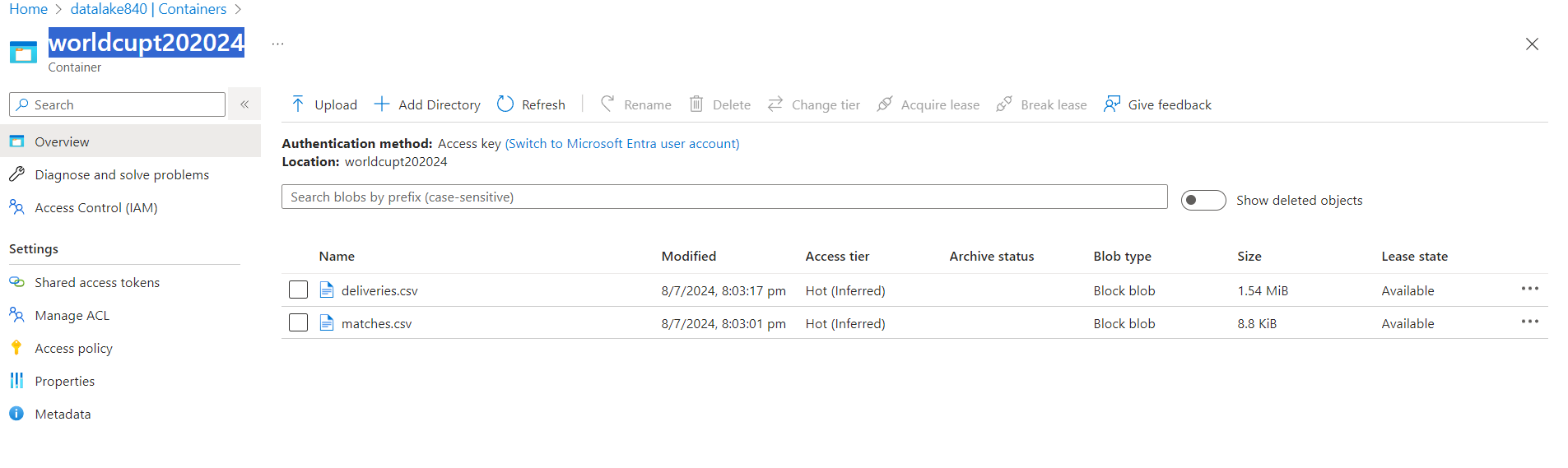
**World-cup-T20 2024**

**Skills: Azure Data Lake, DataBrick, Pyspark, Pysql, DataBrick Dashboard, SQL**

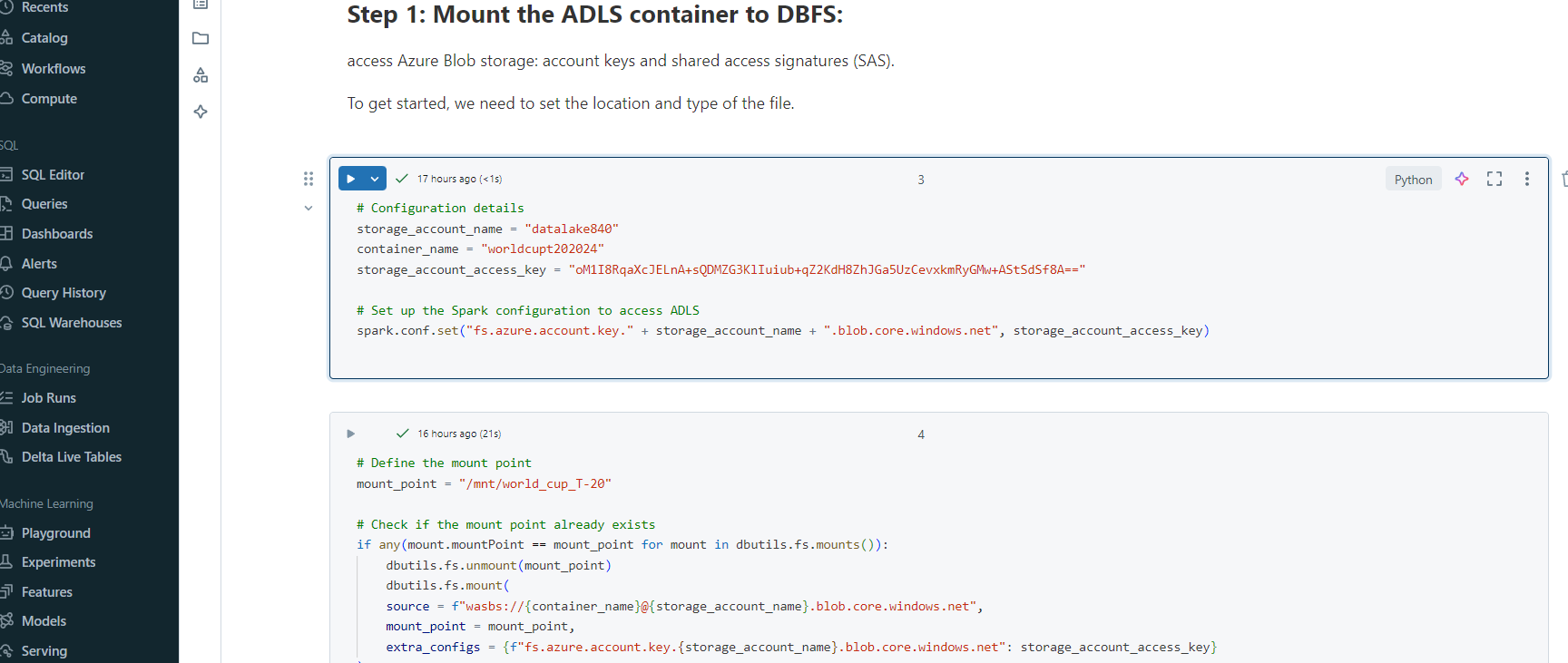
**Objective:**

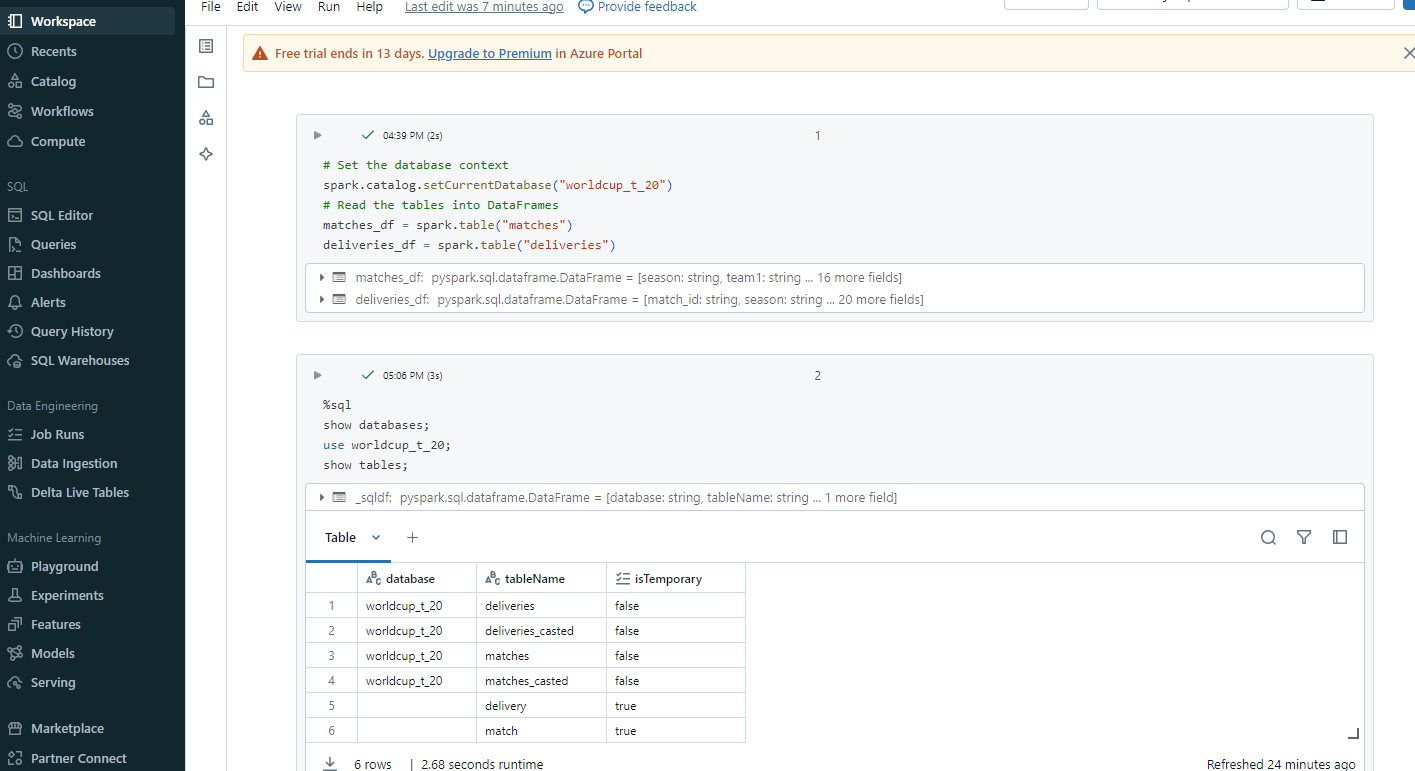
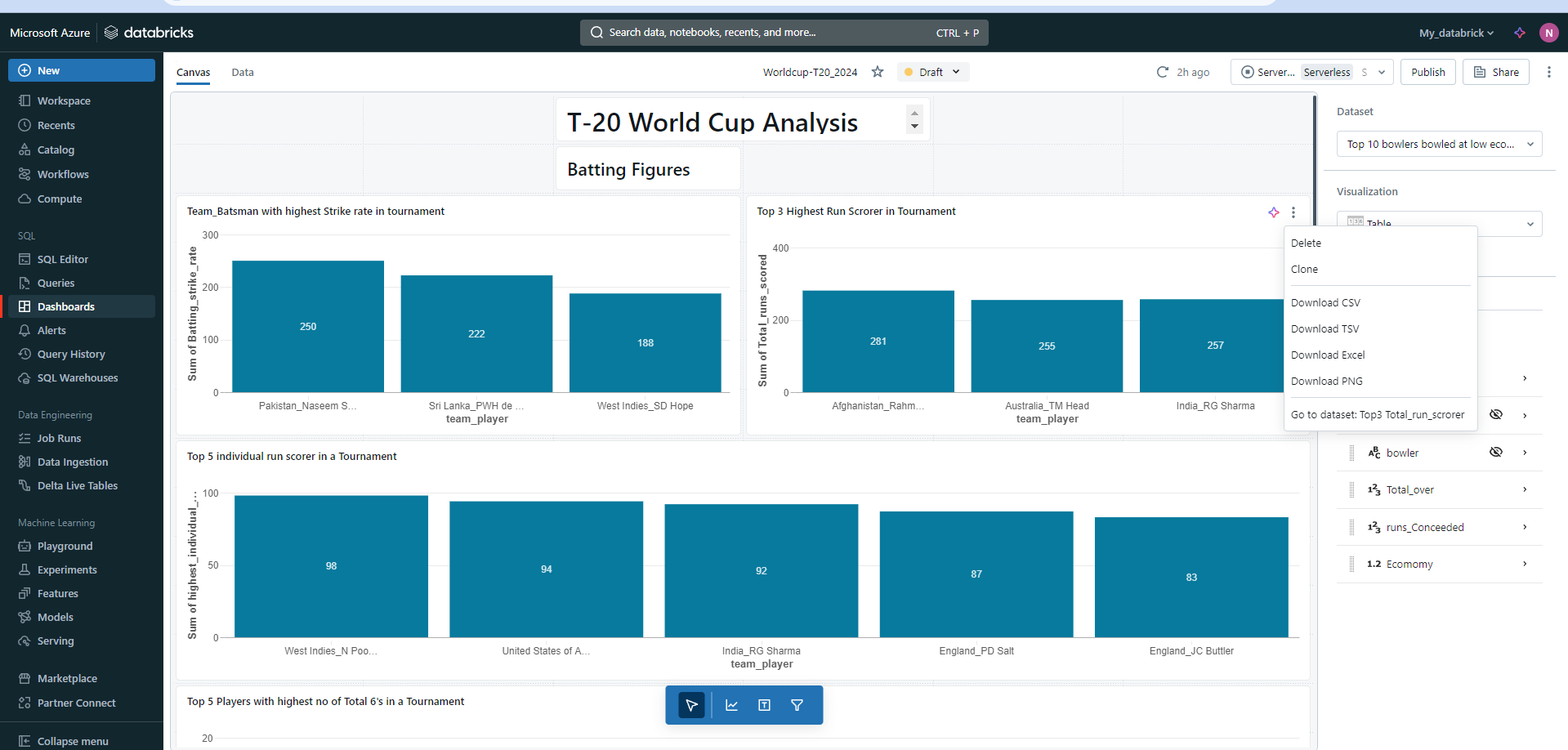
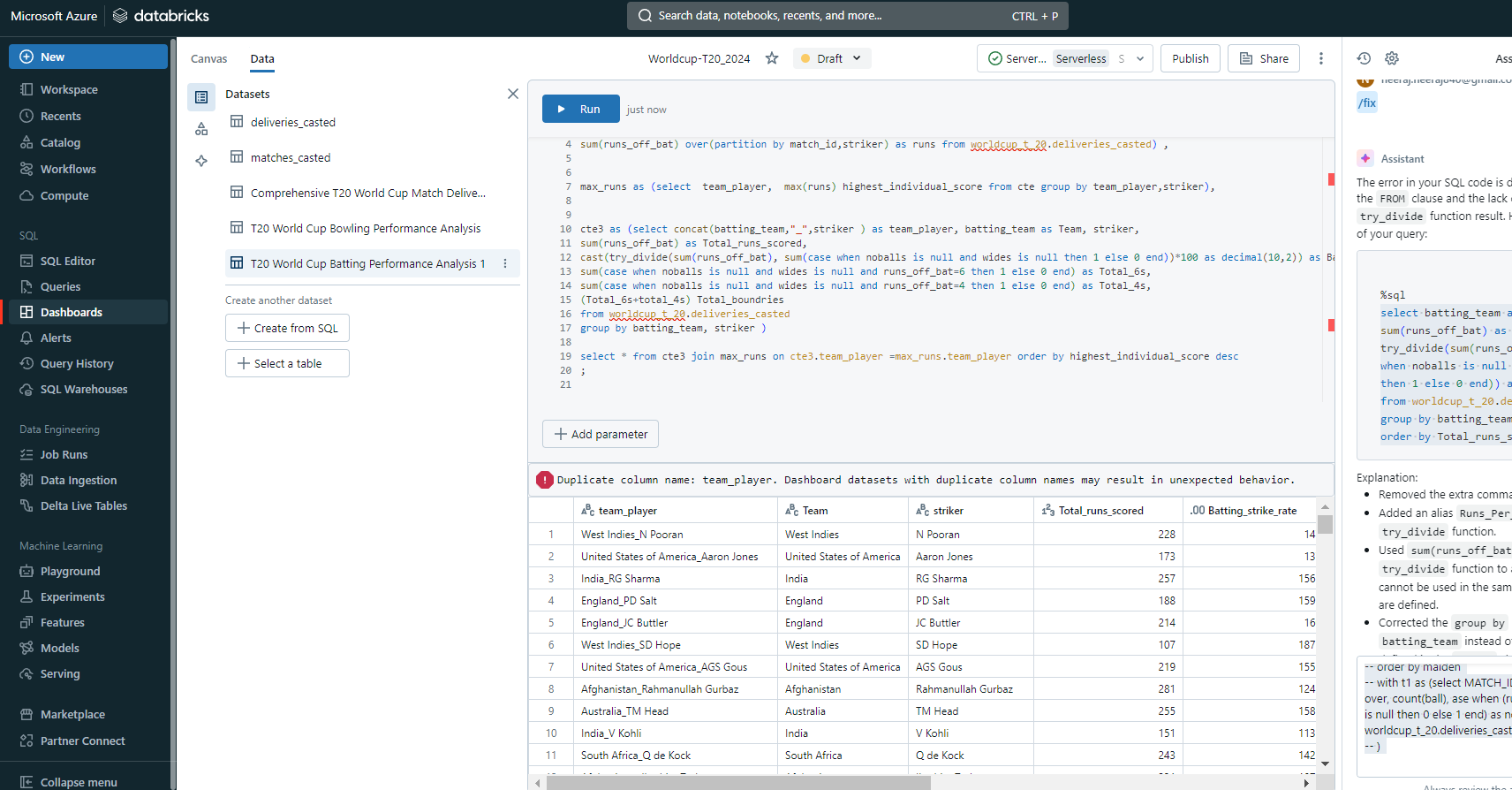
**This project aims to analyze the T20 World Cup 2024 using Databricks to create an interactive and insightful dashboard. This dashboard will leverage the power of PySpark and PySQL to process, analyze, and visualize data. The goal is to provide comprehensive insights into team performances, player statistics, match outcomes, and tournament trends, enabling users to explore and understand the dynamics of the T20 World Cup 2024.**

1. **Data Collection and Ingestion**:
   1. **Create an Azure blob storage account, inside the ‘worldcupt202024’ container, upload the source CSV file.**

****

* 1. **Mount the ADLS container to DBFS using access key:**
  2. **Create the dataframe and upload it on database as table.**
  3. **Codes for this task mentioned in ‘Import from Azure Blob Storage’ Notebook**

****

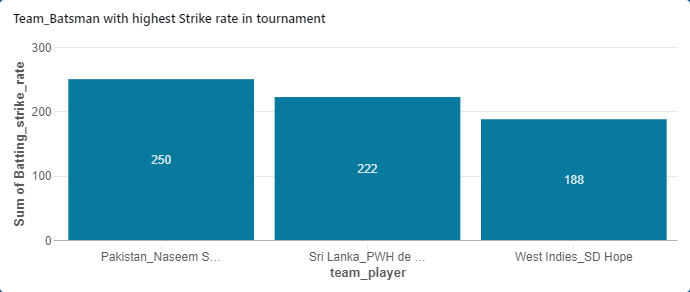
1. **Data Transformation and Data cleaning**
   1. **Created a new notebook to transform and clean data (‘T20Worldcup\_Data Transformation’)**
   2. ****
   3. **Store the clean data in database as a table.**
2. **Dashboard Creation**
   1. **Using data tab on Databrick dashboard select database and tables required to create the dashboard.** ****
   2. **Create own customised SQL queries for dashboard** 
      1. **T20 World Cup Bowling Performance Analysis.sql**
      2. **T20 World Cup Batting Performance Analysis.sql**
      3. **Below mentioned queries and dashboard derived:**
         1. **Team\_Batsman with highest Strike rate in the tournament**

select concat(batting\_team,"\_",striker ) as team\_player, batting\_team as Team, striker,

cast(try\_divide(sum(runs\_off\_bat), sum(case when noballs is null and wides is null then 1 else 0 end))\*100 as decimal(10,2)) as Batting\_strike\_rate

from worldcup\_t\_20.deliveries\_casted

group by batting\_team, striker order by Batting\_strike\_rate desc limit 3;

****

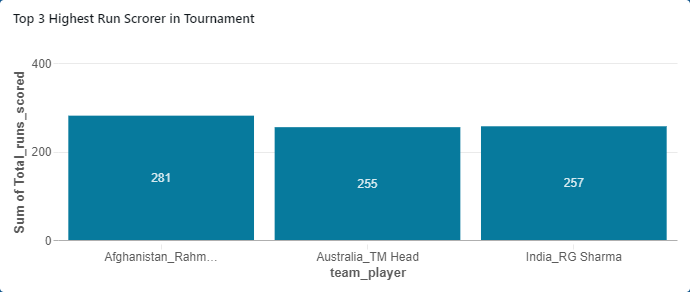
* + - 1. **Top 3 Highest Run Scrorer in Tournament**

select concat(batting\_team,"\_",striker ) as team\_player, batting\_team as Team, striker,

sum(runs\_off\_bat) as Total\_runs\_scored

from worldcup\_t\_20.deliveries\_casted

group by batting\_team, striker  order by Total\_runs\_scored desc limit 3;

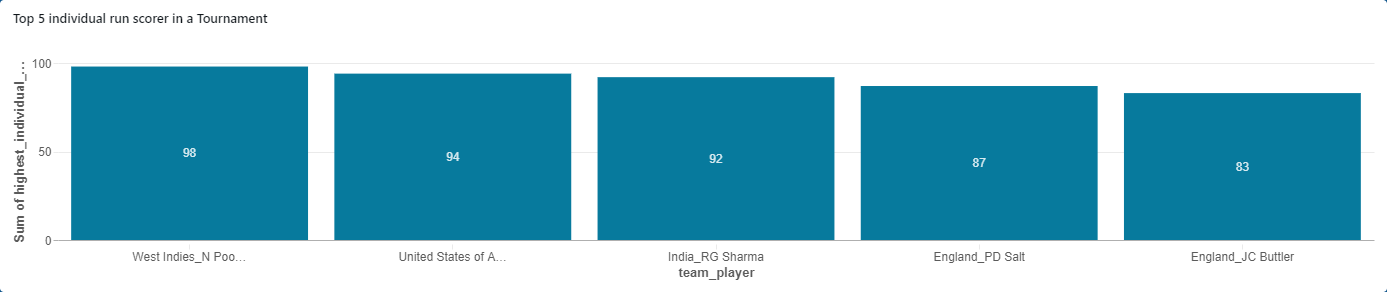
****

* + - 1. **Top 5 individual run scorer in a Tournament**

with cte as(select distinct concat(batting\_team,"\_",striker ) as team\_player,match\_id, batting\_team as Team, striker,

sum(runs\_off\_bat) over(partition by match\_id,striker) as runs from worldcup\_t\_20.deliveries\_casted)

select  team\_player,  max(runs) highest\_individual\_score from cte group by team\_player,striker order by highest\_individual\_score desc limit 5;

****

* + - 1. **Top 5 Players with highest no of Total 6's in a Tournament**

select concat(batting\_team,"\_",striker ) as team\_player, batting\_team as Team, striker,

sum(runs\_off\_bat) as Total\_runs\_scored,

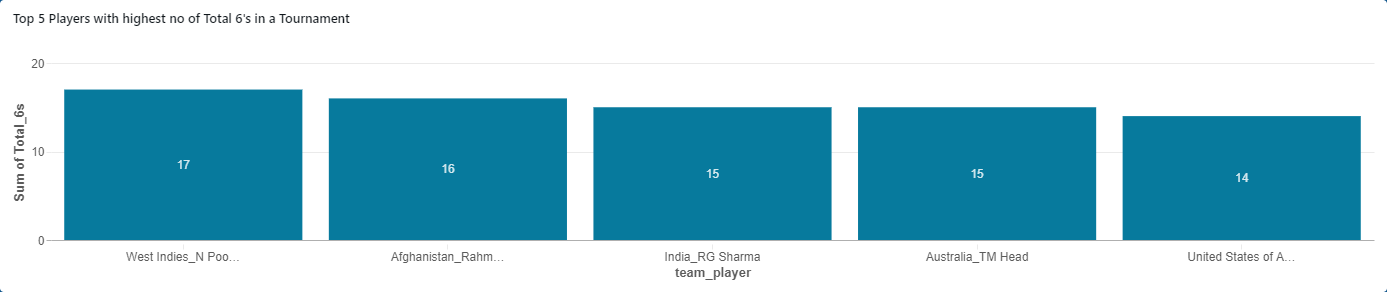
sum(case when noballs is null and wides is null and runs\_off\_bat=6 then 1 else 0 end) as Total\_6s,

sum(case when noballs is null and wides is null and runs\_off\_bat=4 then 1 else 0 end) as Total\_4s

from worldcup\_t\_20.deliveries\_casted

group by batting\_team, striker

order by   Total\_6s desc limit 5;

****

* + - 1. **Top 5 Players with highest no of Total Boundries in a Tournament**

select concat(batting\_team,"\_",striker ) as team\_player, batting\_team as Team, striker,

sum(runs\_off\_bat) as Total\_runs\_scored,

sum(case when noballs is null and wides is null and runs\_off\_bat=6 then 1 else 0 end) as Total\_6s,

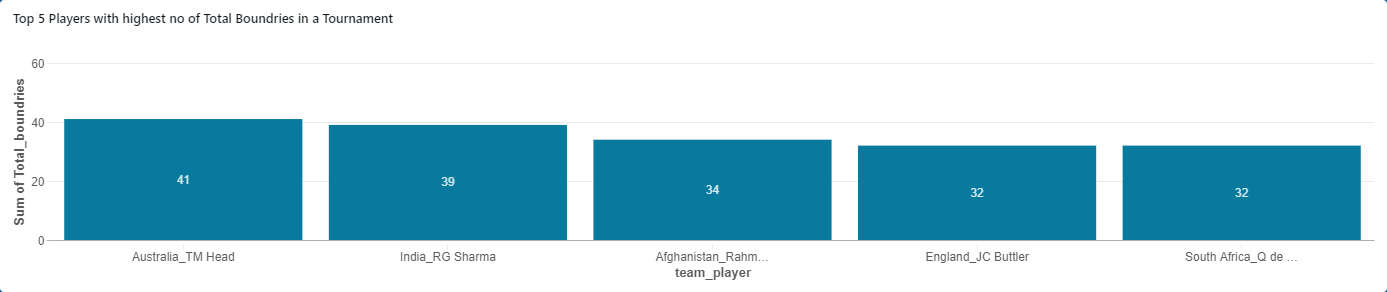
sum(case when noballs is null and wides is null and runs\_off\_bat=4 then 1 else 0 end) as Total\_4s,

(Total\_6s+total\_4s) Total\_boundries

from worldcup\_t\_20.deliveries\_casted

group by batting\_team, striker

order by  Total\_boundries desc limit 5;

****

* + - 1. **Top 5 Wicket takers with their Bowling strike rates**

select concat(bowling\_team,"\_",bowler ) as team\_player,

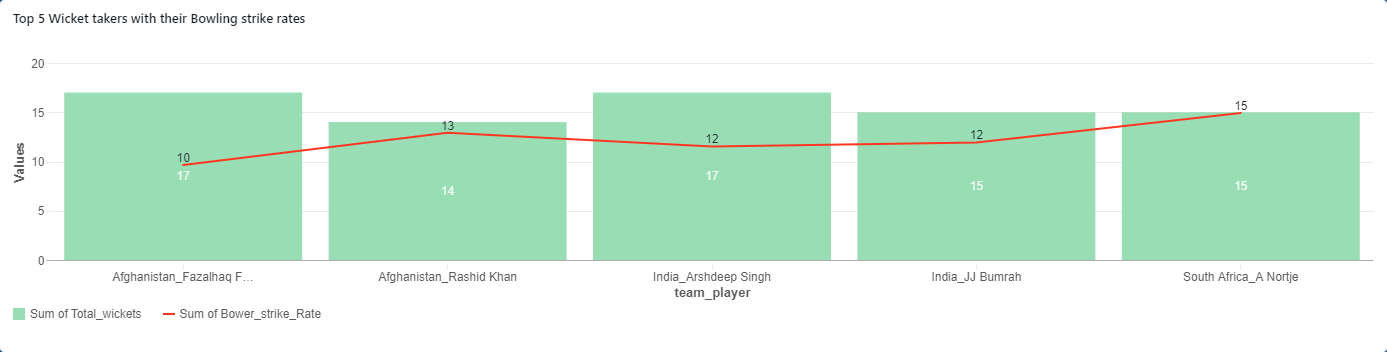
bowling\_team as Team, bowler,

cast(try\_divide(sum(case when noballs != null or wides != null  then 0 else 1 end),sum(case when wicket\_type != 'run out' then 1 else 0 end))as decimal(10,2)) as Bower\_strike\_Rate,

sum(case when wicket\_type != 'run out' then 1 else 0 end) as Total\_wickets

from worldcup\_t\_20.deliveries\_casted

group by bowling\_team,bowler order by Total\_wickets desc limit 5;

****

* + - 1. **Top 10 Bowlers Bowled with lowest economy Rate with alteast 12 over bowles in Tournament**

select concat(bowling\_team,"\_",bowler ) as team\_player,

bowling\_team as Team, bowler,

count(distinct int(ball),match\_id) as Total\_over,

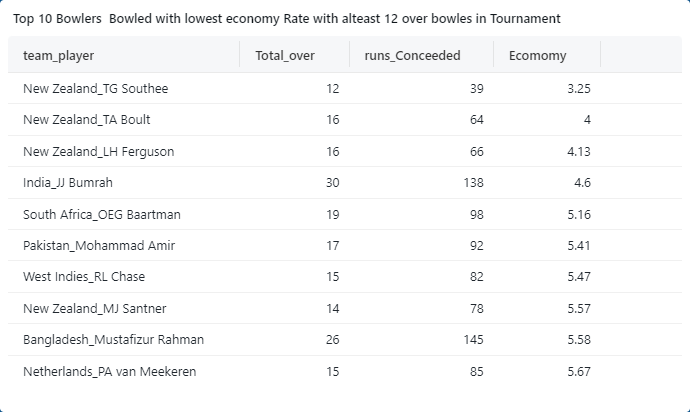
sum(runs\_off\_bat)+sum(extras) runs\_Conceeded,

cast(try\_divide(runs\_Conceeded,Total\_over) as decimal(10,2)) as Ecomomy

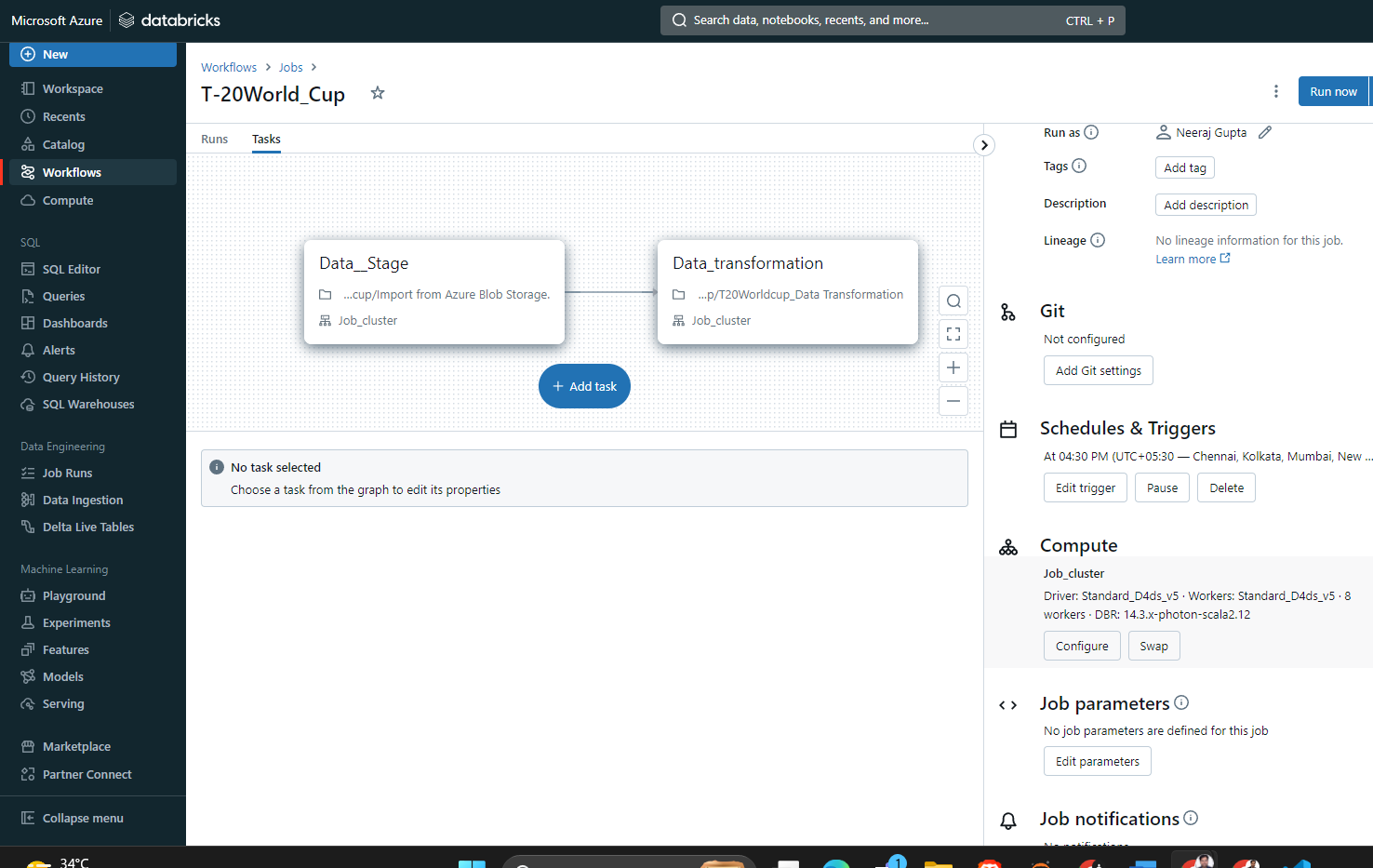
from worldcup\_t\_20.deliveries\_casted

group by bowling\_team,bowler having Ecomomy is not null  and  Total\_over>=12

order by Ecomomy limit 10;

****

* + 1. **Publish and schedule the baseboard to automate the data refresh** .****

1. **Finally schedule the workflow to automate the data flow, in this case it is not requires but if source data is streaming or live than we can automate the workflow using job run option in databrick.** ****