Assignment 2: Writing to GCS from Databricks

This document outlines the steps taken by Sakina Banu to read a dataset from Google Cloud Storage (GCS) and write it back in three formats: CSV, JSON, and Parquet using Apache Spark in Databricks.

spark.conf.set("spark.hadoop.google.cloud.auth.service.account.enable", "true")

spark.conf.set("spark.hadoop.fs.gs.auth.service.account.email", "databricks-compute@calcium-field-455700-p2.iam.gserviceaccount.com")

spark.conf.set("spark.hadoop.fs.gs.project.id", "calcium-field-455700-p2")

spark.conf.set("spark.hadoop.fs.gs.auth.service.account.private.key", "MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQDHj4/0mF191m/B\nI7ftge8vA7zyOeHhiA/2KzT98kP4CQCP63N/12wGbv9P7OOi9gMGTMJTeeVW9FJ9\nRLjFXB/7e0YxsIYh9l8Z+QlLwZSIPPVQ9EuazpJHte9hFBmLo1iTpB1diJpWIP8g\nOCQMmt2lWOlJXgFnGkNOThO1RcE3/n9CPO/LmUhTcDHDRccN9YmiIiZFDcalN1gI\n6/+HLfd0LsOWhYaHc1dBx5u9Mnb1IHg5idyaMiTjFWYBhrFmSRQhdAFbDsPozWV+\n+IuTflEgib6xwdLgckXGCbWMzKVFCFGzmXk6pDZXUtbFEOeiieTScSy+qAc58aDX\nVsCsHFyBAgMBAAECggEADtkMgsxB24KkkLwbu5sI7Oh0KtkZ6HX/Swdys0jO3BHu\nZebksnC4p79jNWMImle/yJyWMlAN22MFDq856KxGsGc/BSQvTLBYKdZsxcNOzfas\nIXR1uhxPx7BekcxlLK2wAdcqfKq04f7lQJRQwVPV9S8Y92WcSNusQZOfVYPNwSDH\nmilIjXBQq4R2GNly1Bzq0iv4RkjObP5XhRnCjaoIglQVqSG26ueN9eCbhL62P2Ly\nIazrhHQeSztMaNwD+Ar6ffEWDxLxKwjWsEAGPcdhiyErrOGTVuBUwKkuGTQBmPqE\nlADoLJr5h96ykAYyQ3Ch3A/2KS6wkaBZ0J/EN8m6OQKBgQDnZCdX/7V6lnVQY/xm\nPDaqlnnRxX07rn0fwMN6WeLUaH0XdOZHY/KjE10zqKwnAIpIs/PSJXtMYY7bLJNy\nYN9KxWcebI7EHDmPcJ8FFXWogrFyph4jOgfqc+QstzH5N11aFhkrpNmHuAv0BeeA\nAZw/VsEKpJ1MKtQsDfyHpe5myQKBgQDcyMt1EZW+FrJiTd+qTc/A8469Icmc+bgS\nu9eOWU3iU32KLaNdh8R2jI8YziexfXB01Fqosx4JNv1I8RTUUtJcDPyayYoZdcfE\nb/oKR4gbXOWYtqWz0DJzKBja59sWeR4a2UEb8EF4j6i5bh6vpVgF+3vvj+776DWu\nieNPdXHL+QKBgQCVbf1me/Fj6FR8hQWTtGENSz4CdUpJx4Zwghj7I58z0wKtqkWo\nZuhj3fZ7jPsX+OZUyzmzRZdZnUsXSbMy01KAQ/kzEaIlDaEye+WF7/TazQPxaUvf\nLfO8ufjp06jHK7Tyn4DDyaLdXcMCZOm6pc+Uq97gk6wtax84EBQ8ObE62QKBgQCH\nNJgUGy6+eBvJw7LDEl+14/F3tWdIME0fe9UnoVzmywZIRmaS8n/HsZA4OpdoRE9H\n3HugcABm2DwIHUsKM6YLJRDUobNmoBA0JfbTsLaTOzKHcOLM2STsDJQcCnzKqGVd\nP8p1l44gjKlg1SUJn5HmJ/nf3R8XOT8Woe6ITH3h0QKBgBwOhU3P4zoBSahNzVC6\nk1AC2nKKJI72DaNZUYTSTLwmGrt5ADOfMAZQypF31lVPYnP6DFFlTNLihsvrX65K\nFc6yZmIOV2yXfIfAfQjOovz5RpAGikZOxbRKkrtghOl9WgjynJOiZ/wu//brrSEG\nBJ9nIPJ3fmYDQze4Y+8nkYDB")

spark.conf.set("spark.hadoop.fs.gs.auth.service.account.private.key.id", "cc792d2f8ef8022756e4bdf8e1f3c2105aa4ec1c")

# Step 1: Load CSV from GCS into DataFrame

The following code reads the CSV file from the 'heart\_raw-bronze' GCS bucket into a Spark DataFrame:

df = spark.read \  
 .option("header", "true") \  
 .csv("gs://heart\_raw-bronze/cars.csv")

# Step 2: Write DataFrame to GCS as CSV

The following code writes the DataFrame to GCS in CSV format with headers:

df.write \  
 .mode("overwrite") \  
 .option("header", "true") \  
 .csv("gs://heart\_raw-bronze/car\_csv\_out")

# Step 3: Write DataFrame to GCS as JSON

The following code writes the DataFrame to GCS in JSON format:

df.write \  
 .mode("overwrite") \  
 .format("json") \  
 .save("gs://heart\_raw-bronze/car\_json\_out")

# Step 4: Write DataFrame to GCS as Parquet

The following code writes the DataFrame to GCS in Parquet format:

df.write \  
 .mode("overwrite") \  
 .format("parquet") \  
 .save("gs://heart\_raw-bronze/car\_parquet\_out")

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.