

## About the Dataset

Our dataset consists of 3 tables. These table consists of the stats of the players taking part in the T10 format of the game. The objective of the project is to present the best KPIs of players in each position to help contestants choose best players.

a. Cricket\_bating:

This table gives us important statistics of the players with respect to bating over the last 2 years.

Columns:

player_name	total_runs	hundred
nbr_match	hs	fifty
inns	avg_run	fours
not_outs	ball_faced	sixes
player_id	sr_bat	catches
		stumping

b. Cricket\_bowling:

This table gives us important statistics of the players with respect to bating over the last 2 years.

Columns:

player_id	runs	econ
player_name	wkts	sr
nbr_match	bbi	five_wkt
balls	avg_b	

c. Cricket\_tournamanet:

This table gives us information about the last 17 tournaments (since records were recorded). This tells us about different contestants and how they faired against each other.

Columns:

final_id	venue_choice	bowling_select	venue
date_final	winner	player_top_scorer	
nbr_teams	runner_up	player_top_wkts	
game_format	toss_win	player_series	

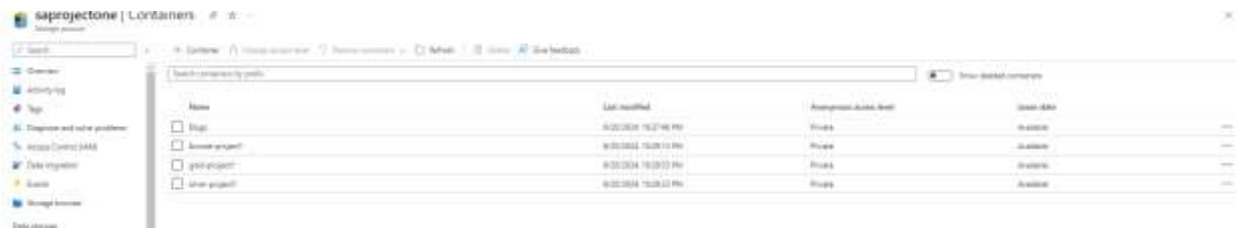
## Setting up Environment

### a. Key Vault:

Safeguarded all the security access tokens and other logins using Key Vaults.

Name	Type	Status	Expiration date
self-identity		✓ Enabled	
Storage-account-key	Storage account key	✓ Enabled	
azuriosdmservice-principal-credential	Service Principal Credential	✓ Enabled	
azuriosdmservice-principal-credential	Service Principal Credential	✓ Enabled	
azuriosdmservice-principal-credential	Service principal credential	✓ Enabled	
api-identity	API Token	✓ Enabled	
api-identity-pool	API Access Pool	✓ Enabled	

### b. ADLS Gen2 with different containers (Medallion Structure):



Name	Last modified	Permissions	Status
staging	8/22/2024 15:27:48 PM	Private	Enabled
development	8/22/2024 15:28:10 PM	Private	Enabled
prod-project	8/22/2024 15:28:10 PM	Private	Enabled
staging-project	8/22/2024 15:28:10 PM	Private	Enabled

### c. Setting up ADF:



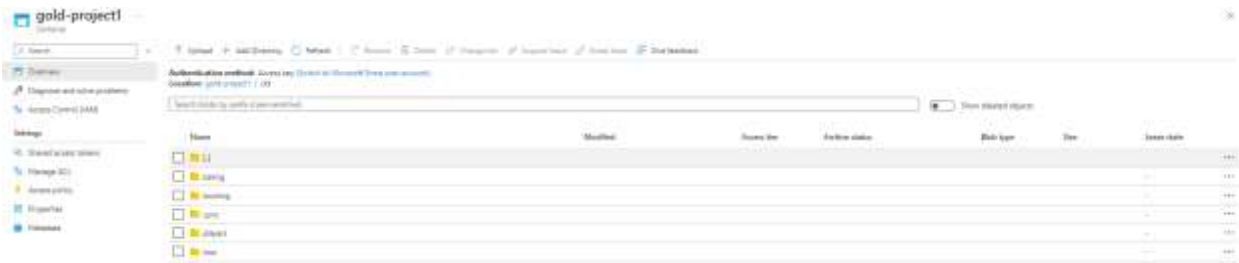
Name	Type	Status	Authentication
AzureDataLakeStore	Azure Data Lake Store	✓	Service Principal
AzureDataLakeStorage	Azure Data Lake Storage (Gen2)	✓	Service Principal
AzureKeyVault	Azure Key Vault	✓	Service Principal
AzureSQLDatabase	Azure SQL Database	✓	Service Principal
SQLServer	SQL Server	✓	Service Principal

### d. Setting up Databricks and Mounting the containers:



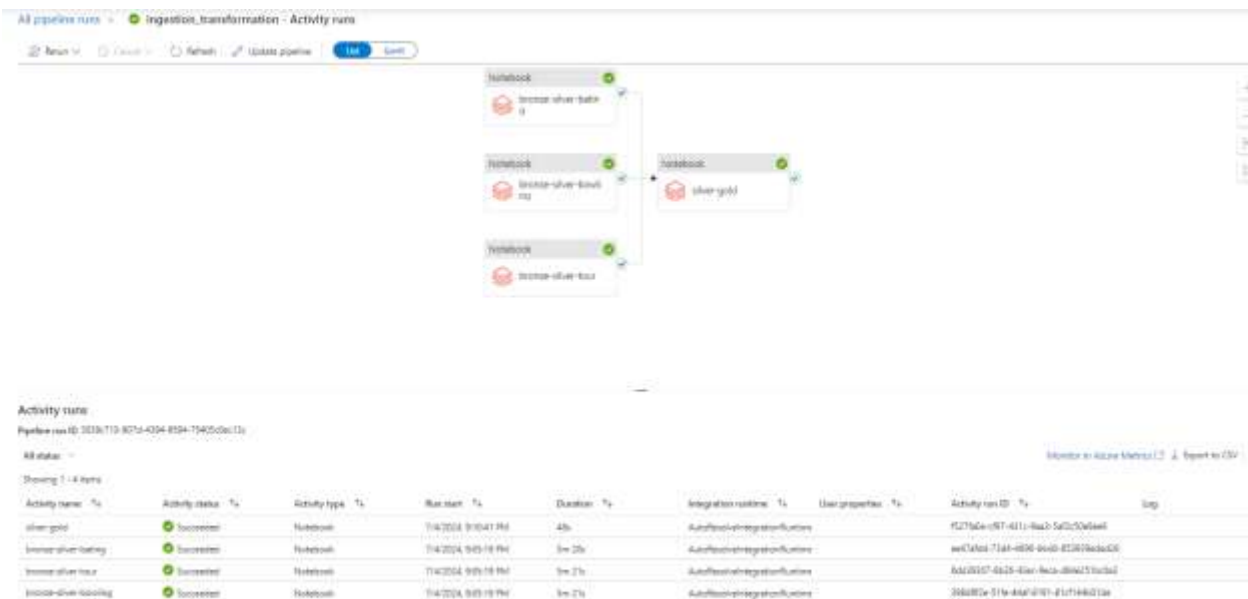
1\_Mounting ADLS  
Using Service Princp





### c. ADF Pipeline Run Silver to Gold:

We implement the first pipeline in Azure Data Factory to Extract the data from ADLS gen2 and Transform the data using Pyspark in Databricks. This takes data from Bronze container and then save the final data to Gold container via Silver container.

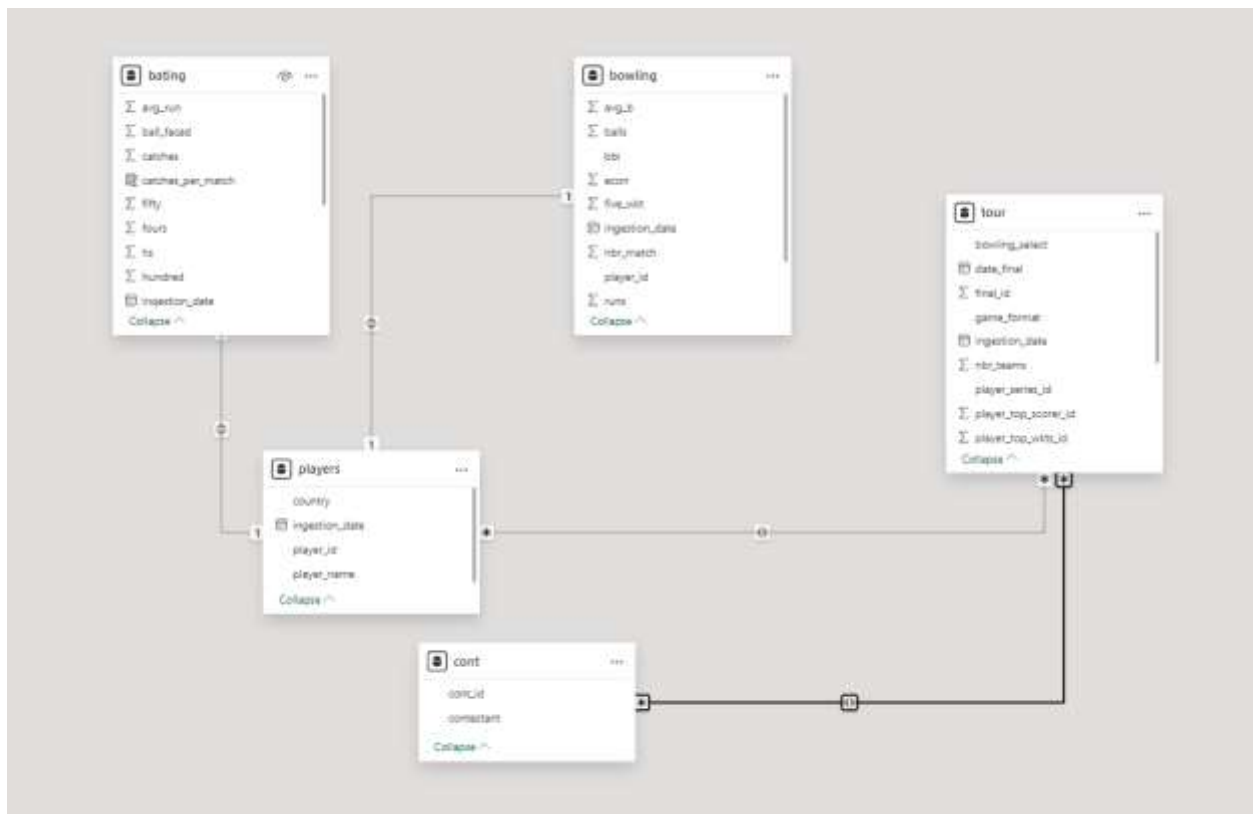


## Data Loading

### a. Loading the transformed data to Data Warehouse (Synapse Serverless Pool):

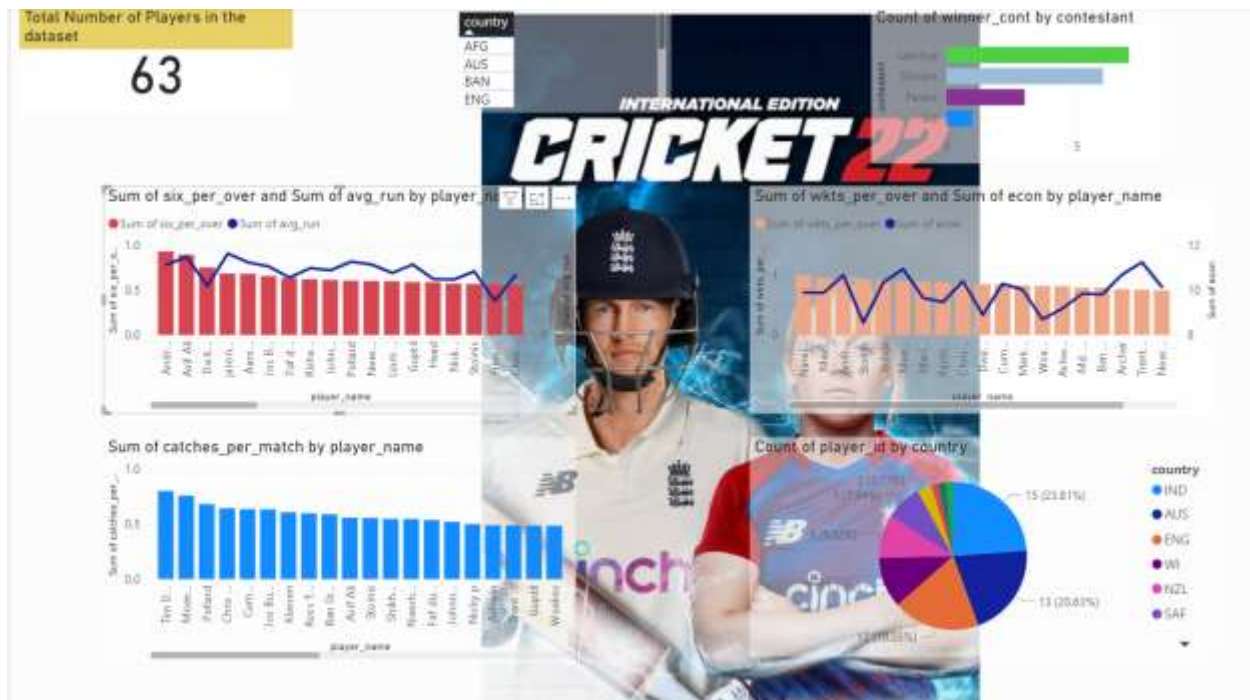
Here we are loading the data using External Table concept (Polybase) to analyse data in Synapse Analytics. We will use the below mentioned Stored Procedure to make views in Synapse so those can be accessed when needed.





## Data Analysis

a. Power BI Dashboard:



## b. SQL Queries using Serverless Pool

-- Top wicket takers

```
SELECT TOP 5 p.player_name, b.wkts, b.sr
FROM bowling b
JOIN players p ON b.player_id = p.player_id
ORDER BY b.wkts DESC;
```

View Table Chart [Export results](#)

Search		
player_name	wkts	sr
Rashid Khan	480	6.8
Archer	319	7.78
Md. Amir	281	7.56
Jadeja	278	6.5
Cummins	239	7.05

00:00:04 Query executed successfully.

```

-- Players and their awards
WITH PlayerAwards AS (
    SELECT
        p.player_id,
        p.player_name,
        SUM(CASE WHEN t.player_series_id = p.player_id THEN 1 ELSE 0 END) AS PlayerOfTheSeriesCount,
        SUM(CASE WHEN t.player_top_scorer_id = p.player_id THEN 1 ELSE 0 END) AS HighestRunScorerCount,
        SUM(CASE WHEN t.player_top_wkts_id = p.player_id THEN 1 ELSE 0 END) AS HighestWicketsCount,
        SUM(CASE WHEN t.player_series_id = p.player_id THEN 1 ELSE 0 END +
            CASE WHEN t.player_top_scorer_id = p.player_id THEN 1 ELSE 0 END +
            CASE WHEN t.player_top_wkts_id = p.player_id THEN 1 ELSE 0 END) AS TotalAwards
    FROM
        players p
    LEFT JOIN
        tour t ON p.player_id IN (t.player_series_id, t.player_top_scorer_id, t.player_top_wkts_id)
    GROUP BY
        p.player_id, p.player_name
)
SELECT
    player_id,
    player_name,
    PlayerOfTheSeriesCount,
    HighestWicketsCount,
    HighestRunScorerCount,
    TotalAwards,
    RANK() OVER (ORDER BY TotalAwards DESC) AS Ranking
FROM
    PlayerAwards
WHERE
    TotalAwards > 0
ORDER BY
    Ranking;

```

player_id	player_name	PlayerOfTheSeriesCount	HighestWicketsCount	HighestRunScorerCount	TotalAwards	Ranking
2	Rashid Khan	4	3	0	7	1
27	Chris Gayle	3	3	1	5	2
55	Ashwin	3	1	1	4	3
51	Jason Roy	0	0	3	3	4
44	Guptill	1	0	2	3	4
23	Aaron Finch	1	0	2	3	4
59	Aqib Ilyas	1	2	0	3	4
34	Rahul Patel	0	2	0	2	6
42	Ben Stokes	1	0	1	2	6
60	Zampa	0	1	0	1	10
6	Mazhar	0	1	0	1	10