Homework 2: Getting Started with Spark and Climate Change
Author: Shivani Tayade

Setting Up Spark in Google Colab & Replicating Chapter 2 Example

To begin the assignment, we set up our Spark environment using Google Colab, as recommended. Colab allows easy integration with PySpark for lightweight big data tasks. Below are the steps we followed:

1. Spark Setup on Google Colab:

- We initially installed Java and Spark dependencies via `apt-get` and `pip`, but later commented out these steps after confirming the Colab runtime had compatible versions preinstalled.

```
# Install Java (required for Spark)
!apt-get install openjdk-8-jdk-headless -qq > /dev/null

# Download and extract Apache Spark 3.5.0 with Hadoop 3
!wget -q https://archive.apache.org/dist/spark/spark-3.5.0/spark-3.5.0-bin-hadoop3.tgz
!tar xf spark-3.5.0-bin-hadoop3.tgz

# Install required Python packages
!pip install -q findspark
!pip install pyspark==3.5.5
```

- Spark and Java configuration-

```
import os

# Automatically detect Java version path (Java 8 or 11, whichever is installed)
java_8_path = "/usr/lib/jvm/java-8-openjdk-amd64"
java_11_path = "/usr/lib/jvm/java-11-openjdk-amd64"

# Use whichever Java version is available
if os.path.exists(java_8_path):
    os.environ["JAVA_HOME"] = java_8_path
elif os.path.exists(java_11_path):
    os.environ["JAVA_HOME"] = java_11_path
else:
    raise EnvironmentError("No compatible Java version found.")

# Set Spark path
os.environ["SPARK_HOME"] = "/content/spark-3.5.0-bin-hadoop3"
```

```
# Initialize Spark
import findspark
findspark.init()

from pyspark.sql import SparkSession

# Customize Spark Configuration
spark = SparkSession.builder \
    .appName("Homework2_Spark_Chapter2") \
    .master("local[2]") \
    .config("spark.executor.memory", "1g") \
    .config("spark.driver.memory", "1g") \
    .getOrCreate()
```

2. Chapter 2 -

- We loaded flight summary CSV files (2010–2015) using a wildcard pattern
    - **df = spark.read.csv("*.csv", header=True, inferSchema=True)**
- We performed transformations like sorting, aggregation, and SQL queries, using both:
    - DataFrame API
    - Spark SQL (via `createOrReplaceTempView`)

- We added a new column to extract the year from filenames and conducted grouped analyses to identify top destinations per year.

The google colab link for the above steps is
https://colab.research.google.com/drive/1CeaGsDKNGqKmgQcRrHta9-Oky374Jtz4

3. Running Pyspark on Google Cloud Dataproc

To scale our Spark workflow and prepare for working with larger datasets, we deployed our code to Google Cloud Dataproc. Below are the steps we followed:

- Set Up Google Cloud Platform (GCP): The account has been set up with the organization details as 'ucdavis.edu
- Created a GCP project: **big-data-msba**
- After the GCP project has been created, Google cloud SDK has been installed. The reference file has been download https://cloud.google.com/sdk/docs/install
- SDK has been initialized through Homebrew, following the below steps-

```
/bin/bash -c "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

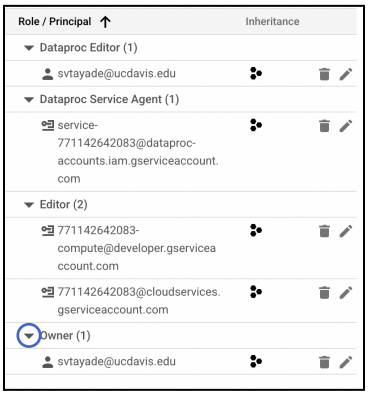
- After installation finishes, run this to update your PATH

```
echo 'eval "$($(which brew) shellenv)"' >> ~/.zprofile
eval "$($(which brew) shellenv)"
```

- Now, install Google Cloud SDK using homebrew.
- After Initialising the SDK, link the GCP account/pass the GCP credentials in the terminal
- Setup the **IAM** permissions and assign the below roles

```
gcloud projects add-iam-policy-binding PROJECT_ID \
  --member="user:student_email@ucdavis.edu" \
  --role="roles/dataproc.editor"
```

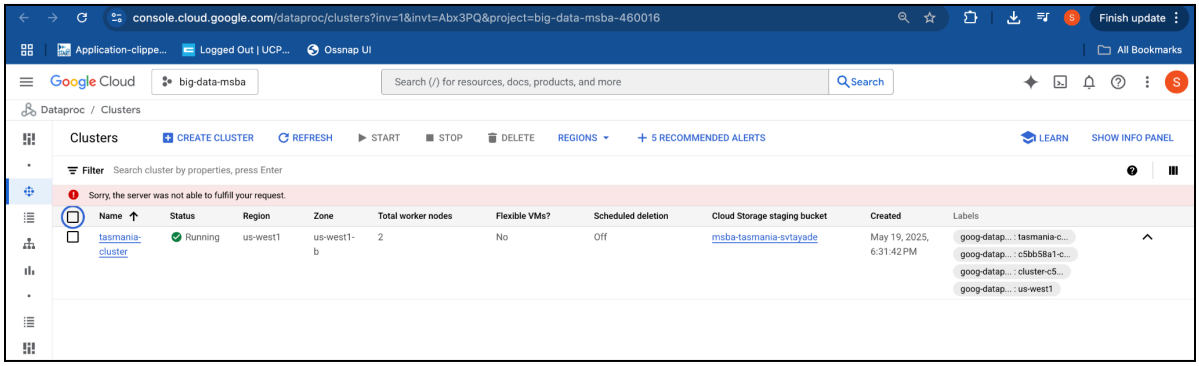
- Permissions screenshot under the cluster-



Homework 2: Getting Started with Spark and Climate Change
Author: Shivani Tayade

- Enable the APIs and Create GCS bucket.

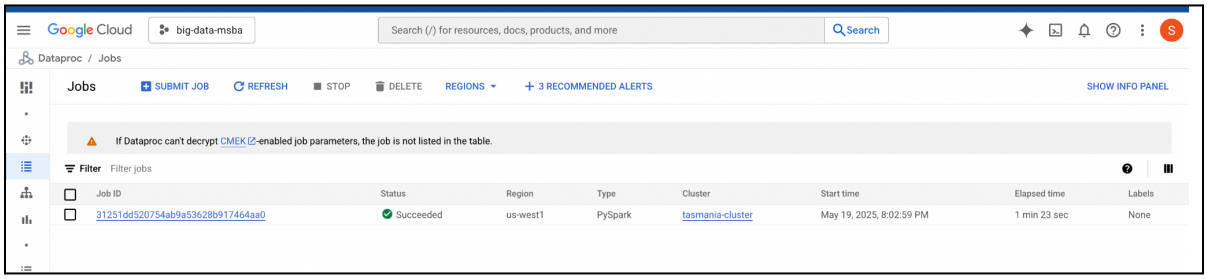
- Create a low cost spark cluster-



- Submit the **Spark** job
We encountered multiple challenges while running the Spark job on Google Cloud Dataproc and resolved them systematically:

Challenge	Resolution
CSV wildcard failed on Colab	Used Dataproc with GCS path wildcards like 201*-summary.csv
Java & Spark setup issues in Colab	Skipped entirely in Dataproc due to pre-installed runtime
Incorrect column references	Fixed logic to match actual dataset schema (e.g., Country not DEST_COUNTRY_NAME)
GCS file not found errors	Double-checked file paths and confirmed uploads to correct bucket
Mixed schema types	Added inferSchema=True and validated columns via printSchema()

After considering the above changes, the job has been successfully executed. Please refer to the below snapshot-



Homework 2: Getting Started with Spark and Climate Change
Author: Shivani Tayade

The google colab link for the above steps is
https://colab.research.google.com/drive/1DzhgnsyfFFDlwR_6mPUv3dTyye_3SEvF

4. Everyday activity to make a positive change for the environment

In my daily life, I've become more conscious of my habits and actively make efforts to reduce energy consumption—like turning off devices when they're not in use. I opt for recyclable items that are easier to dispose of responsibly and have shifted toward consuming more local and organic foods instead of packaged alternatives. Additionally, I try to minimize the use of electrical appliances like microwaves and dishwashers, embracing a more traditional lifestyle wherever possible. Beyond these personal choices, I stay informed about environmental issues and share what I learn with others—because awareness is a powerful tool. While I understand that individual actions alone won't solve climate change, I truly believe that small, consistent steps can inspire broader, lasting change.