databricks Data from SQL server

(https://databricks.com)

1

```
dbutils.fs.mount(
    source="wasbs://raw-sql@casestudy1new.blob.core.windows.net",
    mount_point = "/mnt/raw-sql",
    extra_configs={"fs.azure.account.key.casestudy1new.blob.core.windows.net": dbutils.secrets.get(scope =
    "casestudy", key = "storage")})
```

```
True
```

Read CSV file

3

```
# Define the path to the txt file on the mounted container
file_path = "/mnt/raw-sql/dbo.football.txt"

# Read the txt file into a DataFrame
# Assuming the file is comma-delimited (adjust the delimiter as necessary)
df = spark.read.option("delimiter", ",").csv(file_path, header=True, inferSchema=True)

# Show the DataFrame to inspect the data
df.show(10)
```

```
9 00:00:00|
|   3|          Ali Daei|                Iran|          AFC|  108| 148|0.7300000190734863|  1993-2006|2000-01-0
9 00:00:00|
|   4|     Sunil Chhetri|               India|          AFC|   94| 151|0.6200000047683716|  2005-2024|2015-12-3
1 00:00:00|
|   5|    Mokhtar Dahari|            Malaysia|          AFC|   89| 142|0.6299999952316284|  1972-1985|1976-08-2
2 00:00:00|
|   6|      Ali Mabkhout| United Arab Emir...|          AFC|   85| 115|0.7400000095367432|      2009-|2019-08-3
1 00:00:00|
|   6|     Romelu Lukaku|             Belgium|         UEFA|   85| 120|0.7099999785423279|      2010-|2019-10-1
0 00:00:00|
|   8|     Ferenc Puskás|       Hungary_Spain|         UEFA|   84|  89|0.9399999976158142|  1945-1962|1952-07-2
4 00:00:00|
|   8|Robert Lewandowski|              Poland|         UEFA|   84| 156|0.5400000214576721|      2008-|2017-10-0
5 00:00:00|
|  10|   Godfrey Chitalu|              Zambia|          CAF|   79| 111|0.7099999785423279|  1968-1980|1978-11-0
7 00:00:00|
+----+------------------+--------------------+-------------+-----+----+------------------+-----------+---------
----------+
only showing top 10 rows
```

4

```
# Get the total number of rows in the DataFrame
total_rows = df.count()

# Display the total number of rows
print(f"Total number of rows: {total_rows}")
```

```
Total number of rows: 82
```

data cleaning

6

```
# Remove rows with null values
df_cleaned = df.dropna()
```

7

```
from pyspark.sql import functions as F
from pyspark.sql.window import Window

# Step 1: Filter the rows where 'Goals' >= 80
df_cleaned = df.filter(df['Goals'] >= 80)  #

# Step 2: Re-index the 'Rank' column to make it consecutive after filtering
# using row_number() over an ordered window to reassign the 'Rank'
window_spec = Window.orderBy(F.col('Goals').desc())  # Rank higher Goals first
df_cleaned = df_cleaned.withColumn('Rank', F.row_number().over(window_spec))

# Step 3: Show the cleaned DataFrame
df_cleaned.show(10)
```

```
|   1| Cristiano Ronaldo|            Portugal|        UEFA|  135| 217|0.6200000047683716|      2003-|2014-06-2
6 00:00:00|
|   2|      Lionel Messi|           Argentina|    CONMEBOL|  112| 190|0.5899999737739563|      2005-|2016-03-2
9 00:00:00|
|   3|          Ali Daei|                Iran|         AFC|  108| 148|0.7300000190734863|  1993-2006|2000-01-0
9 00:00:00|
|   4|     Sunil Chhetri|               India|         AFC|   94| 151|0.6200000047683716|  2005-2024|2015-12-3
1 00:00:00|
|   5|    Mokhtar Dahari|            Malaysia|         AFC|   89| 142|0.6299999952316284|  1972-1985|1976-08-2
2 00:00:00|
|   6|      Ali Mabkhout| United Arab Emir...|         AFC|   85| 115|0.7400000095367432|      2009-|2019-08-3
1 00:00:00|
|   7|     Romelu Lukaku|             Belgium|        UEFA|   85| 120|0.7099999785423279|      2010-|2019-10-1
0 00:00:00|
|   8|     Ferenc Puskás|       Hungary_Spain|        UEFA|   84|  89|0.9399999976158142|  1945-1962|1952-07-2
4 00:00:00|
|   9|Robert Lewandowski|              Poland|        UEFA|   84| 156|0.5400000214576721|      2008-|2017-10-0
5 00:00:00|
+----+------------------+--------------------+------------+-----+----+------------------+-----------+---------
----------+
```

8

```
# Check the total number of rows in the cleaned DataFrame
total_rows_cleaned = df_cleaned.count()

# Print the total number of rows
print(f"Total rows in the cleaned DataFrame: {total_rows_cleaned}")
```

```
Total rows in the cleaned DataFrame: 9
```

Save cleaned DataFrame to processed container

10

```
dbutils.fs.mount(
    source="wasbs://processed-sql@casestudy1new.blob.core.windows.net",
    mount_point = "/mnt/processed-sql",
    extra_configs={"fs.azure.account.key.casestudy1new.blob.core.windows.net": dbutils.secrets.get(scope =
    "casestudy", key = "storage")})
```

```
True
```

11

```
# Define the path to the 'processed-sql' container
processed_container_path = "/mnt/processed-sql/dbo_football_cleaned"

# Save the cleaned DataFrame to the 'processed-sql' container using 'append' mode
df_cleaned.write.mode('append').parquet(processed_container_path)
```

For staging

13

```
dbutils.fs.mount(
    source="wasbs://staging-sql@casestudy1new.blob.core.windows.net",
    mount_point = "/mnt/staging-sql",
    extra_configs={"fs.azure.account.key.casestudy1new.blob.core.windows.net": dbutils.secrets.get(scope =
    "casestudy", key = "storage")})
```

```
True
```

14