AWS Glue Studio Notebook

You are now running a AWS Glue Studio notebook; To start using your notebook you need to start an AWS Glue Interactive Session.

Optional: Run this cell to see available notebook commands ("magics").

In [ ]:

```
%help
```

Run this cell to set up and start your interactive session.

In [1]:

```
%idle_timeout 100
%glue_version 3.0
%worker_type G.1X
%number_of_workers 2

import sys
from awsglue.transforms import *
from awsglue.utils import getResolvedOptions
from pyspark.context import SparkContext
from awsglue.context import GlueContext
from awsglue.job import Job

sc = SparkContext.getOrCreate()
glueContext = GlueContext(sc)
spark = glueContext.spark_session
job = Job(glueContext)
```

```
Welcome to the Glue Interactive Sessions Kernel
For more information on available magic commands, please type %help in any new cell.

Please view our Getting Started page to access the most up-to-date information on the Int
eractive Sessions kernel: https://docs.aws.amazon.com/glue/latest/dg/interactive-sessions
.html
Installed kernel version: 0.37.3
Current idle_timeout is 2800 minutes.
idle_timeout has been set to 100 minutes.
Setting Glue version to: 3.0
Previous worker type: G.1X
Setting new worker type to: G.1X
Previous number of workers: 5
Setting new number of workers to: 2
Authenticating with environment variables and user-defined glue_role_arn: arn:aws:iam::68
7003041478:role/orka-glue-role
Trying to create a Glue session for the kernel.
Worker Type: G.1X
Number of Workers: 2
Session ID: a82ae804-f97a-44ef-8b27-bdd0d615a2bc
Job Type: glueetl
Applying the following default arguments:
--glue_kernel_version 0.37.3
--enable-glue-datacatalog true
Waiting for session a82ae804-f97a-44ef-8b27-bdd0d615a2bc to get into ready status...
Session a82ae804-f97a-44ef-8b27-bdd0d615a2bc has been created.
```

In [2]:

```
import sqlite3
import boto3
```

```
import pandas as pd

# Set your S3 bucket and file path
bucket_name = 'iplcricketinfo'
file_path = 'input_files/IPL_Deliveries.sqlite'

# Initialize the S3 client
s3 = boto3.client('s3')

# Download the file from S3
s3.download_file(bucket_name, file_path, '/tmp/IPL_Deliveries.sqlite')

# Perform further processing or analysis on the downloaded file
con = sqlite3.connect("/tmp/IPL_Deliveries.sqlite")

# Execute a SQL query and fetch the results into a pandas DataFrame
query = "SELECT * FROM deliveries"
df = pd.read_sql_query(query, con)

# Convert the pandas DataFrame to a Spark DataFrame
spark_df = spark.createDataFrame(df)

# Perform further processing or analysis on the Spark DataFrame
spark_df.show()
```

```
+--------+------+------------------+--------------------+----+----+-----------+------
------+----------+-------------+---------+--------+----------+----------+-----------
-+-----------+---------+----------+---------------+-------------+-------------+
|match_id|inning|      batting_team|        bowling_team|over|ball|     batsman| non_str
iker|     bowler|is_super_over|wide_runs|bye_runs|legbye_runs|noball_runs|penalty_runs|ba
tsman_runs|extra_runs|total_runs|player_dismissed|dismissal_kind|      fielder|
+--------+------+------------------+--------------------+----+----+-----------+------
------+----------+-------------+---------+--------+----------+----------+-----------
-+-----------+---------+----------+---------------+-------------+-------------+
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   1|   1|   DA Warner|    S Dh
awan|   TS Mills|            0|        0|       0|          0|          0|           0|
0|          0|         0|           null|          null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   1|   2|   DA Warner|    S Dh
awan|   TS Mills|            0|        0|       0|          0|          0|           0|
0|          0|         0|           null|          null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   1|   3|   DA Warner|    S Dh
awan|   TS Mills|            0|        0|       0|          0|          0|           0|
4|          0|         4|           null|          null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   1|   4|   DA Warner|    S Dh
awan|   TS Mills|            0|        0|       0|          0|          0|           0|
0|          0|         0|           null|          null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   1|   5|   DA Warner|    S Dh
awan|   TS Mills|            0|        2|       0|          0|          0|           0|
0|          2|         2|           null|          null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   1|   6|    S Dhawan|   DA Wa
rner|   TS Mills|            0|        0|       0|          0|          0|           0|
0|          0|         0|           null|          null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   1|   7|    S Dhawan|   DA Wa
rner|   TS Mills|            0|        0|       0|          1|          0|           0|
0|          1|         1|           null|          null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   2|   1|    S Dhawan|   DA Wa
rner|A Choudhary|            0|        0|       0|          0|          0|           0|
1|          0|         1|           null|          null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   2|   2|   DA Warner|    S Dh
awan|A Choudhary|            0|        0|       0|          0|          0|           0|
4|          0|         4|           null|          null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   2|   3|   DA Warner|    S Dh
awan|A Choudhary|            0|        0|       0|          0|          1|           0|
0|          1|         1|           null|          null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   2|   4|   DA Warner|    S Dh
awan|A Choudhary|            0|        0|       0|          0|          0|           0|
6|          0|         6|           null|          null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   2|   5|   DA Warner|    S Dh
awan|A Choudhary|            0|        0|       0|          0|          0|           0|
0|          0|         0|       DA Warner|        caught|Mandeep Singh|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   2|   6|MC Henriques|    S Dh
awan|A Choudhary|            0|        0|       0|          0|          0|           0|
```

```
0|         0|         0|             null|              null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   2|   7|MC Henriques|    S Dh
awan|A Choudhary|            0|        0|       0|          0|          0|           0|
4|         0|         4|             null|              null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   3|   1|    S Dhawan|MC Henri
ques|   TS Mills|            0|        0|       0|          0|          0|           0|
1|         0|         1|             null|              null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   3|   2|MC Henriques|    S Dh
awan|   TS Mills|            0|        0|       0|          0|          0|           0|
0|         0|         0|             null|              null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   3|   3|MC Henriques|    S Dh
awan|   TS Mills|            0|        0|       0|          0|          0|           0|
0|         0|         0|             null|              null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   3|   4|MC Henriques|    S Dh
awan|   TS Mills|            0|        0|       0|          0|          0|           0|
3|         0|         3|             null|              null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   3|   5|    S Dhawan|MC Henri
ques|   TS Mills|            0|        0|       0|          0|          0|           0|
1|         0|         1|             null|              null|         null|
|       1|     1|Sunrisers Hyderabad|Royal Challengers...|   3|   6|MC Henriques|    S Dh
awan|   TS Mills|            0|        0|       0|          0|          0|           0|
1|         0|         1|             null|              null|         null|
+--------+------+-------------------+--------------------+----+----+------------+------
------+-----------+-------------+---------+--------+-----------+-----------+-----------
-+------------+----------+----------+----------------+--------------+-------------+
only showing top 20 rows
```

In [3]:

```
# Check the schema
spark_df.printSchema()
```

```
root
 |-- match_id: long (nullable = true)
 |-- inning: long (nullable = true)
 |-- batting_team: string (nullable = true)
 |-- bowling_team: string (nullable = true)
 |-- over: long (nullable = true)
 |-- ball: long (nullable = true)
 |-- batsman: string (nullable = true)
 |-- non_striker: string (nullable = true)
 |-- bowler: string (nullable = true)
 |-- is_super_over: long (nullable = true)
 |-- wide_runs: long (nullable = true)
 |-- bye_runs: long (nullable = true)
 |-- legbye_runs: long (nullable = true)
 |-- noball_runs: long (nullable = true)
 |-- penalty_runs: long (nullable = true)
 |-- batsman_runs: long (nullable = true)
 |-- extra_runs: long (nullable = true)
 |-- total_runs: long (nullable = true)
 |-- player_dismissed: string (nullable = true)
 |-- dismissal_kind: string (nullable = true)
 |-- fielder: string (nullable = true)
```

In [7]:

```
from pyspark.sql.types import StringType, LongType

# Define the mapping between Spark SQL types and MySQL types
type_mapping = {
    StringType(): "TEXT",
    LongType(): "INTEGER"
}

table = "deliveries"

# Generate the DDL statement for creating a MySQL table based on the DataFrame schema
ddl = "CREATE TABLE "+table+" (\n"
for field in spark_df.schema:
    field_name = field.name
    field_type = type_mapping.get(field.dataType)
```

```
    ddl += f"  {field_name} {field_type},\n"
ddl = ddl[:-2] + "\n)"  # Remove the trailing comma and newline

# Print the DDL statement
print(ddl)
```

```
CREATE TABLE deliveries (
  match_id INTEGER,
  inning INTEGER,
  batting_team TEXT,
  bowling_team TEXT,
  over INTEGER,
  ball INTEGER,
  batsman TEXT,
  non_striker TEXT,
  bowler TEXT,
  is_super_over INTEGER,
  wide_runs INTEGER,
  bye_runs INTEGER,
  legbye_runs INTEGER,
  noball_runs INTEGER,
  penalty_runs INTEGER,
  batsman_runs INTEGER,
  extra_runs INTEGER,
  total_runs INTEGER,
  player_dismissed TEXT,
  dismissal_kind TEXT,
  fielder TEXT
)
```

In [9]:

```
import pymysql

# Set your Aurora MySQL database connection details
host = 'mydbinstance.ctf19flbptnt.us-east-1.rds.amazonaws.com'
port = 3306
user = 'admin'
password = 'MyPassword123'
database = 'ipl'

# Create a connection to the Aurora MySQL database
connection = pymysql.connect(
    host=host,
    port=port,
    user=user,
    password=password
)

# Create a cursor
cursor = connection.cursor()

# Show databases
cursor.execute("SHOW DATABASES")
results=cursor.fetchall()
for result in results:
    print (result)

# Create the database if it doesn't exist
create_db_query = f"CREATE DATABASE IF NOT EXISTS {database}"
cursor.execute(create_db_query)

# Switch to the specified database
use_db_query = f"USE {database}"
cursor.execute(use_db_query)

# Execute the DDL statement
cursor.execute(ddl)
```

```
('information_schema',)
('innodb',)
('mysql',)
```

```
('performance_schema',)
('sys',)
0
```

In [11]:

```
mysql_url = f"jdbc:mysql://{host}:{port}"
mysql_properties = {
    "user": user,
    "password": password,
    "driver": "com.mysql.jdbc.Driver"
}

spark_df.write \
    .format("jdbc") \
    .option("url", mysql_url) \
    .option("dbtable", f"{database}.{table}") \
    .options(**mysql_properties) \
    .mode("append") \
    .save()
```