Introduction to Big Data
Graded Assignment 5

Name - Avijeet Palit
Roll - 21f1005675
Date - 09/11/2024

Assignment Question

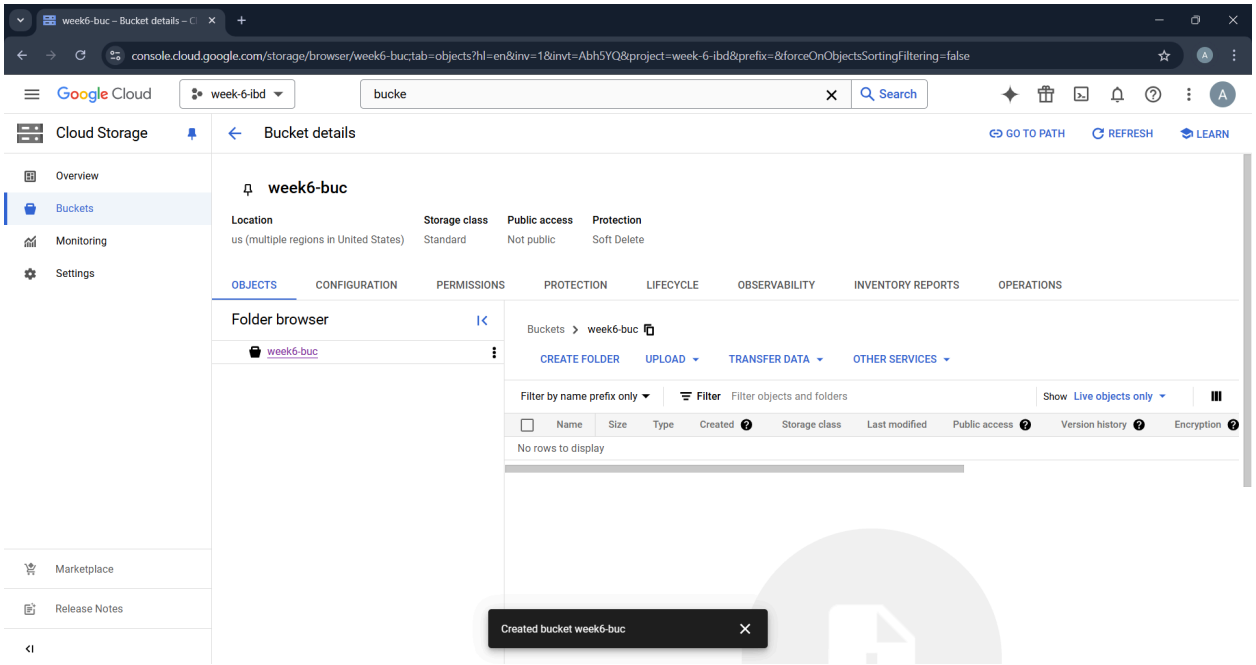
Write SparkSQL code to implement Slowly Changing Dimension (SCD) Type II on a customer master data frame. The SparkSQL code should be executed on a Dataproc cluster using the input files created for Assignment 4.

Implementation Steps

Step 1: Create a Cloud Storage Bucket

Google Cloud Storage (GCS) is used to store the input and output files for Spark jobs.

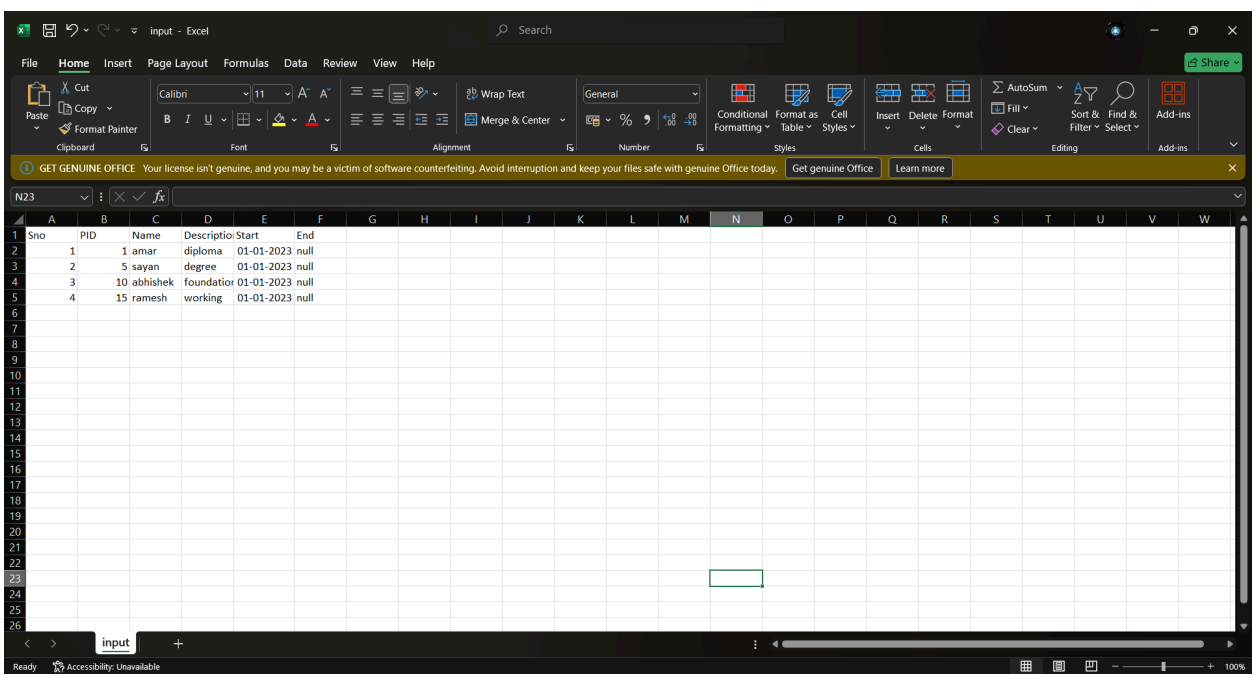
1. Navigate to Google Cloud Console.
2. Click on “Create Bucket.”
3. Name the bucket (e.g., `week6-buc`).



4. Use the default options to complete bucket creation.

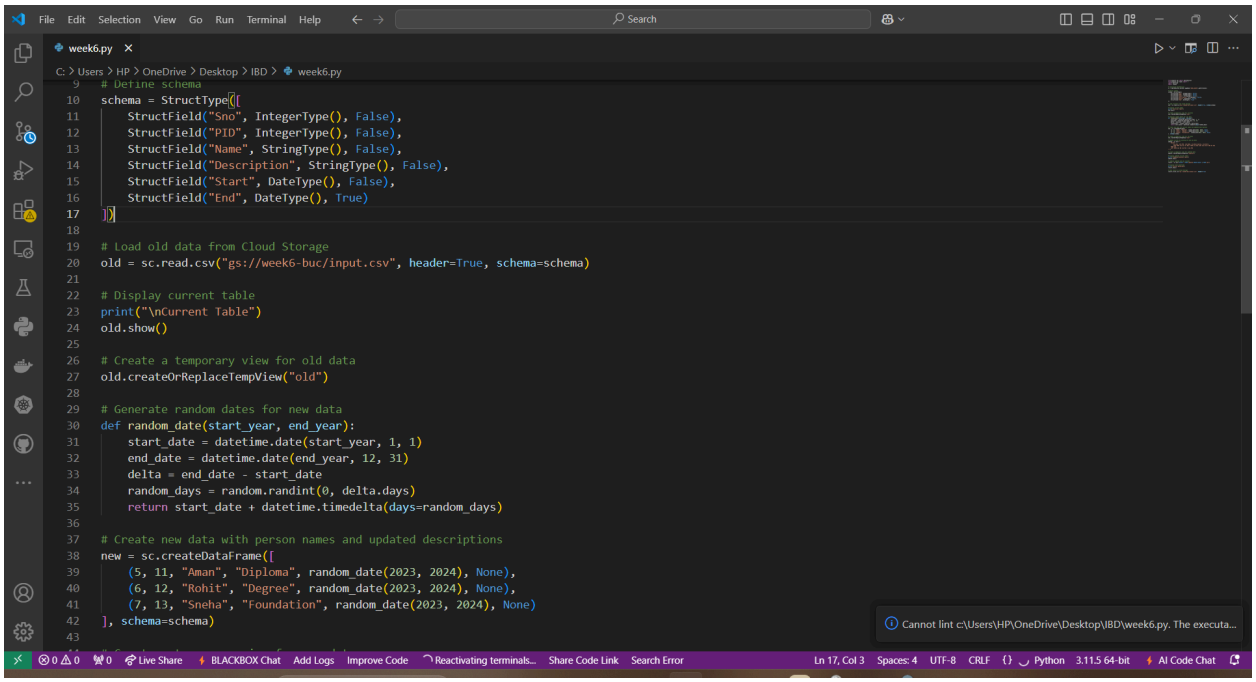
Step 2: Prepare Input File

Create a CSV file manually to serve as the input for the customer master data frame processing.



Step 3: Write PySpark Code

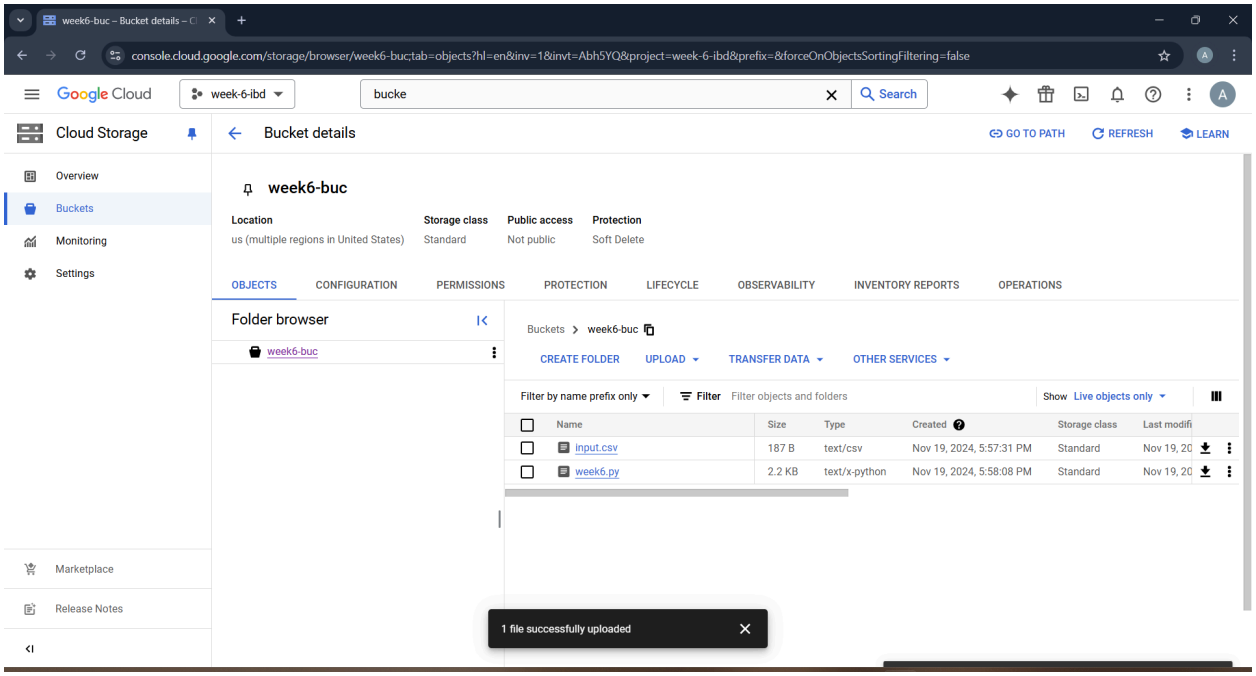
Develop PySpark code to implement SCD Type II logic on the customer master data frame. Save the code in a file named `week6.py`.



```python
# Define schema
schema = StructType([
    StructField("Sno", IntegerType(), False),
    StructField("PID", IntegerType(), False),
    StructField("Name", StringType(), False),
    StructField("Description", StringType(), False),
    StructField("Start", DateType(), False),
    StructField("End", DateType(), True)
])

# Load old data from Cloud Storage
old = sc.read.csv("gs://week6-buc/input.csv", header=True, schema=schema)

# Display current table
print("\nCurrent Table")
old.show()

# Create a temporary view for old data
old.createOrReplaceTempView("old")

# Generate random dates for new data
def random_date(start_year, end_year):
    start_date = datetime.date(start_year, 1, 1)
    end_date = datetime.date(end_year, 12, 31)
    delta = end_date - start_date
    random_days = random.randint(0, delta.days)
    return start_date + datetime.timedelta(days=random_days)

# Create new data with person names and updated descriptions
new = sc.createDataFrame([
    (5, 11, "Aman", "Diploma", random_date(2023, 2024), None),
    (6, 12, "Rohit", "Degree", random_date(2023, 2024), None),
    (7, 13, "Sneha", "Foundation", random_date(2023, 2024), None)
], schema=schema)
```

Step 4: Upload Files to Cloud Storage

1. Open the created bucket (`week6-buc`).
2. Upload the required files (`input.csv` and `week6.py`) into the bucket.

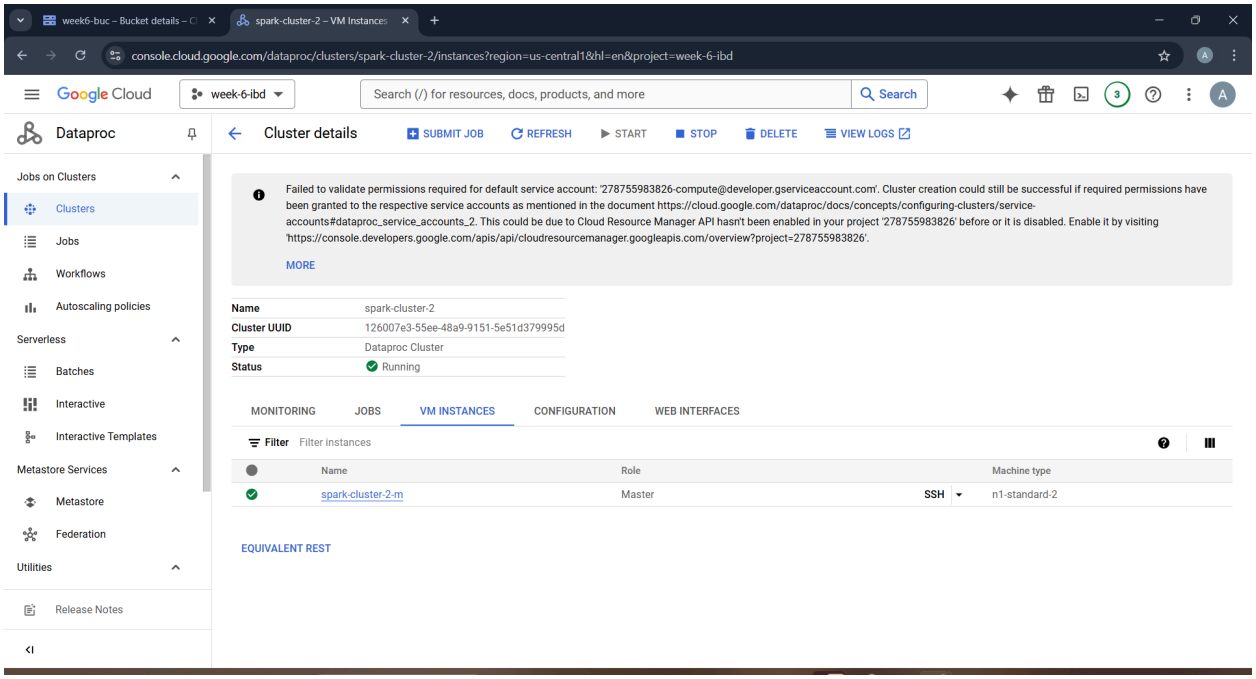


Step 5: Create a Dataproc Cluster

1. Navigate to **Dataproc** in the Google Cloud Console.
   - Go to the "Big Data" section on the left-hand menu or search for "Dataproc."
2. Click “Create Cluster” and configure the following settings:
   - **Cluster Name:** e.g., `spark-cluster`.
   - **Region:** Closest region, e.g., `us-central1`.
   - **Zone:** Leave as "No preference" if unsure.
   - **Cluster Mode:**
     - Use **Standard** for multi-role clusters (for large jobs).
     - Use **Single Node** for testing purposes.
   - **Machine Type:**
     - For small/test jobs: `n1-standard-2` or `n1-standard-4`.
   - **Worker Nodes:** Specify 2-4 nodes for small to medium workloads.
   - **Components:** Ensure PySpark is selected (default).
3. Click “Create” to initialize the cluster.

Step 6: Run Jobs on Dataproc

1. SSH into the cluster and run the following commands:

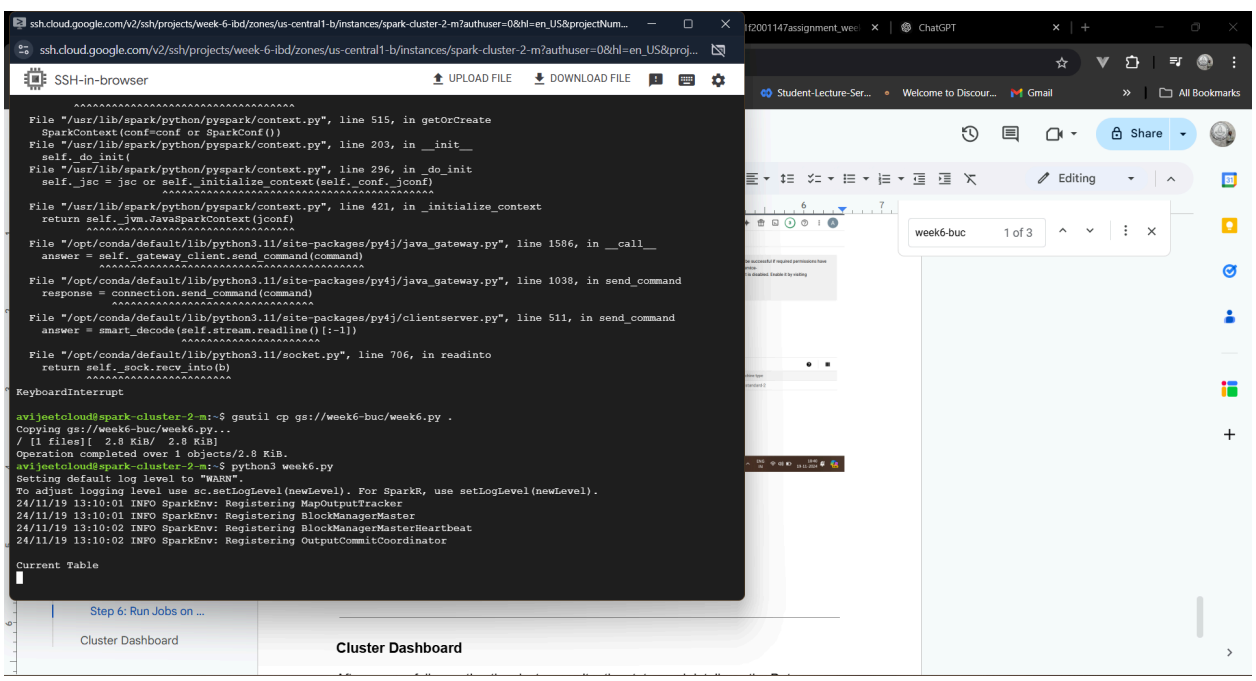


Copy the PySpark script from GCS:

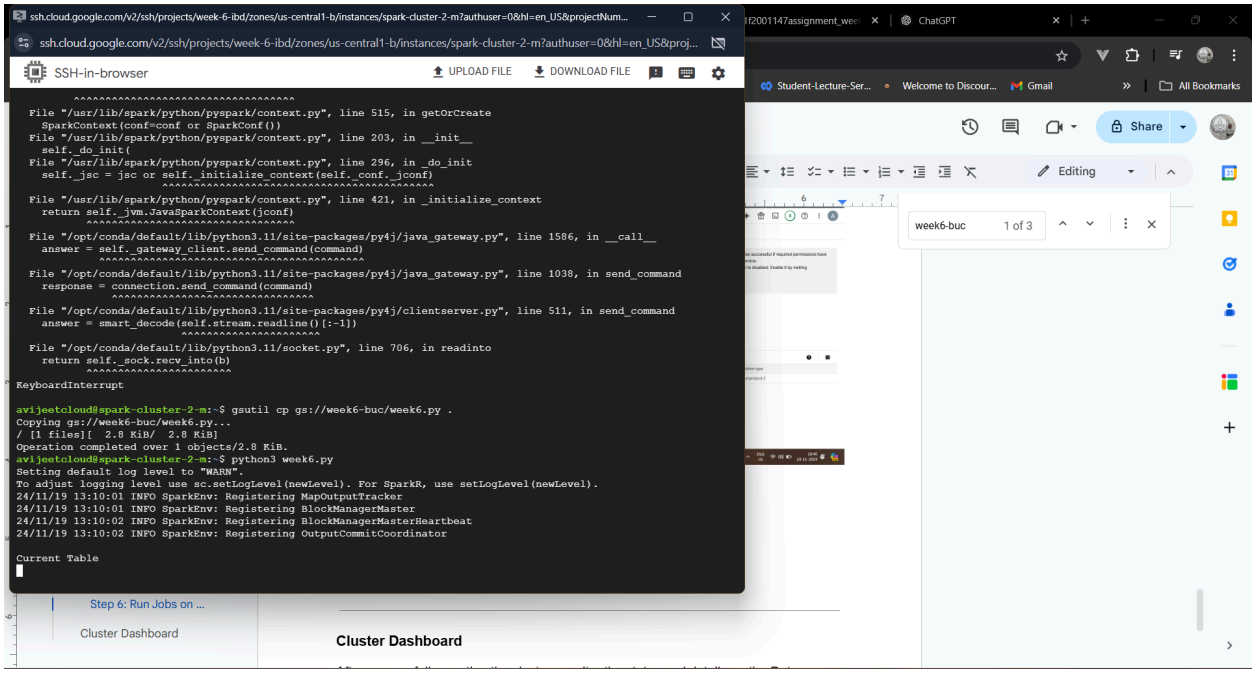
```
gsutil cp gs://week6-buc/week6.py .
```

Execute the PySpark script:

```
python3 week6.py
```

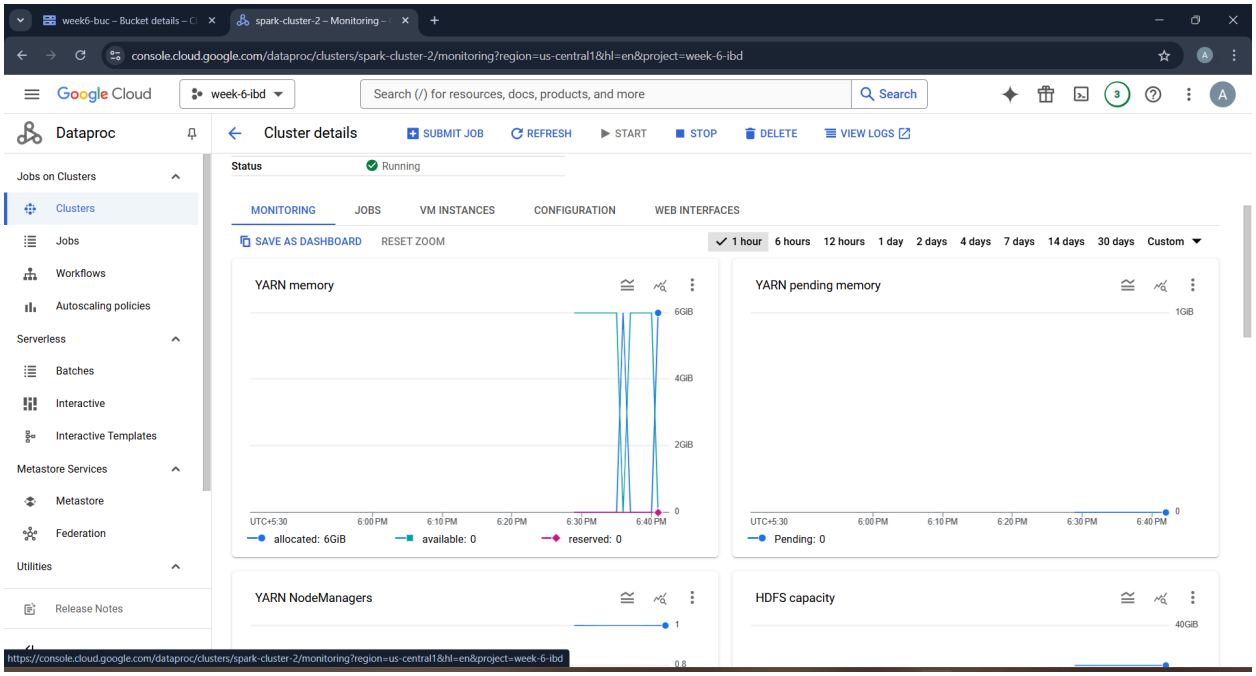


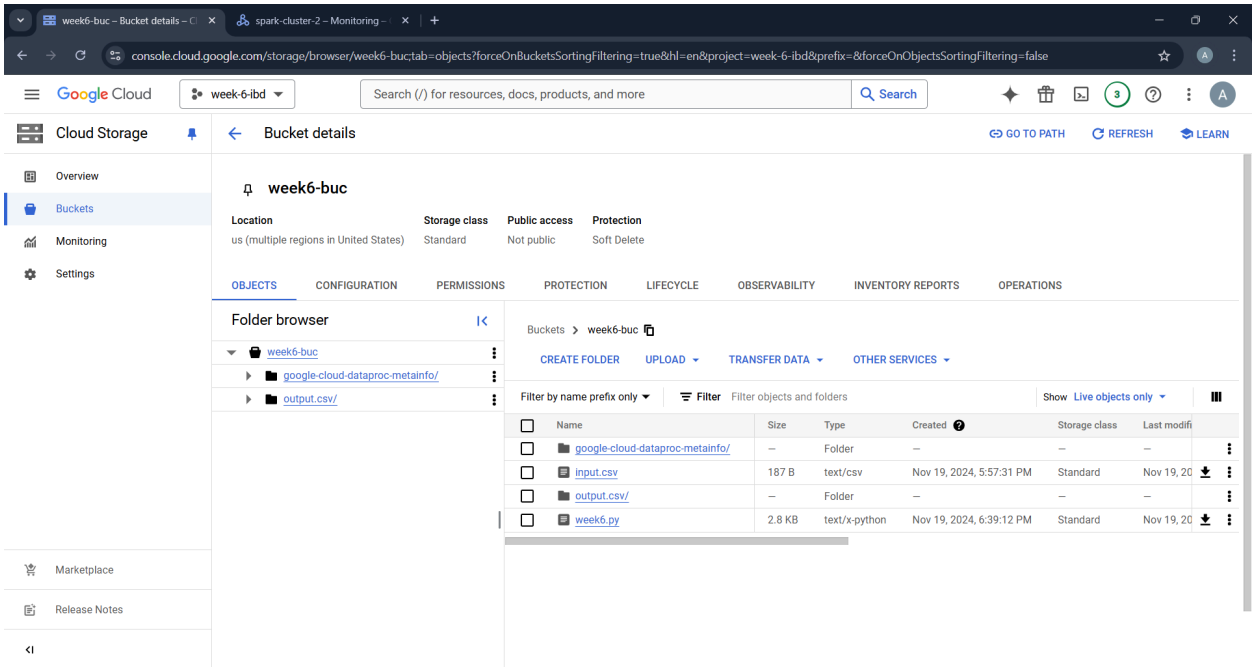
2. View the output generated by the PySpark code.



Cluster Dashboard

After successfully creating the cluster, monitor the status and details on the Dataproc Clusters page. The status should display **Running** when the cluster is ready for use.


week6-buc – Bucket details
spark-cluster-2 – Monitoring
console.cloud.google.com/dataproc/clusters/spark-cluster-2/monitoring?region=us-central1&hl=en&project=week-6-ibd
Google Cloud
week-6-ibd
Search (/) for resources, docs, products, and more
Search
Dataproc
Cluster details
SUBMIT JOB
REFRESH
START
STOP
DELETE
VIEW LOGS
Jobs on Clusters
Clusters
Jobs
Workflows
Autoscaling policies
Serverless
Batches
Interactive
Interactive Templates
Metastore Services
Metastore
Federation
Utilities
Release Notes
Status
Running
MONITORING
JOBS
VM INSTANCES
CONFIGURATION
WEB INTERFACES
SAVE AS DASHBOARD
RESET ZOOM
1 hour
6 hours
12 hours
1 day
2 days
4 days
7 days
14 days
30 days
Custom
YARN memory
6GiB
4GiB
2GiB
0
UTC+5:30
6:00 PM
6:10 PM
6:20 PM
6:30 PM
6:40 PM
allocated: 6GiB
available: 0
reserved: 0
YARN pending memory
1GiB
Pending: 0
YARN NodeManagers
HDFS capacity
40GiB
https://console.cloud.google.com/dataproc/clusters/spark-cluster-2/monitoring?region=us-central1&hl=en&project=week-6-ibd


week6-buc – Bucket details
spark-cluster-2 – Monitoring
console.cloud.google.com/storage/browser/week6-buc;tab=objects?forceOnBucketsSortingFiltering=true&hl=en&project=week-6-ibd&prefix=&forceOnObjectsSortingFiltering=false
Google Cloud
week-6-ibd
Search (/) for resources, docs, products, and more
Search
Cloud Storage
Bucket details
GO TO PATH
REFRESH
LEARN
Overview
Buckets
Monitoring
Settings
week6-buc
Location
us (multiple regions in United States)
Storage class
Standard
Public access
Not public
Protection
Soft Delete
OBJECTS
CONFIGURATION
PERMISSIONS
PROTECTION
LIFECYCLE
OBSERVABILITY
INVENTORY REPORTS
OPERATIONS
Folder browser
week6-buc
google-cloud-dataproc-metainfo/
output.csv/
Buckets > week6-buc
CREATE FOLDER
UPLOAD
TRANSFER DATA
OTHER SERVICES
Filter by name prefix only
Filter
Filter objects and folders
Show
Live objects only
Name
Size
Type
Created
Storage class
Last modifi
google-cloud-dataproc-metainfo/
Folder
input.csv
187 B
text/csv
Nov 19, 2024, 5:57:31 PM
Standard
output.csv/
week6.py
2.8 KB
text/x-python
Nov 19, 2024, 6:39:12 PM
Marketplace
Release Notes