KPI

Total of riders

SELECT COUNT(DISTINCT(Rider)) as total\_of\_riders

FROM

[clasificantion list]

A screenshot of a computer

Description automatically generated

Total of teams

* Total of teams in PGEE since 2007 till 2024:

SELECT COUNT(DISTINCT(city)) as Teams

FROM Tabel

WHERE YEAR >= '2007';



Total of audience

SELECT SUM(Sum) as total\_of\_audience

FROM Attendance



Total of defects

SELECT SUM(D) as total\_of\_Deffects

FROM [clasificantion list]

A screenshot of a computer

Description automatically generated

Total of falls

SELECT SUM(U) as total\_of\_falls

FROM [clasificantion list]

A screenshot of a computer

Description automatically generated

Assumptions of the project:

1. Attendance

* Total attendance in each of 3 last years:

SELECT

(SELECT SUM(Sum) FROM Attendance WHERE Year = '2022') AS total\_2022,

(SELECT SUM(Sum) FROM Attendance WHERE Year = '2023') AS total\_2023,

(SELECT SUM(Sum) FROM Attendance WHERE Year = '2024') AS total\_2024;

A screenshot of a computer

Description automatically generated

* Total number of audience over 3 last years:

SELECT SUM(Sum)

FROM Attendance

WHERE Year IN ('2022','2023','2024');

A screenshot of a computer

Description automatically generated

* Top 3 stadiums:

SELECT TOP 3 ShortName, Total

FROM (

SELECT ShortName, SUM(Sum) AS Total

FROM Attendance

WHERE Year IN ('2022', '2023', '2024')

GROUP BY ShortName

) AS SubQuery

ORDER BY Total DESC;

A screenshot of a computer

Description automatically generated

* 1 bootom stadium:

SELECT TOP 1 ShortName, Total

FROM (

SELECT ShortName, SUM(Sum) AS Total

FROM Attendance

WHERE Year IN ('2022', '2023', '2024')

AND ShortName NOT IN ('ZIE', 'OST', 'KRO')

GROUP BY ShortName

) AS SubQuery

ORDER BY Total ASC;

A screenshot of a computer

Description automatically generated

* Trend over the years:

SELECT

COALESCE(a.ShortName, b.ShortName, c.ShortName) AS ShortName,

COALESCE(a.total\_2022, 0) AS total\_2022,

COALESCE(b.total\_2023, 0) AS total\_2023,

COALESCE(c.total\_2024, 0) AS total\_2024

FROM (

SELECT ShortName, SUM(Sum) AS total\_2022

FROM Attendance

WHERE Year = '2022'

AND ShortName NOT IN ('OST', 'ZIE', 'KRO')

GROUP BY ShortName

) AS a

FULL OUTER JOIN (

SELECT ShortName, SUM(Sum) AS total\_2023

FROM Attendance

WHERE Year = '2023'

AND ShortName NOT IN ('OST', 'ZIE', 'KRO')

GROUP BY ShortName

) AS b ON a.ShortName = b.ShortName

FULL OUTER JOIN (

SELECT ShortName, SUM(Sum) AS total\_2024

FROM Attendance

WHERE Year = '2024'

AND ShortName NOT IN ('OST', 'ZIE', 'KRO')

GROUP BY ShortName

) AS c ON a.ShortName = c.ShortName OR b.ShortName = c.ShortName

ORDER BY total\_2022 DESC, total\_2023 DESC, total\_2024 DESC;

A screenshot of a computer

Description automatically generated

1. Medals

* Name of all teams in PGEE:

SELECT DISTINCT(city)

FROM Tabel

WHERE YEAR >= '2007';



* Teams medals

SELECT place, City, Medal, Year

FROM

Tabel

WHERE

Place = 1

AND year >= '2007';

A table of numbers with black text

Description automatically generated

* Sum of Gold/Silver/Brown per team

1.

SELECT City, COUNT(\*) AS Total\_of\_1st\_places

FROM

Tabel

WHERE

Place = 1

AND

Year >= '2007'

GROUP BY City

ORDER BY Total\_of\_1st\_places DESC;

A screenshot of a computer

Description automatically generated

2.

A screenshot of a table

Description automatically generated

3.

A screenshot of a computer

Description automatically generated

* Total of medals

SELECT City, COUNT(\*) as TotalOfMedals

FROM

Tabel

WHERE

year >= '2007'

AND place IN ('1','2','3')

GROUP BY City

ORDER BY TotalOfMedals DESC;

A screenshot of a computer

Description automatically generated

Percentage of medals per team

WITH TotalMedals AS (

SELECT COUNT(Medal) AS Total

FROM tabel

WHERE Medal IS NOT NULL

)

SELECT

CONCAT(COUNT(Medal), ' Medals -') AS Medals,

City,

ROUND((COUNT(Medal) \* 100.0 / (SELECT Total FROM TotalMedals)), 0) AS Percentage

FROM tabel

WHERE Medal IS NOT NULL

GROUP BY City

ORDER BY Medals DESC;

A screenshot of a computer

Description automatically generated

* Relegation after poor season

SELECT City, COUNT(\*) as Relegated

FROM

Tabel

WHERE

Medal = 'Relegated'

AND year >= '2007'

GROUP BY

City

ORDER BY

Relegated DESC

A screenshot of a table

Description automatically generated

1. Players Classification

* TOP 5 most points scored in history

SELECT TOP 5

Rider, sum(Scr) as total

FROM

[clasificantion list]

GROUP BY Rider

ORDER BY total DESC;

A screenshot of a computer

Description automatically generated

* the fewest points scored in history

SELECT TOP 5

Rider, sum(Scr) as total

FROM

[clasificantion list]

GROUP BY Rider

HAVING

SUM(Scr) <> 0

ORDER BY total ASC;

A screenshot of a computer

Description automatically generated

* TOP 5 most bonus points scored in history

SELECT TOP 5

Rider, sum(Bon) as total

FROM

[clasificantion list]

GROUP BY Rider

ORDER BY total DESC;

A screenshot of a computer

Description automatically generated

* The highest average in history

SELECT TOP 5

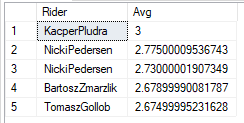
Rider, Avg

FROM

[clasificantion list]

ORDER BY

Avg DESC;



* 5 players with the most first places

SELECT TOP 5

Rider, SUM(I) as Total

FROM

[clasificantion list]

GROUP BY Rider

ORDER BY Total DESC;

A screenshot of a computer

Description automatically generated

* 5 players with the most second places

SELECT TOP 5

Rider, SUM(II) as Total

FROM

[clasificantion list]

GROUP BY Rider

ORDER BY Total DESC;

A screenshot of a computer

Description automatically generated

* 5 players with the most third places

SELECT TOP 5

Rider, SUM(III) as Total

FROM

[clasificantion list]

GROUP BY Rider

ORDER BY Total DESC;

A screenshot of a computer

Description automatically generated

* 5 players with the most fourth places

SELECT TOP 5

Rider, SUM(IV) as Total

FROM

[clasificantion list]

GROUP BY Rider

ORDER BY Total DESC;

A screenshot of a computer

Description automatically generated

* TOP 5 players with the most matches

SELECT TOP 5

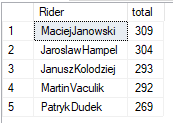
Rider,SUM(Me) as total

FROM

[clasificantion list]

GROUP BY Rider

ORDER BY total DESC;



* TOP 5 players with the most defects

SELECT TOP 5

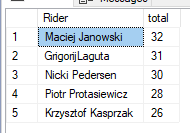
Rider,SUM(D) as total

FROM

[clasificantion list]

GROUP BY Rider

ORDER BY total DESC;



* TOP 5 players with the most exclusions

SELECT TOP 5

Rider,SUM(W) as total

FROM

[clasificantion list]

GROUP BY Rider

ORDER BY total DESC;

A screenshot of a computer

Description automatically generated