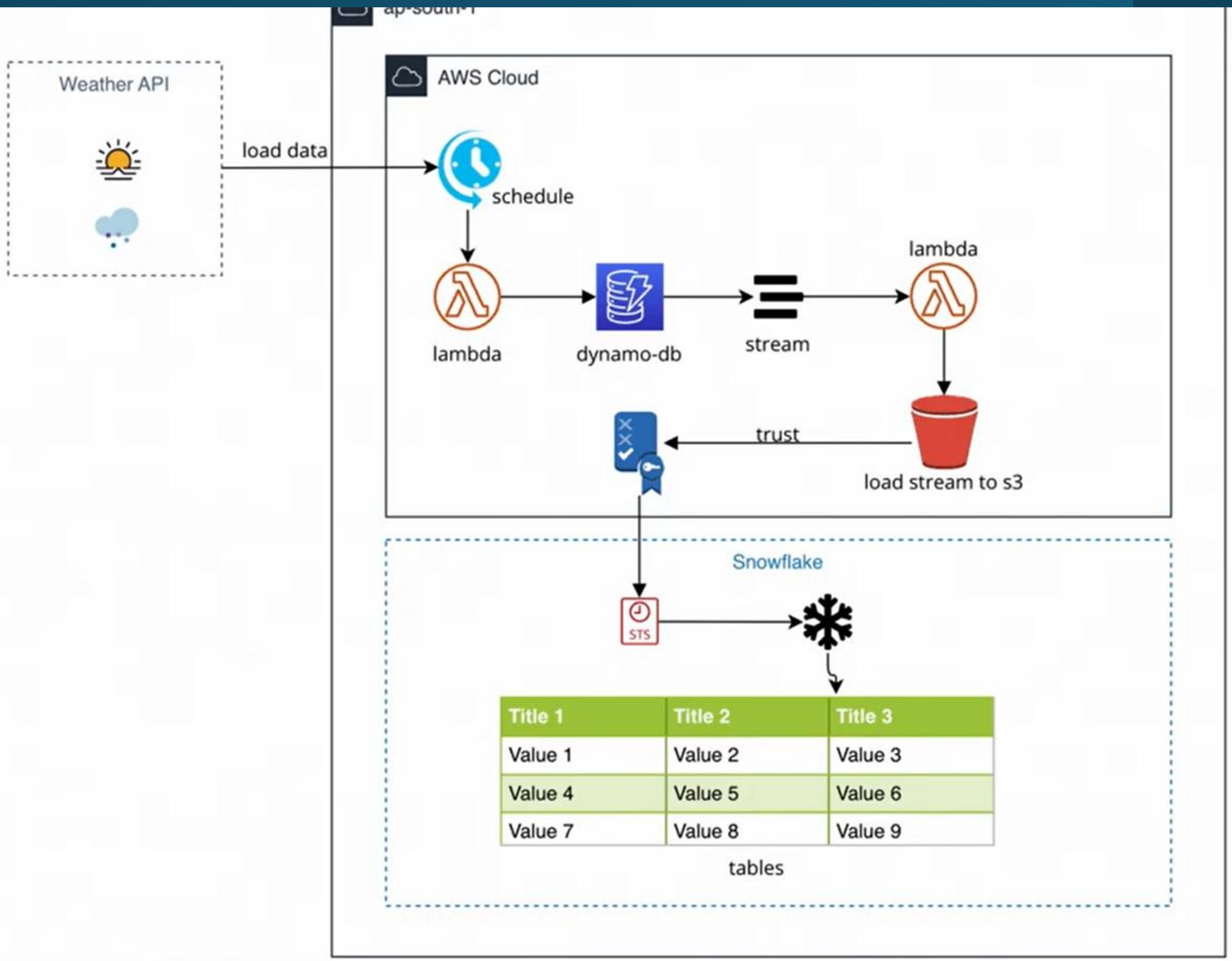


Title 1	Title 2	Title 3
Value 1	Value 2	Value 3
Value 4	Value 5	Value 6
Value 7	Value 8	Value 9

Weather_Pipeline

Architecture:

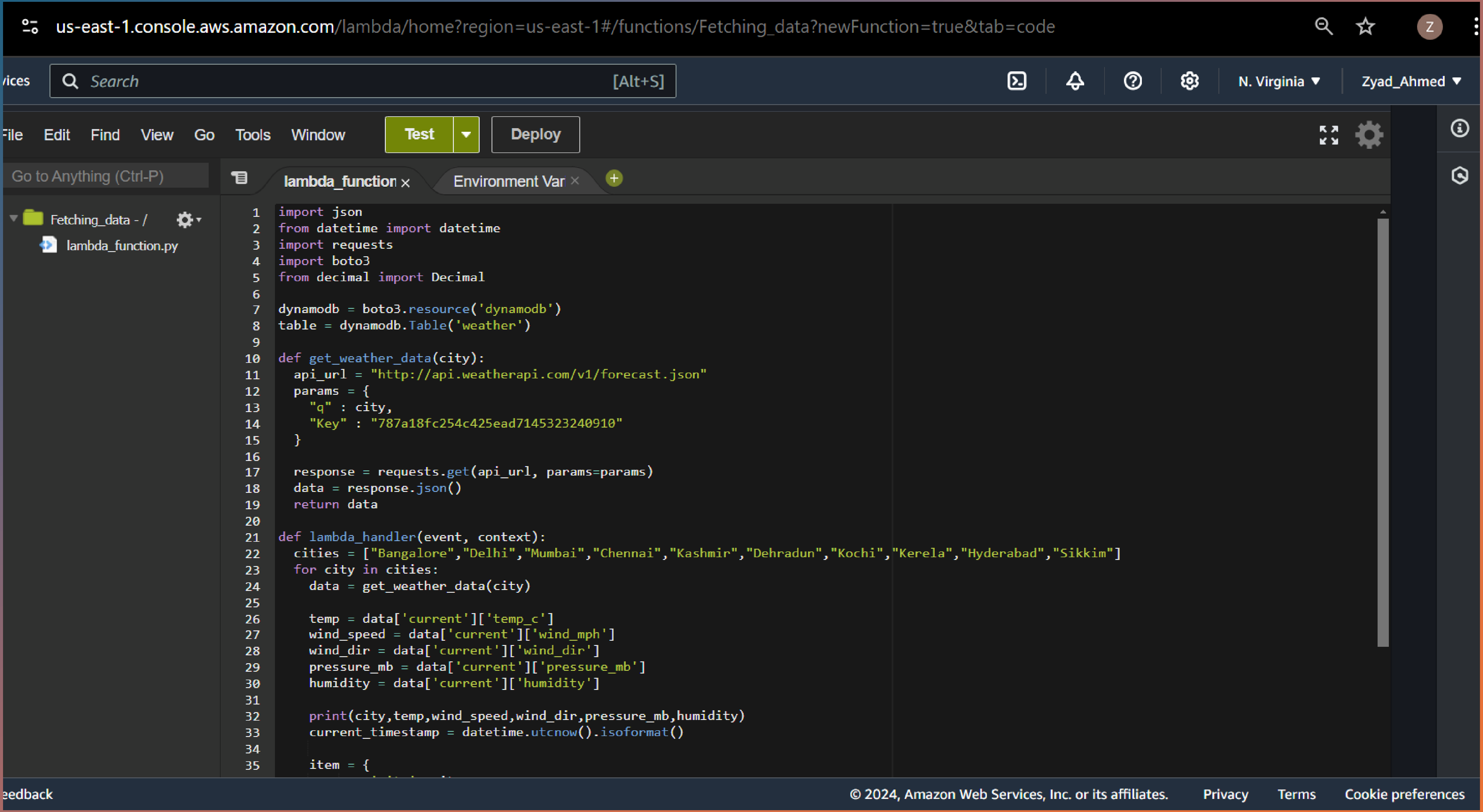
DynamoDB: *NoSQL database used for storing the ingested real-time weather data.*

AWS Lambda: *Serverless function to trigger and process the flow of data between DynamoDB and Snowflake.*

Snowpipe: *Snowflake feature used for automating the data loading process from AWS S3 to Snowflake in near real-time.*

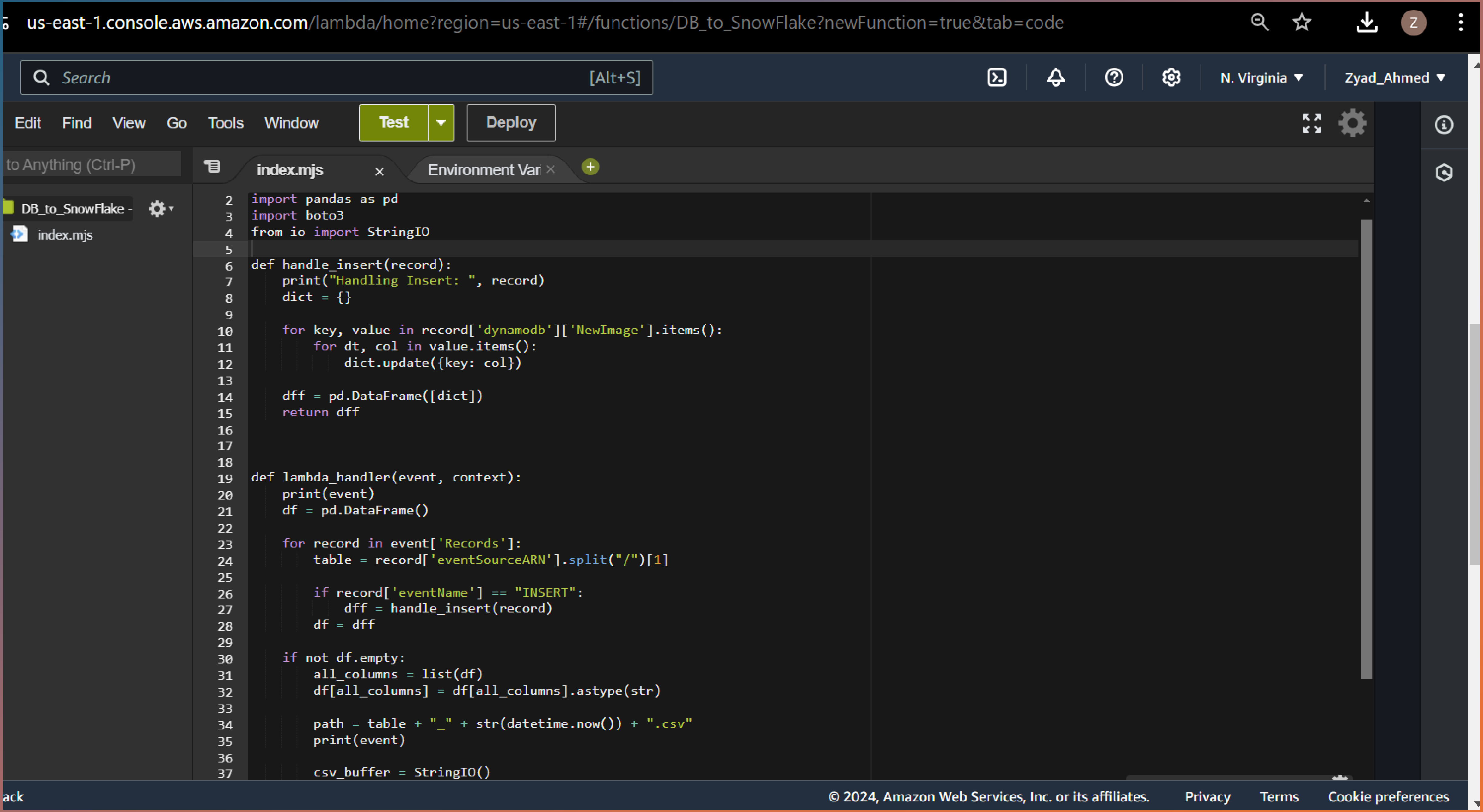
Snowflake: *Used for storing, querying, and analyzing the data.*

aws Lambda


us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#/functions/Fetching_data?newFunction=true&tab=code
Search [Alt+S]
N. Virginia
Zyad_Ahmed
File Edit Find View Go Tools Window
Test
Deploy
Go to Anything (Ctrl-P)
lambda_function
Environment Var
Fetching_data - /
lambda_function.py

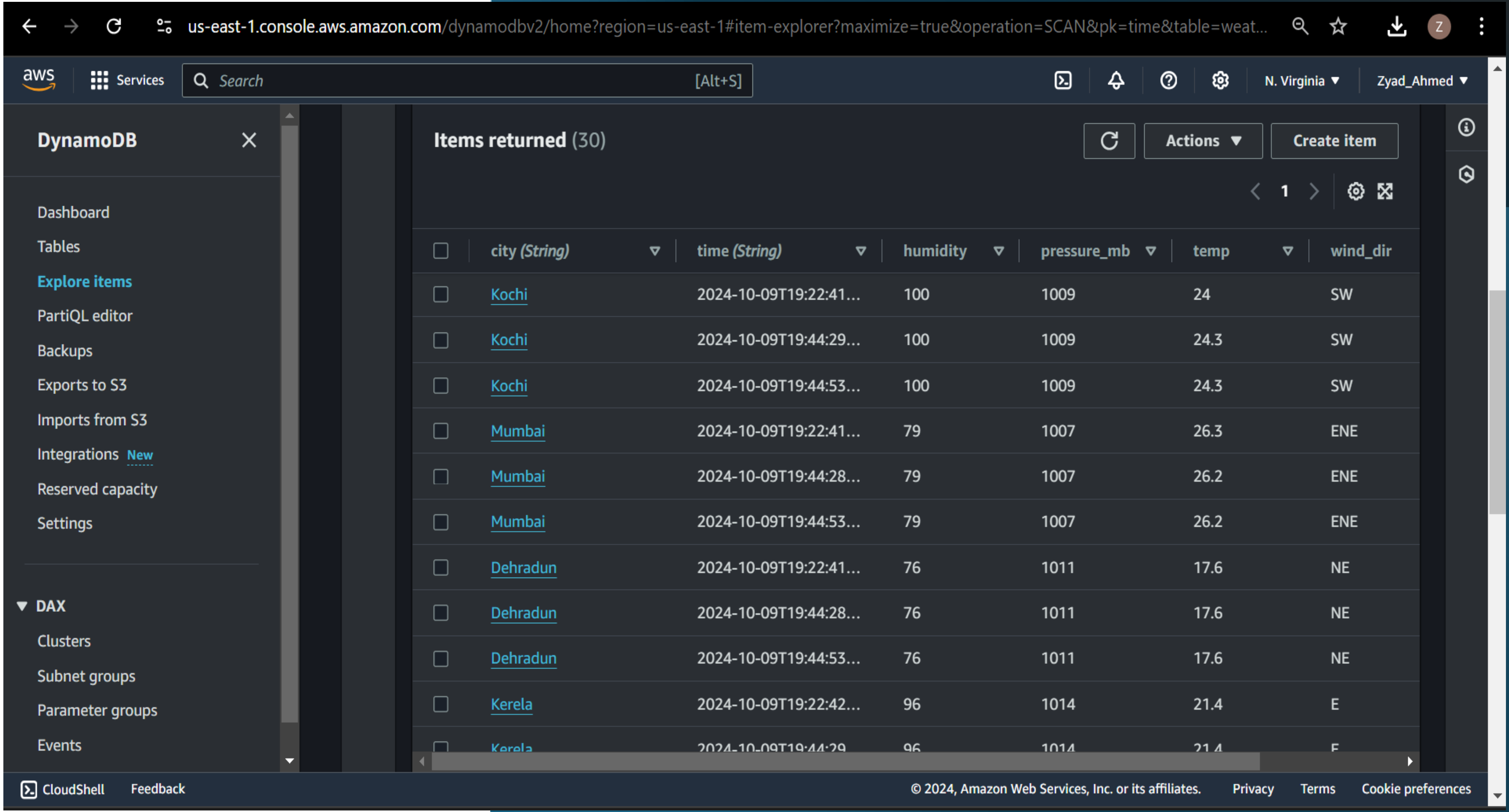
```python
import json
from datetime import datetime
import requests
import boto3
from decimal import Decimal

dynamodb = boto3.resource('dynamodb')
table = dynamodb.Table('weather')

def get_weather_data(city):
  api_url = "http://api.weatherapi.com/v1/forecast.json"
  params = {
    "q" : city,
    "Key" : "787a18fc254c425ead7145323240910"
  }

  response = requests.get(api_url, params=params)
  data = response.json()
  return data

def lambda_handler(event, context):
  cities = ["Bangalore","Delhi","Mumbai","Chennai","Kashmir","Dehradun","Kochi","Kerela","Hyderabad","Sikkim"]
  for city in cities:
    data = get_weather_data(city)

    temp = data['current']['temp_c']
    wind_speed = data['current']['wind_mph']
    wind_dir = data['current']['wind_dir']
    pressure_mb = data['current']['pressure_mb']
    humidity = data['current']['humidity']

    print(city,temp,wind_speed,wind_dir,pressure_mb,humidity)
    current_timestamp = datetime.utcnow().isoformat()

    item = {
```

eedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences


us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#/functions/DB_to_SnowFlake?newFunction=true&tab=code
Search [Alt+S]
N. Virginia
Zyad_Ahmed
Edit Find View Go Tools Window
Test
Deploy
to Anything (Ctrl-P)
index.mjs
Environment Var
DB_to_SnowFlake -
index.mjs
ack
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

```
import pandas as pd
import boto3
from io import StringIO

def handle_insert(record):
    print("Handling Insert: ", record)
    dict = {}

    for key, value in record['dynamodb']['NewImage'].items():
        for dt, col in value.items():
            dict.update({key: col})

    dff = pd.DataFrame([dict])
    return dff



def lambda_handler(event, context):
    print(event)
    df = pd.DataFrame()

    for record in event['Records']:
        table = record['eventSourceARN'].split("/")[1]

        if record['eventName'] == "INSERT":
            dff = handle_insert(record)
        df = dff

    if not df.empty:
        all_columns = list(df)
        df[all_columns] = df[all_columns].astype(str)

        path = table + "_" + str(datetime.now()) + ".csv"
        print(event)

        csv_buffer = StringIO()
```

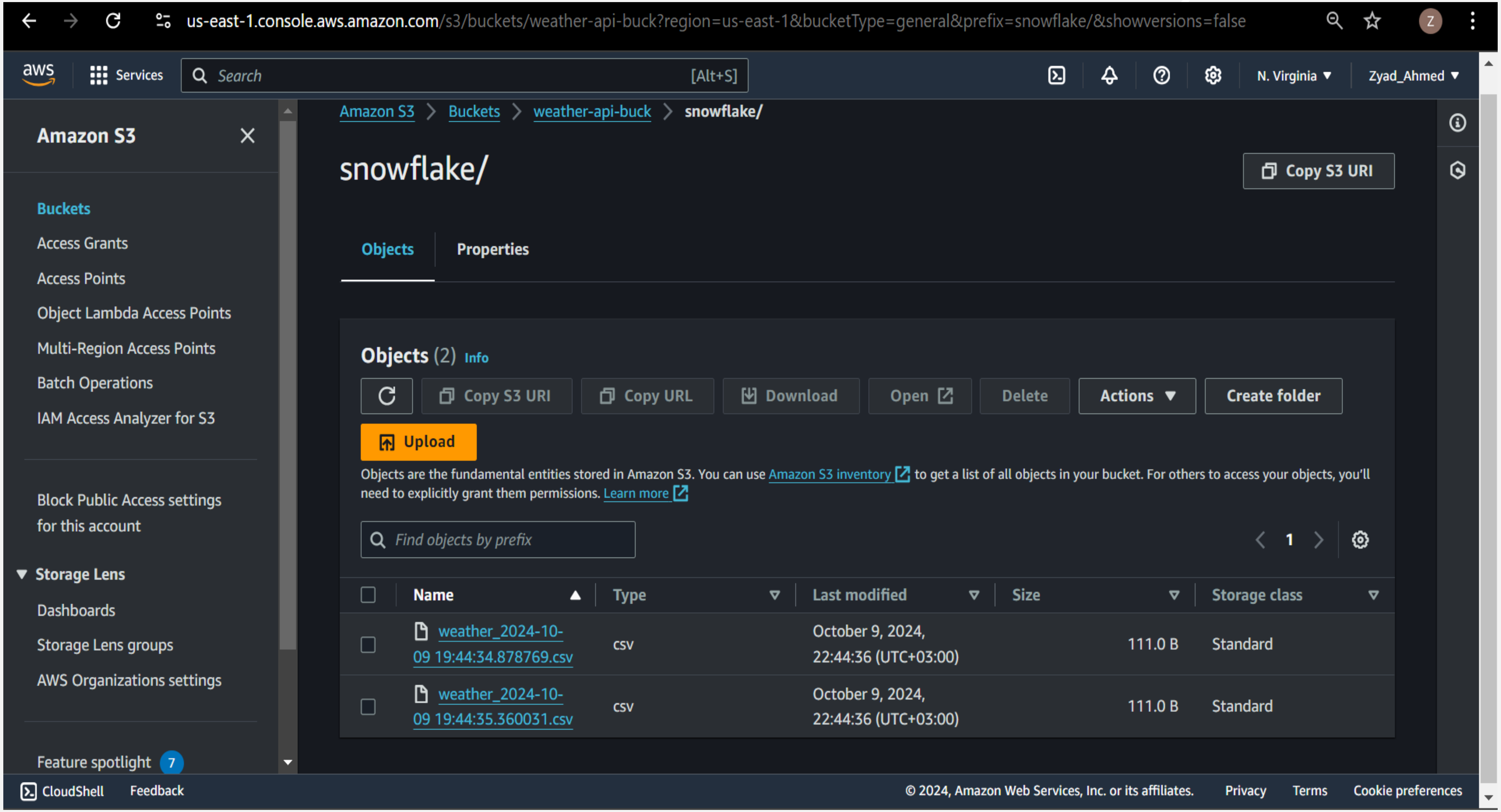
amazon
DynamoDB



Items returned (30)

city (String)	time (String)	humidity	pressure_mb	temp	wind_dir
Kochi	2024-10-09T19:22:41...	100	1009	24	SW
Kochi	2024-10-09T19:44:29...	100	1009	24.3	SW
Kochi	2024-10-09T19:44:53...	100	1009	24.3	SW
Mumbai	2024-10-09T19:22:41...	79	1007	26.3	ENE
Mumbai	2024-10-09T19:44:28...	79	1007	26.2	ENE
Mumbai	2024-10-09T19:44:53...	79	1007	26.2	ENE
Dehradun	2024-10-09T19:22:41...	76	1011	17.6	NE
Dehradun	2024-10-09T19:44:28...	76	1011	17.6	NE
Dehradun	2024-10-09T19:44:53...	76	1011	17.6	NE
Kerela	2024-10-09T19:22:42...	96	1014	21.4	E
Kerela	2024-10-09T19:44:29	96	1014	21.4	E

amazon
S3


us-east-1.console.aws.amazon.com/s3/buckets/weather-api-buck?region=us-east-1&bucketType=general&prefix=snowflake/&showversions=false
Services
Search
[Alt+S]
N. Virginia
Zyad_Ahmed
Amazon S3
Buckets
Access Grants
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
Storage Lens groups
AWS Organizations settings
Feature spotlight 7
Amazon S3 > Buckets > weather-api-buck > snowflake/
Copy S3 URI
Objects
Properties
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
Find objects by prefix
CloudShell
Feedback
Privacy
Terms
Cookie preferences

snowflake/

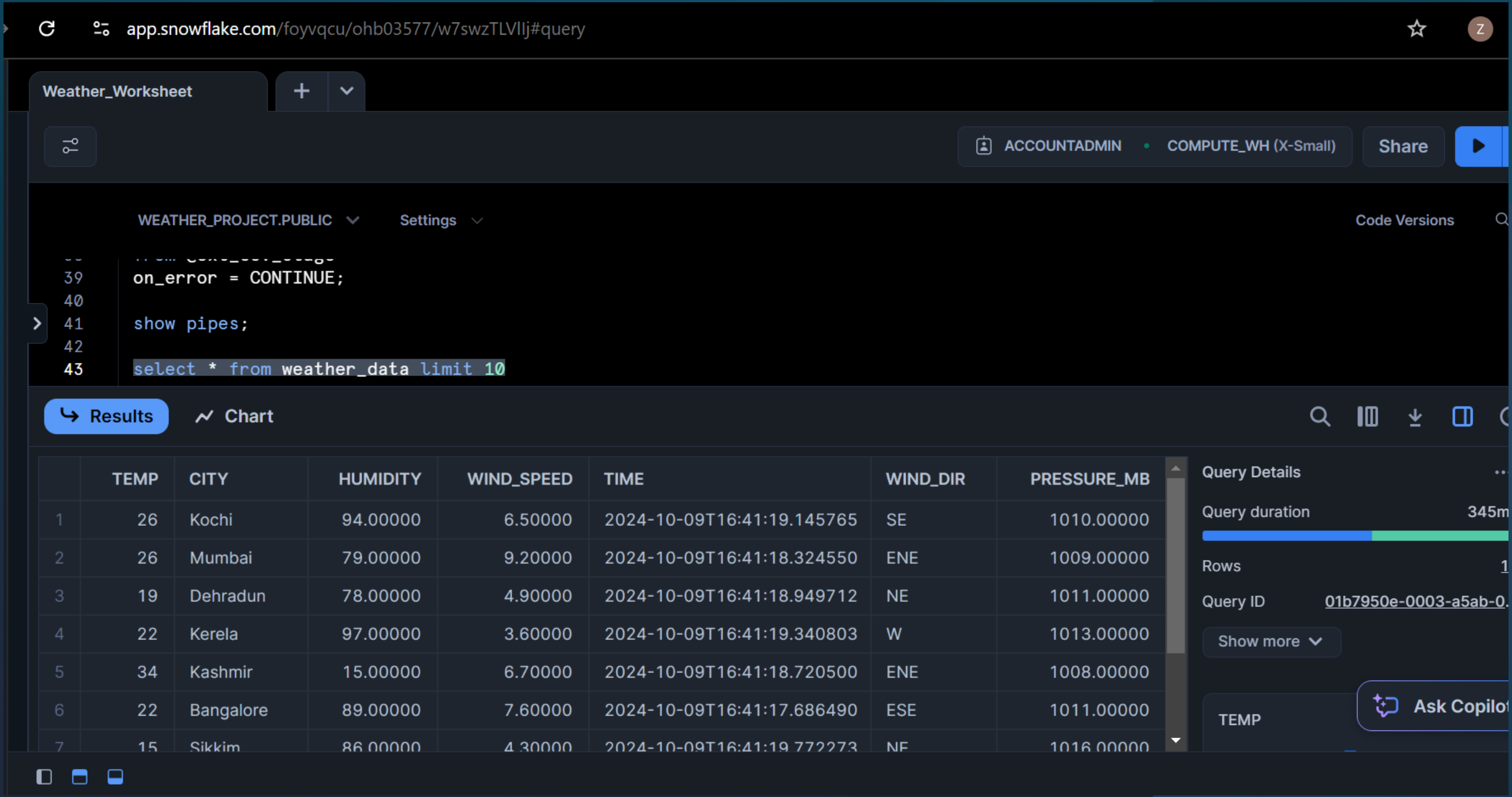
Objects (2) Info

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Name	Type	Last modified	Size	Storage class
weather_2024-10-09 19:44:34.878769.csv	csv	October 9, 2024, 22:44:36 (UTC+03:00)	111.0 B	Standard
weather_2024-10-09 19:44:35.360031.csv	csv	October 9, 2024, 22:44:36 (UTC+03:00)	111.0 B	Standard

© 2024, Amazon Web Services, Inc. or its affiliates.

snowflake®



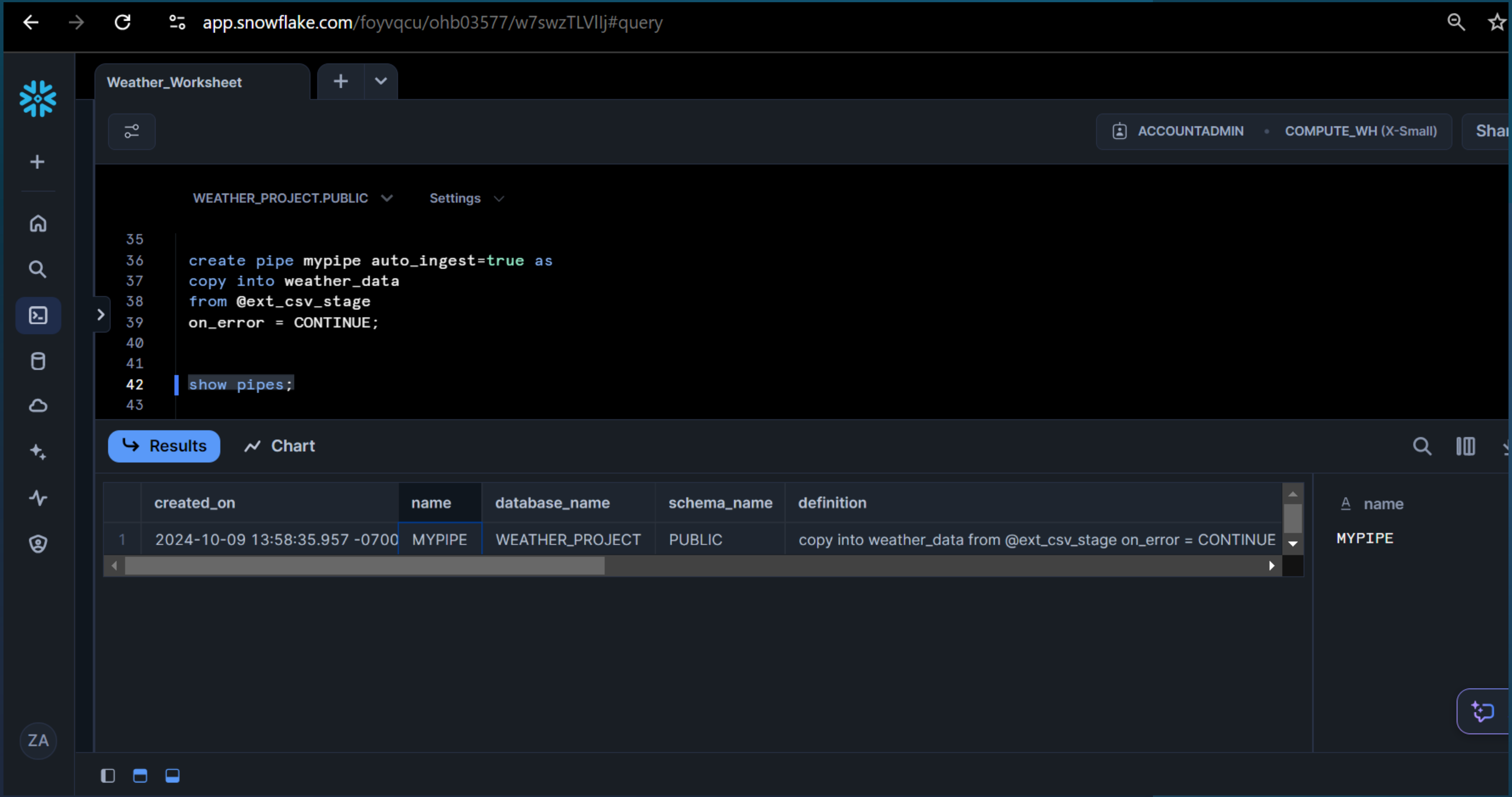
```
on_error = CONTINUE;

show pipes;

select * from weather_data limit 10
```

	TEMP	CITY	HUMIDITY	WIND_SPEED	TIME	WIND_DIR	PRESSURE_MB
1	26	Kochi	94.00000	6.50000	2024-10-09T16:41:19.145765	SE	1010.00000
2	26	Mumbai	79.00000	9.20000	2024-10-09T16:41:18.324550	ENE	1009.00000
3	19	Dehradun	78.00000	4.90000	2024-10-09T16:41:18.949712	NE	1011.00000
4	22	Kerela	97.00000	3.60000	2024-10-09T16:41:19.340803	W	1013.00000
5	34	Kashmir	15.00000	6.70000	2024-10-09T16:41:18.720500	ENE	1008.00000
6	22	Bangalore	89.00000	7.60000	2024-10-09T16:41:17.686490	ESE	1011.00000
7	15	Sikkim	86.00000	4.30000	2024-10-09T16:41:19.772273	NE	1016.00000

Query Details

Query duration	345m
Rows	1
Query ID	01b7950e-0003-a5ab-0.

TEMP



app.snowflake.com/foyvqcu/ohb03577/w7swzTLVlIj#query

Weather_Worksheet

ACCOUNTADMIN • COMPUTE_WH (X-Small)

WEATHER_PROJECT.PUBLIC   Settings

```
create pipe mypipe auto_ingest=true as
copy into weather_data
from @ext_csv_stage
on_error = CONTINUE;


show pipes;
```

Results   Chart

	created_on	name	database_name	schema_name	definition
1	2024-10-09 13:58:35.957 -0700	MYPIPE	WEATHER_PROJECT	PUBLIC	copy into weather_data from @ext_csv_stage on_error = CONTINUE

name

MYPIPE