BAHRIA UNIVERSITY KARACHI CAMPUS

DEPARTMENT OF ARTIFICIAL INTELLIGENCE

Bahria University
Discovering Knowledge

Programming for Artificial Intelligence Lab

AIL-202

Lab Project: Formula1 Dashboard

Name	Enrolment
Syed Aazain Hasan	02-136232-030
Abdul Mateen	02-136232-048
Talha Shahid	02-136232-028

SUBMITTED TO

Ms.Salas Akbar

ACKNOWLEDGEMENT

We have taken efforts in this report. However, it would not have been possible without the kind support and help of our respected Teacher. We would like to extend our sincere thanks to her.

We would like to express our gratitude to Ms. Salas Akbar for her valuable guidance and constant encouragement throughout the completion of this project. Her insights and feedback played a crucial role in shaping this report.

ABSTRACT

This report presents a comprehensive analysis of Formula 1 data using interactive dashboards created in Power BI. The project explores the performance of drivers and constructors across multiple dimensions, including race wins, points, participation, and efficiency. The Drivers Dashboard highlights individual driver achievements, such as the number of wins, participation rates, pole positions, and their impact on overall performance. It also examines key factors affecting race outcomes, such as starting grid position and non-finish reasons.

The Constructors Dashboard delves into team-level analytics, emphasizing the dominance of teams like Ferrari, McLaren, and Mercedes in terms of wins, points, and race participation. Metrics like pit stop efficiency, total points by season, and win-to-point ratios provide insights into constructors' performance trends over time.

By leveraging DAX measures and relational data modeling, the dashboards provide actionable insights into the evolution of Formula 1, the factors influencing success, and the interplay between drivers and constructors. The report demonstrates how advanced data visualization techniques can enhance decision-making in competitive sports.

TABLE OF CONTENTS

Chapter 1: INTRODUCTION

1.1 History

F1 (Formula 1), is the pinnacle of motorsport. It is a world-renowned racing championship featuring the fastest cars and most skilled drivers. Established in 1950, the sport has evolved significantly over the decades, with technological advancements and fierce competition defining its legacy.

1.2 Overview

This project focuses on analyzing and visualizing the performance of Formula 1 constructors and drivers over multiple seasons using Power BI. By leveraging historical data, the analysis aims to uncover patterns, trends, and key insights into the sport.

1.3 Objectives

- To analyze constructors' and drivers' performance across seasons.
- To identify key metrics such as wins, points, and championships.
- To provide visual insights into various aspects of Formula 1 data.
- To create interactive dashboards for constructors and drivers.

1.4 Problem Statement

Analyzing performance in Formula 1 involves understanding complex relationships between drivers, constructors, race conditions, and historical trends. With vast amounts of data generated across decades, it is challenging to identify key factors influencing success, such as driver consistency, constructor dominance, race reliability, and team efficiency. Without effective data visualization and analysis, deriving actionable insights from this data to celebrate achievements, improve performance strategies, and predict future trends becomes difficult. This report addresses this challenge by leveraging interactive dashboards to uncover meaningful patterns and insights within Formula 1 data, enabling informed decision-making for enthusiasts, analysts, and teams.

Chapter 2: Data Sources and Methodology

2.1 Data Sources

The dataset used in this project was sourced from Kaggle, containing historical Formula 1 data spanning several decades. The dataset consists of multiple tables containing relational data about F1 races, drivers, constructors, circuits, and results:

- **Drivers**: Contains driver forename, driver surname, nationality, dob, driver id, driverref.
- **Results:** Central fact table containing race outcomes with key columns containing driver_id, constructor_id, points and position.
- **Constructors Table:** Contains information about constructors such as constructor_id, nationality, constructor name, etc.
- **Circuits:** Contains details about F1 circuits such as location, country, circuit id and name.
- **Seasons:** Contains all the F1 seasons and years.
- **Constructor Standings:** Contains columns like constructor id, points, and position.
- **Driver Standings:** Contains driver id, points, position and race id.
- **Pit Stops:** Contains driver id, duration, lap, etc.
- **Races:** Contains circuit id, date, name, race id and year.
- **Status:** Contains status of race scenario of a driver and their respective status id.

2.2 Methodology

2.2.1 Data Cleaning & Transformation

Data Loading:

```
import pandas as pd

import numpy as np

drivers = pd.read_csv("drivers.csv", na_values=r'\N')

constructors = pd.read_csv("constructors.csv", na_values=r'\N')

races = pd.read_csv("races.csv", na_values=r'\N')

results = pd.read_csv("results.csv", na_values=r'\N')

seasons = pd.read_csv("seasons.csv", na_values=r'\N')
```

```
circuits = pd.read_csv("circuits.csv", na_values=r'\N')
status = pd.read_csv("status.csv", na_values=r'\N')
driver_standings = pd.read_csv("driver_standings.csv", na_values=r'\N')
constructor_standings = pd.read_csv("constructor_standings.csv", na_values=r'\N')
pit_stops = pd.read_csv("pit_stops.csv", na_values=r'\N')
```

Dropping Unnecessary Columns & Filling Null Values:

```
drivers = drivers.rename(columns=lambda x: x.strip().lower())
drivers.fillna("Unknown", inplace=True)
drivers.drop(columns=['number', 'code', 'url'], errors='ignore', inplace=True)

constructors = constructors.rename(columns=lambda x: x.strip().lower())
constructors.fillna("Unknown", inplace=True)
constructors.drop(columns=['url'], errors='ignore', inplace=True)

races = races.rename(columns=lambda x: x.strip().lower())
races["date"] = pd.to_datetime(races["date"], errors="coerce")
races.fillna("Unknown", inplace=True)
races.drop(columns=['time', 'url', 'fp1_date', 'fp1_time', 'fp2_date', 'fp2_time',
                    'fp3_date', 'fp3_time', 'quali_date', 'quali_time',
                    'sprint_date', 'sprint_time'], errors='ignore', inplace=True)

results = results.rename(columns=lambda x: x.strip().lower())
results.drop(columns=['number', 'fastestlap', 'fastestlapspeed', 'milliseconds',
                      'time', 'positiontext'], errors='ignore', inplace=True)
```

```
results.fillna(0, inplace=True)

seasons = seasons.rename(columns=lambda x: x.strip().lower())
seasons.fillna("Unknown", inplace=True)
seasons.drop(columns=['url'], errors='ignore', inplace=True)

circuits = circuits.rename(columns=lambda x: x.strip().lower())
circuits.fillna("Unknown", inplace=True)
circuits.drop(columns=['url', 'lat', 'lng', 'alt'], errors='ignore', inplace=True)

status = status.rename(columns=lambda x: x.strip().lower())
status.fillna("Unknown", inplace=True)

driver_standings = driver_standings.rename(columns=lambda x: x.strip().lower())
driver_standings.drop(columns=['positiontext'], errors='ignore', inplace=True)

constructor_standings = constructor_standings.rename(columns=lambda x: x.strip().lower())
constructor_standings.drop(columns=['positiontext'], errors='ignore', inplace=True)
```

Data Transformation:

```
pit_stops['duration'] = pit_stops['milliseconds'] / 1000
pit_stops.drop(columns=["time"], errors='ignore', inplace=True)
```

Merging & Exporting Data:

```
drivers.to_csv("cleaned_drivers.csv", index=False)
```

```
constructors.to_csv("cleaned_constructors.csv", index=False)
races.to_csv("cleaned_races.csv", index=False)
results.to_csv("cleaned_results.csv", index=False)
seasons.to_csv("cleaned_seasons.csv", index=False)
circuits.to_csv("cleaned_circuits.csv", index=False)
status.to_csv("cleaned_status.csv", index=False)
driver_standings.to_csv("cleaned_driver_standings.csv", index=False)
constructor_standings.to_csv("cleaned_constructor_standings.csv", index=False)

output_file = "merged_f1_data.xlsx"

with pd.ExcelWriter(output_file, engine='xlsxwriter') as writer:
    drivers.to_excel(writer, sheet_name='Drivers', index=False)
    constructors.to_excel(writer, sheet_name='Constructors', index=False)
    races.to_excel(writer, sheet_name='Races', index=False)
    results.to_excel(writer, sheet_name='Results', index=False)
    seasons.to_excel(writer, sheet_name='Seasons', index=False)
    circuits.to_excel(writer, sheet_name='Circuits', index=False)
    status.to_excel(writer, sheet_name='Status', index=False)
    pit_stops.to_excel(writer, sheet_name='Pit Stops', index=False)
    driver_standings.to_excel(writer, sheet_name='Driver Standings', index=False)
    constructor_standings.to_excel(writer, sheet_name='Constructor Standings', index=False)

print("Data cleaning, transformation, and export completed!")
```

2.2.2 Relationship Mapping

The relationships between tables ensure seamless data integration:

1. Results ↔ Drivers
   Link: driver_id
   Type: Many-to-One (One driver can participate in multiple races).

2. Results ↔ Constructors
   Link: constructor_id
   Type: Many-to-One (One constructor can field multiple cars).

3. Results ↔ Races
   Link: race_id
   Type: Many-to-One (One race can have multiple results).

4. Races ↔ Seasons
   Link: year
   Type: Many-to-One (One season contains multiple races).

5. Results ↔ Status
   Link: status_id
   Type: Many-to-One (One race result can have many status).

6. Constructors ↔ Constructor_Standings
   Link: constructor_id
   Type: One-to-Many (One constructor can have many standings).

7. Races ↔ Circuits
   Link: circuit_id
   Type: Many-to-One (One circuit can host multiple races).

8. Drivers ↔ Driver_Standings
   Link: driver_id
   Type: One-to-Many (One driver can have many standings).

9. Drivers ↔ Pit Stops
   Link: driver_id
   Type: One-to-Many (One driver can have multiple pit stops).

10. Drivers ↔ Results
    Link: driver_id

Type: One-to-Many (One driver can have many results).

11. Constructor ↔ Results
    Link: constructor_id
    Type: One-to-Many (One driver can have many results).

12. Races ↔ Pit Stops
    Link: race_id
    Type: One-to-Many (One race can have many pitstops).

2.2.3 DAX Measures

- **Accident %:** Calculates the percentage of races where accidents occurred.

```
Accident (%) =
DIVIDE(
    COUNTROWS(FILTER(Results, Results[statusid] = 3)),
    COUNTROWS(Results)
)
```

- **Total Points**: Calculates the total points.

```
Total Points = SUM(results[points])
```

- **Finished Race (%):** Calculates the percentage of races that drivers finished.

```
Finished Race (%) =
DIVIDE(
    COUNTROWS(FILTER(Results, Results[statusid] = 1)),
    COUNTROWS(Results)
)
```

Most Participating Racer Name: Identifies the name of the racer who participated in the most races.

```
Most Participating Racer Name =
VAR ParticipationTable =
    SUMMARIZE(
        Results,
        Results[driverId],
        "Participations", COUNT(Results[resultId])
    )
VAR TopParticipant =
    TOPN(1, ParticipationTable, [Participations], DESC)
RETURN
    MAXX(
        TopParticipant,
        LOOKUPVALUE(Drivers[drivername], Drivers[driverid], Results[driverid])
    )
```

- **Most Pole Position Driver Name**: Finds the driver with the most pole positions.

```
Most Pole Position Driver Name =
VAR PolePositionTable =
    SUMMARIZE(
        Results,
        Drivers[drivername],
        "PoleCount", CALCULATE(COUNT(Results[resultid]), Results[grid] = 1)
    )
VAR TopDriver =
    TOPN(1, PolePositionTable, [PoleCount], DESC)
RETURN
    MAXX(TopDriver, Drivers[drivername])
```

- **Top 10 Finishes**: Calculates the number of times drivers finished within the top 10 positions

```
Top 10 Finishes =
CALCULATE(
    COUNTROWS(Results),
    Results[positionOrder] <= 10
)
```

- **Most Pole Positions Count**: Calculates the count of pole positions for the driver with the most.

```
Most Pole Positions Count =
VAR PolePositionTable =
    SUMMARIZE(
        Results,
        Drivers[drivername],
        "PoleCount", CALCULATE(COUNT(Results[resultId]), Results[grid] = 1)
    )
VAR TopDriver =
    TOPN(1, PolePositionTable, [PoleCount], DESC)
RETURN
    MAXX(TopDriver, [PoleCount])
```

- **Most Winning Racer Name**: Identifies the name of the racer with the most race wins.

```
Most Winning Racer Name =
VAR WinningTable =
    SUMMARIZE(
        FILTER(Results, Results[positionorder] = 1),
        Results[driverId],
        "Wins", COUNT(Results[resultid])
    )
VAR TopWinner =
    TOPN(1, WinningTable, [Wins], DESC)
RETURN
    MAXX(
        TopWinner,
        LOOKUPVALUE(Drivers[drivername], Drivers[driverid], Results[driverid])
    )
```

- **Most Winning Racer Wins**: Counts the total wins for the racer with the most race victories.

```
Most Winning Racer Wins =
VAR WinningTable =
    SUMMARIZE(
        FILTER(Results, Results[positionorder] = 1),
        Results[driverid],
        "Wins", COUNT(Results[resultid])
    )
VAR TopWinner =
    TOPN(1, WinningTable, [Wins], DESC)
RETURN
    MAXX(TopWinner, [Wins])
```

- **Non-Finish Count**: Counts the number of races that drivers did not finish.

```
Non-Finish Count =
COUNTX(
    FILTER(Results, NOT Results[statusid] = 1),
    Results[resultid]
)
```

- **Win Rate (%)**: Calculates the win rate for drivers or constructors in percentage, filtered for a threshold of 5%.

```
Win Rate (%) =
VAR TotalWins =
    COUNTROWS(FILTER(Results, Results[positionOrder] = 1))
VAR TotalRaces =
    COUNTROWS(FILTER(Results, NOT(ISBLANK(Results[raceId]))))
VAR WinRate =
    DIVIDE(TotalWins, TotalRaces, 0) * 100
RETURN
    IF(WinRate >= 5, WinRate, BLANK())
```

- **Total Constructors**: Counts the number of constructors in the dataset.

```
Total Constructors =
CALCULATE(
    DISTINCTCOUNT(Results[constructorid]),
    VALUES(Races[year])
)
```

- **Nationality Count**: Counts the total number of nationalities represented by drivers.

```
Nationality Count = COUNT(Drivers[driverid])
```

- **Highest Win Rate %**: Finds the highest win rate percentage among drivers with at least 10 participations.

```
Highest Win Rate % =
VAR ParticipationTable =
    SUMMARIZE(
        Results,
        Results[driverid],
        "Wins", COUNTX(FILTER(Results, Results[positionorder] = 1), Results
        [resultid]),
        "Participations", COUNT(Results
        [resultid])
    )
VAR AddWinRate =
    ADDCOLUMNS(
        ParticipationTable,
        "WinRate", DIVIDE([Wins], [Participations], 0)
    )
VAR FilteredWinRate =
    FILTER(
        AddWinRate,
        [Participations] >=
        10                                                    --
    )
VAR TopWinRate =
    TOPN(1, FilteredWinRate, [WinRate], DESC)
RETURN
    MAXX(TopWinRate, [WinRate])
```

- **Constructor Total Points By Season:** Calculates the points of constructors by season.

```
Constructor Total Points by Season =
CALCULATE(
    SUM('results'[points]),
    ALLEXCEPT(
        'results',
        'constructors'[constructorId],
        'races'[year]
    )
)
```

- **Highest Win Rate Racer Name**: Identifies the name of the driver with the highest win rate

```
Highest Win Rate Racer Name =
VAR ParticipationTable =
    SUMMARIZE(
        Results,
        Results[driverid],
        "Wins", COUNTX(FILTER(Results, Results[positionorder] = 1), Results[resultid]),
        "Participations", COUNT(Results[resultid])
    )
VAR AddWinRate =
    ADDCOLUMNS(
        ParticipationTable,
        "WinRate", DIVIDE([Wins], [Participations], 0) * 100
    )
VAR FilteredWinRate =
    FILTER(
        AddWinRate,
        [Participations] >= 10
    )
VAR TopWinRate =
    TOPN(1, FilteredWinRate, [WinRate], DESC)
RETURN
    MAXX(
        TopWinRate,
        LOOKUPVALUE(
            Drivers[drivername],
            Drivers[driverid], [driverid]
        )
    )
```

- **Average Pit Stop Time by Constructor**: Calculates the average pit stop time for each constructor.

```
Average Pit Stop Time by Constructor =
CALCULATE(
    AVERAGE(Pit_Stops[duration]),
    RELATEDTABLE(Results),
    VALUES(Constructors[constructorId])
)
```

- **Constructor With Most Points:** Returns constructor with most points.

```
Constructor with Most Points =
VAR ConstructorPoints =
    SUMMARIZE(
        'results',
        'constructors'[constructorId],
        'Constructors'[name],
        "TotalPoints", SUM('results'[points])
    )

VAR MaxPoints =
    MAXX(ConstructorPoints, [TotalPoints])

VAR TopConstructor =
    FILTER(ConstructorPoints, [TotalPoints] = MaxPoints)

RETURN
    MAXX(TopConstructor, 'constructors'[name])
```

- **Most Participating Constructor Count**: Returns the constructor with most participations

```
Most Participating Constructor Count =
VAR ParticipationTable =
    SUMMARIZE(
        Results,
        Results[constructorid],
        "UniqueRaces", DISTINCTCOUNT(Results[raceId])
    )
VAR TopParticipant =
    TOPN(1, ParticipationTable, [UniqueRaces], DESC)
RETURN
    MAXX(TopParticipant, [UniqueRaces])
```

- **Most Participating Constructor Name:**

```
Most Participating Constructor Name =
VAR ParticipationTable =
    SUMMARIZE(
        Results,
        Results[constructorid],
        "Participations", COUNT(Results[resultId])
    )
VAR TopParticipant =
    TOPN(1, ParticipationTable, [Participations], DESC)
RETURN
    MAXX(
        TopParticipant,
        LOOKUPVALUE(Constructors[name], Constructors[constructorid], Results[constructorid])
    )
```

- **Constructor With Most Wins**: Returns constructor with most wins.

```
Constructor with Most Wins =
VAR ConstructorRaceWins =
    SUMMARIZE(
        'results',
        'constructors'[constructorId],
        'constructors'[name],
        "Wins", CALCULATE(COUNTROWS('results'), 'results'[positionOrder] = 1)
    )

VAR MaxWins =
    MAXX(ConstructorRaceWins, [Wins])

VAR TopConstructor =
    FILTER(ConstructorRaceWins, [Wins] = MaxWins)

RETURN
    MAXX(TopConstructor, 'constructors'[name])
```

Chapter 3: Dashboard Design and Development

3.1 Drivers Dashboard

Figure 3.1 1

3.1.1 Multi-Row Cards:

Figure 3.1 2

Most Winning Racer: Highlights the driver with the most race wins (e.g., Lewis Hamilton).

Most Participating Racer: Shows the driver who has participated in the most races (e.g., Fernando Alonso).

Most Pole Positions: Displays the driver with the most pole positions, indicating their dominance in qualifying rounds (e.g., Lewis Hamilton).

Highest Win Rate (%): Reflects the driver with the best win rate, calculated as a percentage of races won relative to total races participated in (e.g., Juan Fangio with 41.38%).

3.1.2 KPI Cards

Figure 3.1 3

Total Drivers**:** Displays the total number of drivers who participated in Formula 1 races.

Total Races**:** Shows the total number of Formula 1 races held to date.

3.1.3 **Slicers**

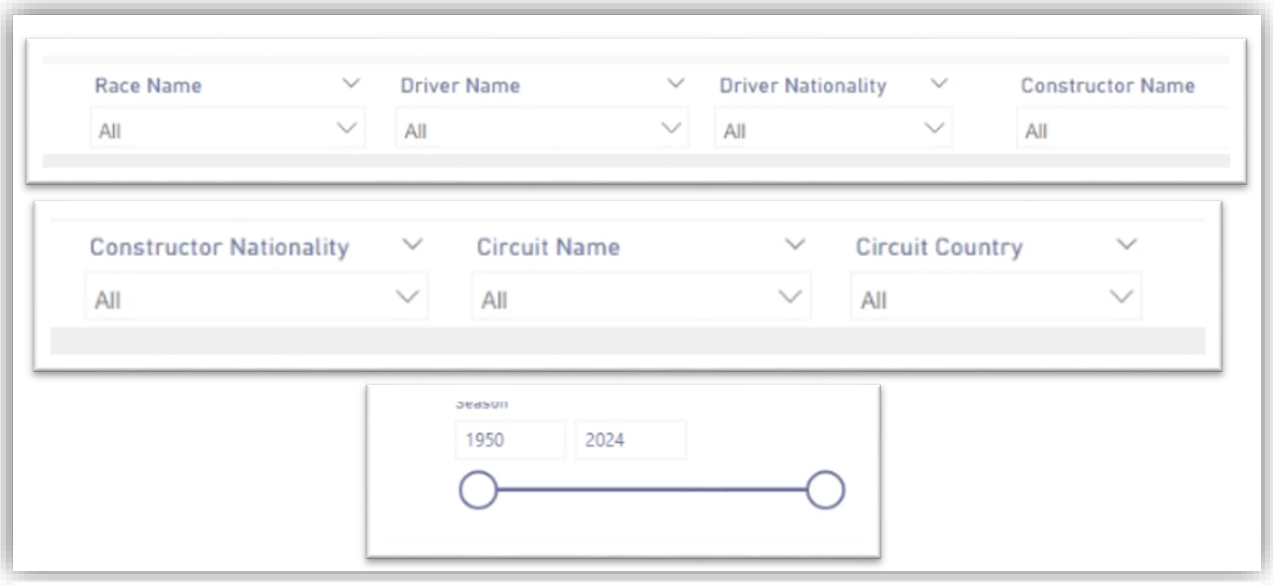

Figure 3.1 4

1. Race Name: Filters the visuals to display data for a specific race.

2. Driver Name: Filters the visuals to display data for a selected driver.

3. Driver Nationality: Filters data based on the nationality of drivers.

4. Constructor Name: Filters visuals based on the constructor associated with the driver.

5. Constructor Nationality: Filters data by the nationality of constructors.

6. Circuit Name: Filters data by the circuit where races were held.

7. Circuit Nationality: Filters data by the circuit nationality.

8. Seasons Slicer: Filter the duration of seasons.

3.1.4 Graphs & Charts

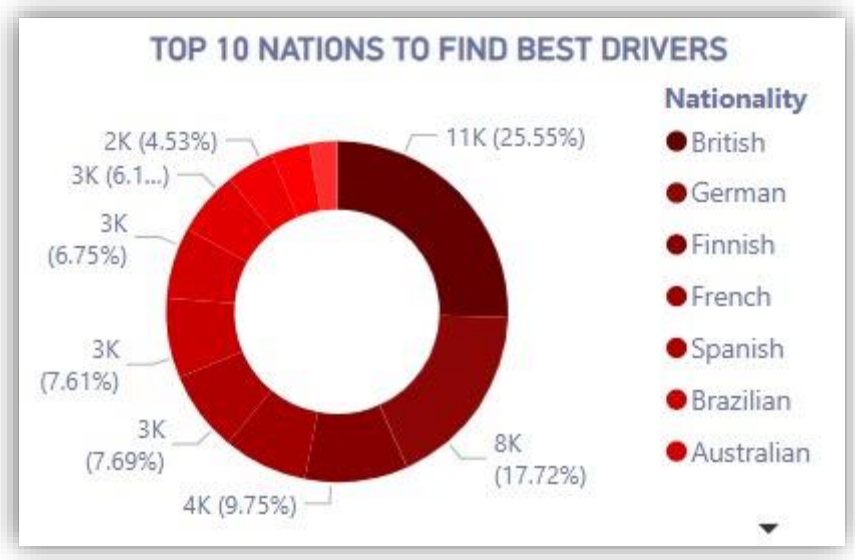

Figure 3.1 5

Figure 3.1.3: (Donut) Top 10 Nations to Find Best Drivers

Displays the nationality distribution of the top 10 drivers.

Insights: Countries like the UK, Germany, and Finland have produced many successful drivers.

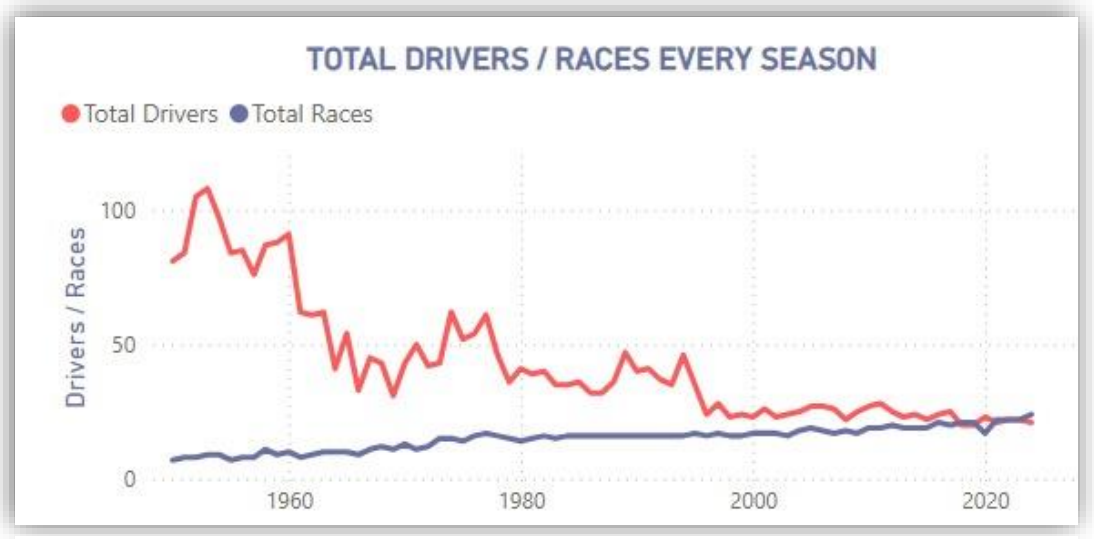

Figure 3.1 6

Figure 3.1.6: (Line Chart)Total Drivers/Races Every Season

Tracks the growth in the number of drivers and races over time.

Insights: A significant increase in races and participating drivers is observed after the 1980s, reflecting the sport's globalization.

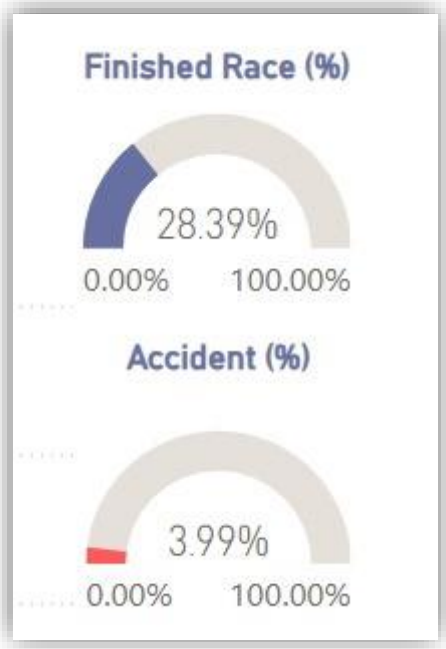

Figure 3.1 7

TOP RACERS

Driver Name	Nationality	Total Points
Adrian Sutil	German	124
Aguri Suzuki	Japanese	8
Alain Prost	French	798
Alan Brown	British	2
Alan Jones	Australian	206
Alberto Ascari	Italian	139
Alessandro Nannini	Italian	65
Alessandro Zanardi	Italian	1

Figure 3.1 8

Figure 3.1.7: (Gauges) Finished Race (%) and Accident (%)

Finished Race (%): Shows the percentage of races completed by drivers.

Accident (%): Reflects the percentage of races ending in accidents.

Figure 3.1.8: (Table) Top Racers

Provides a list of top-performing drivers, including their nationality and total points.

Insights: Allows quick identification of the most successful drivers in Formula 1 history.

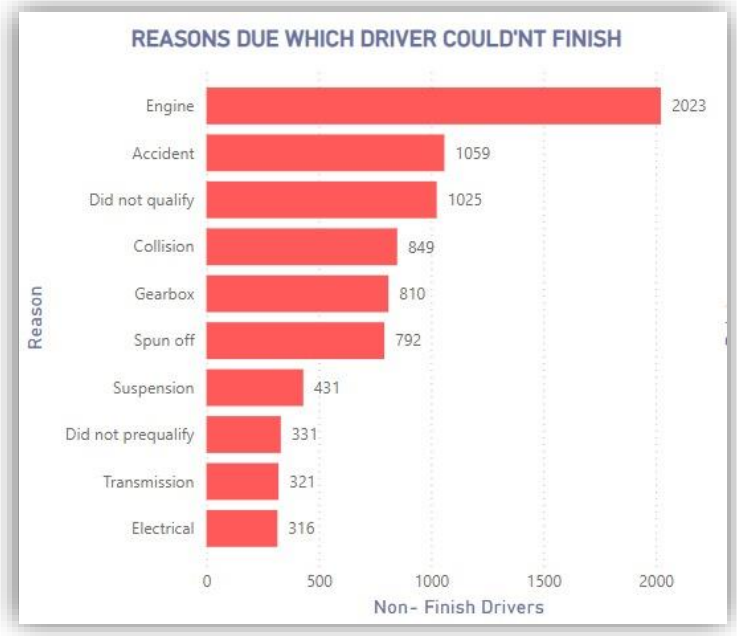

Figure 3.1 9

Figure 3.1.9: (Bar Chart) Reasons Drivers Couldn’t Finish

Highlights the most common reasons for non-finishes (DNFs).

Insights: Drivers starting closer to the front (lower grid numbers) generally score more points, showing the importance of qualifying.

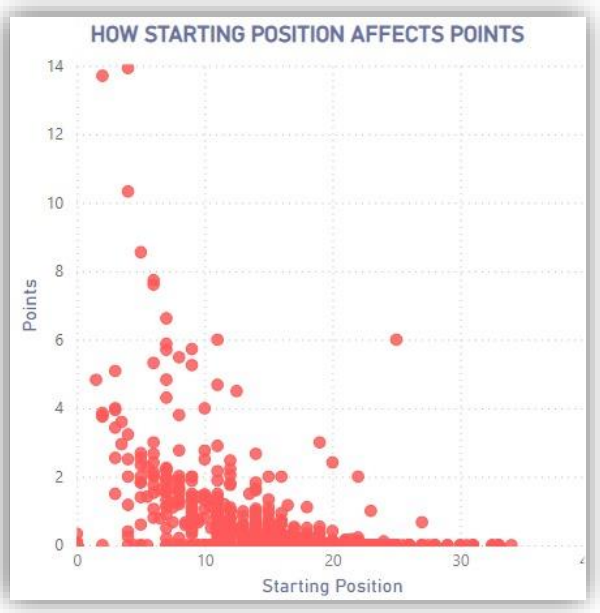

Figure 3.1.10

Figure 3.1.10: (Scatter Plot) How Starting Position Affects Points

Visualizes the relationship between starting grid positions and points scored.

Insights: Drivers starting closer to the front (lower grid numbers) generally score more points, showing the importance of qualifying.

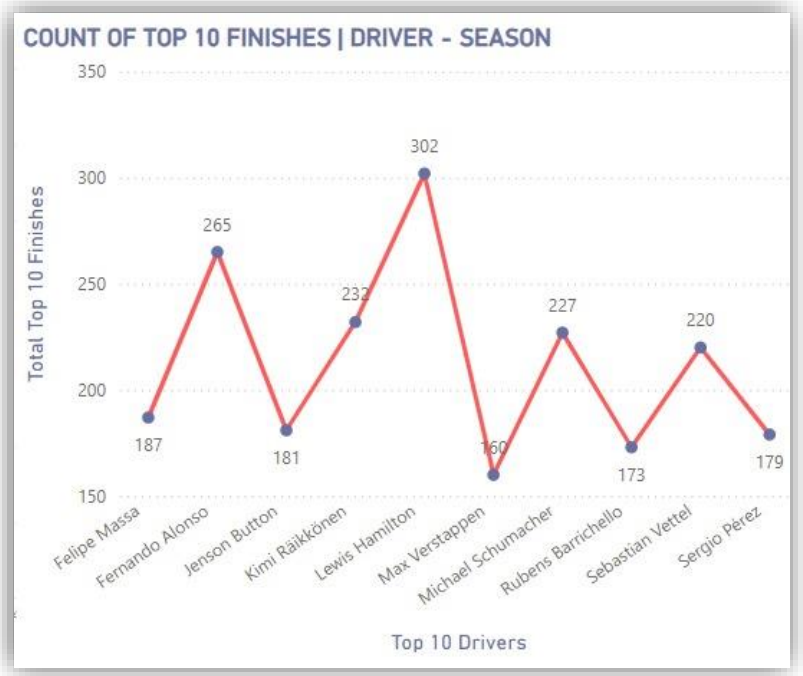

Figure 3.1 11

3.2 Constructors Dashboard

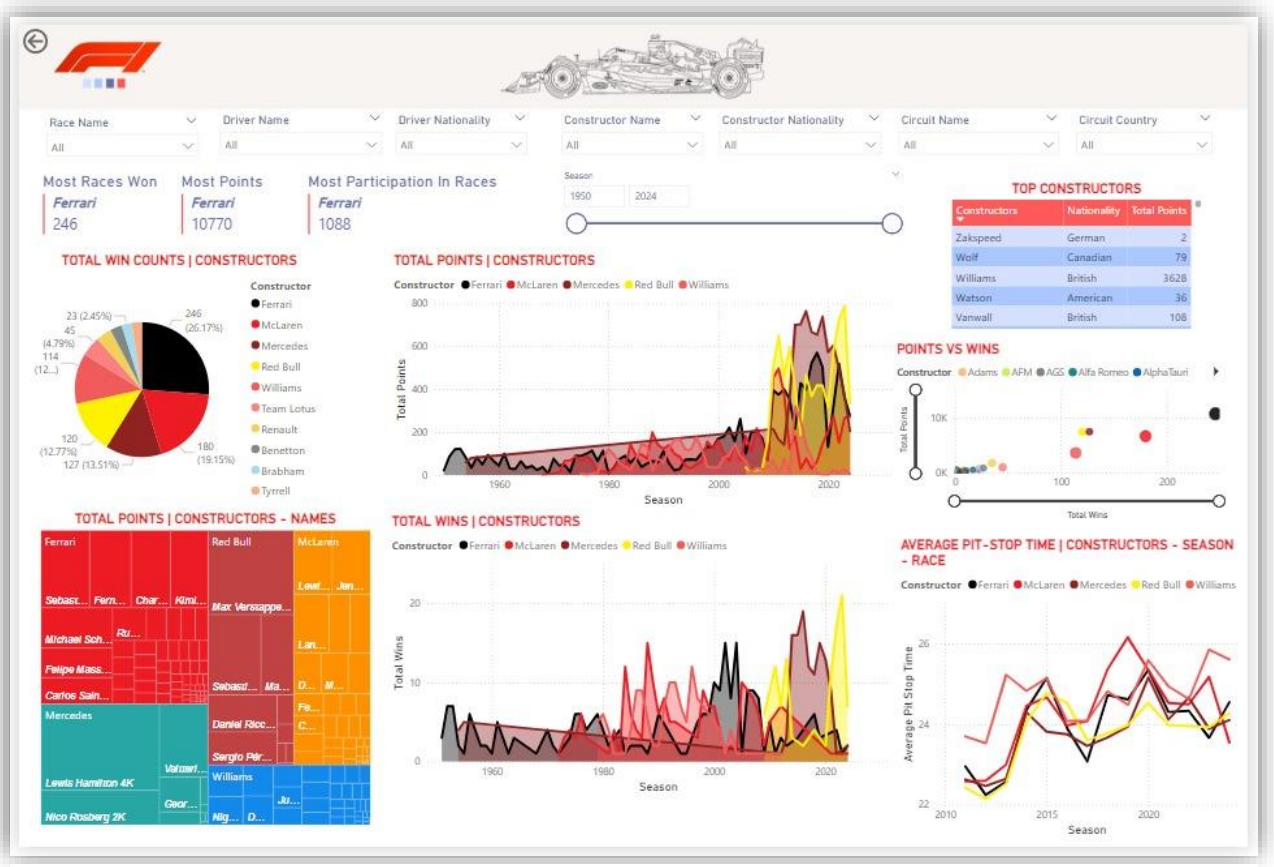

Figure 3.2 1

3.2.1 Multi-Row Cards

Figure 3.2 2

Most Races Won: Shows the constructor with the highest number of race wins (e.g., Ferrari with 246 wins).

Most Points: Displays the constructor with the most accumulated points (e.g., Ferrari with 10,770 points).

Most Participation in Races: Highlights the constructor with the most race participations (e.g., Ferrari with 1,088 races).

3.2.2 Slicers

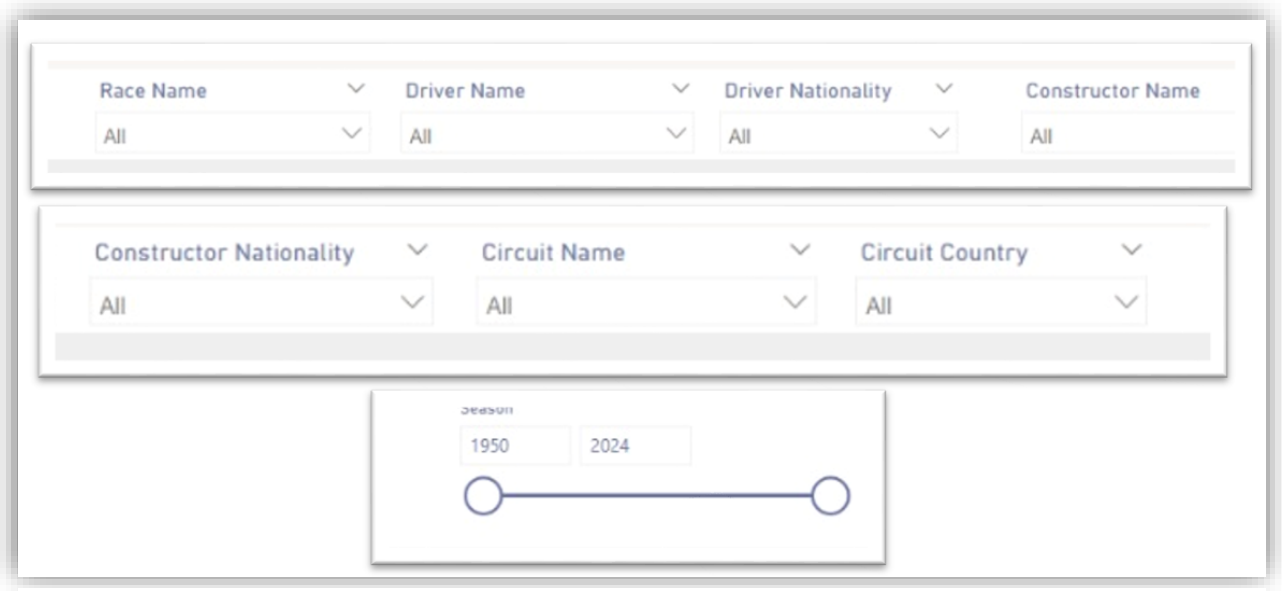

Figure 3.2 3

1. Race Name: Filters the visuals to display data for a specific race.

2. Driver Name: Filters the visuals to display data for a selected driver.

3. Driver Nationality: Filters data based on the nationality of drivers.

4. Constructor Name: Filters visuals based on the constructor associated with the driver.

5. Constructor Nationality: Filters data by the nationality of constructors.

6. Circuit Name: Filters data by the circuit where races were held.

7. Circuit Nationality: Filters data by the circuit nationality.

8. Seasons Slicer: Filter the duration of seasons.

3.2.3 Graphs & Charts

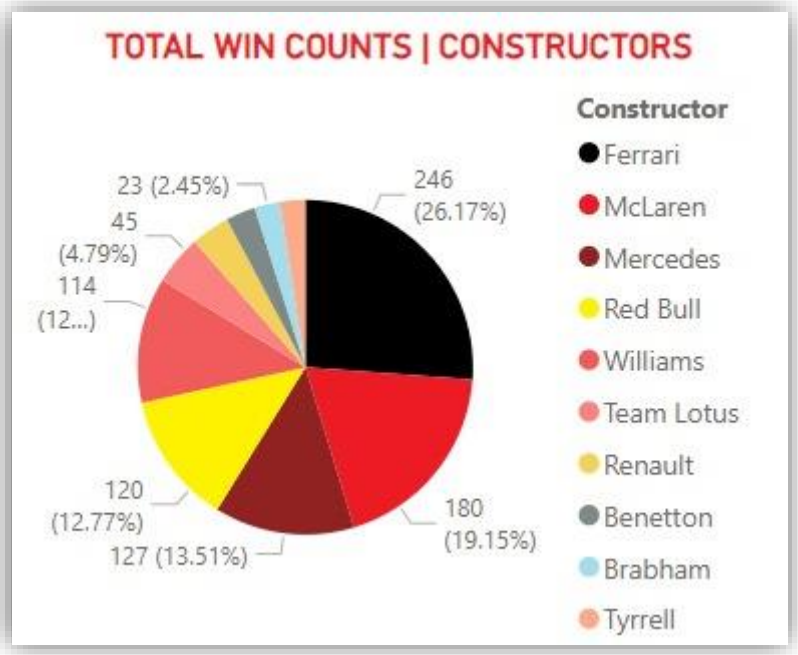

Figure 3.2 4

Figure 3.2.4: (Pie Chart) Total Win Counts | Constructors

Visualizes the distribution of race wins among constructors.

Constructors	Nationality	Total Points
Zakspeed	German	2
Wolf	Canadian	79
Williams	British	3628
Watson	American	36
Vanwall	British	108

Figure 3.2 5

Figure 3.2.5: (Table) Top Constructors

Displays the top-performing constructors, including their nationality and total points.

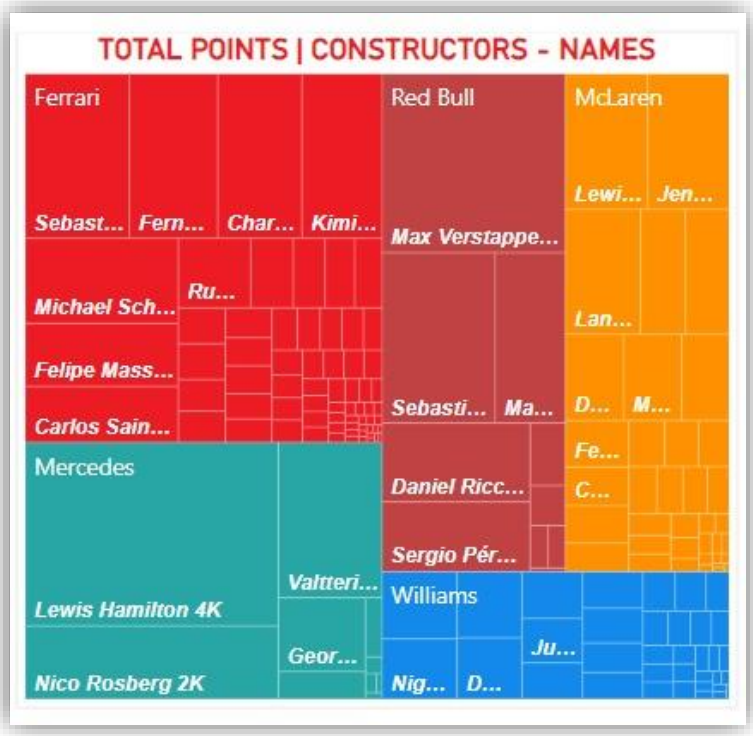

Figure 3.2 6

Figure 3.2.6: (Tree Map) Total Points | Constructors – Driver Names

Breaks down total points scored by constructors and their respective drivers.

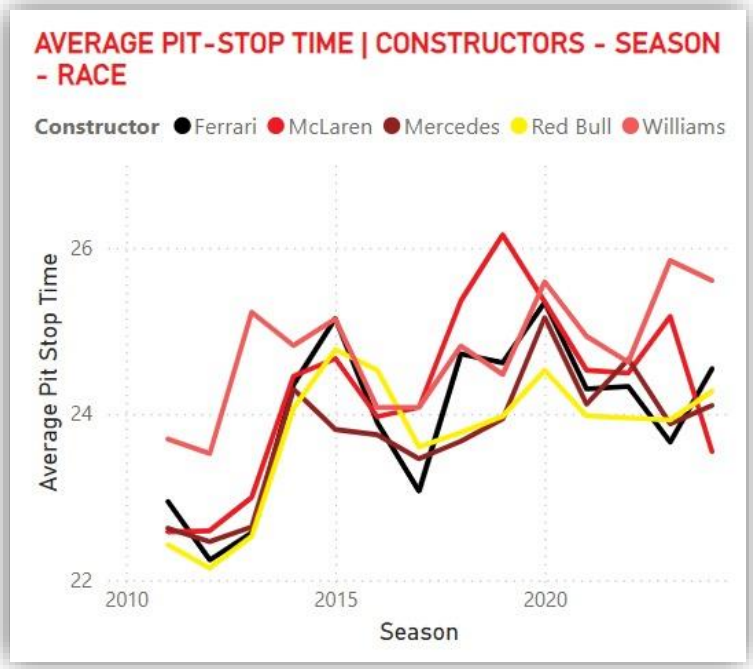

Figure 3.2 7

Figure 3.2.7: (Line Chart) Average Pit Stop Time by Constructor

Tracks the average pit stop time for constructors across seasons.

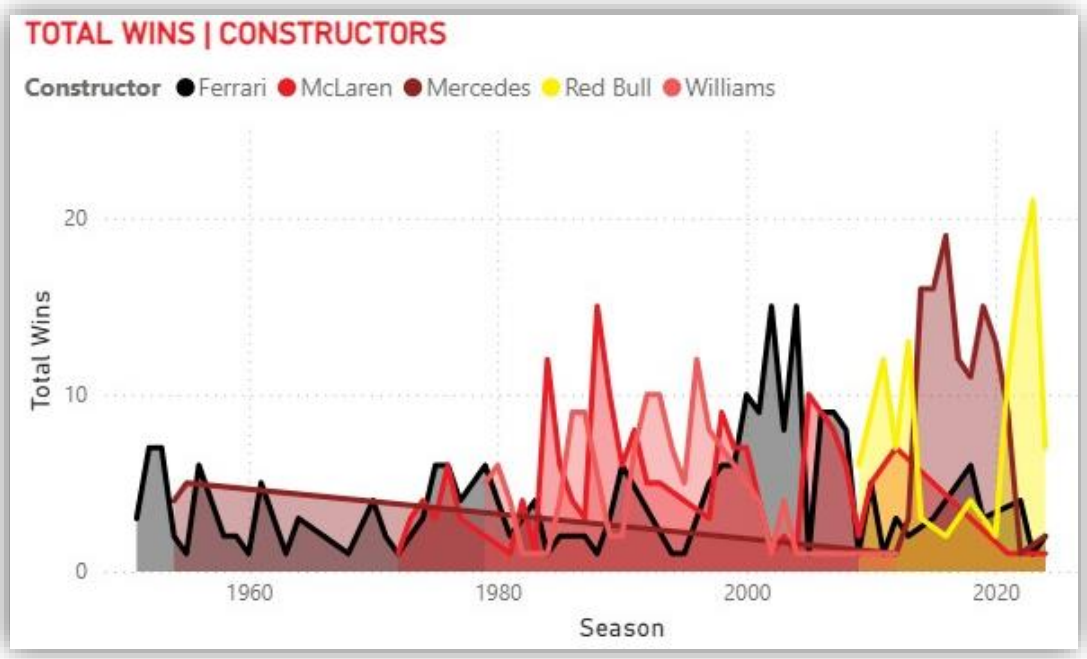

Figure 3.2 8

Figure 3.2.8: (Area Chart) Total Wins Over The Seasons

Breaks down total wins of constructors by seasons..

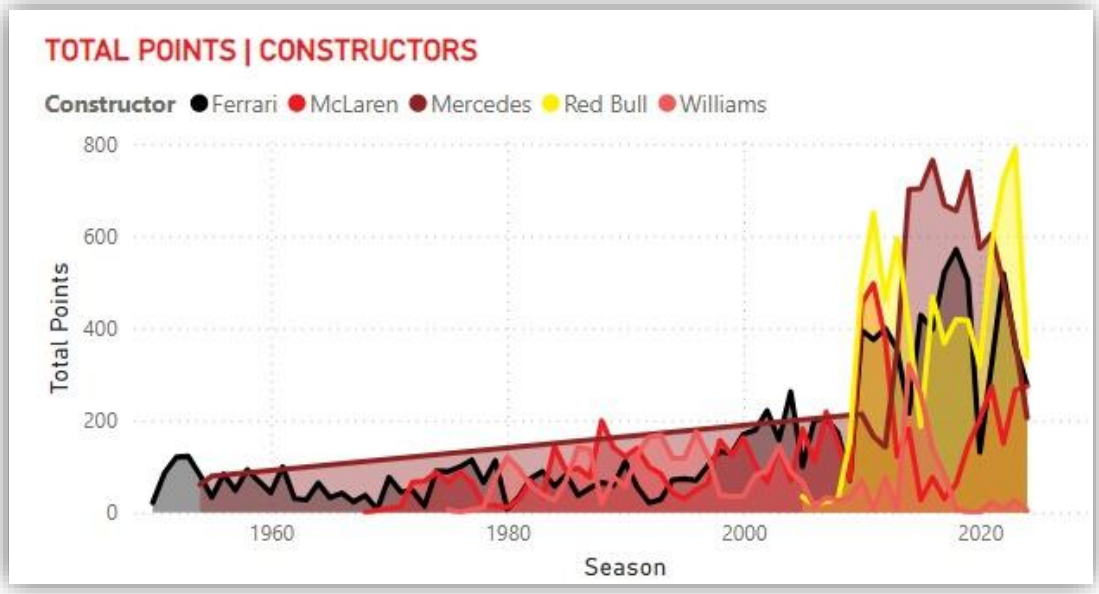

Figure 3.2 9

Figure 3.2.9: (Area Chart) Total Points Over The Seasons

Breaks down total points of constructors by seasons

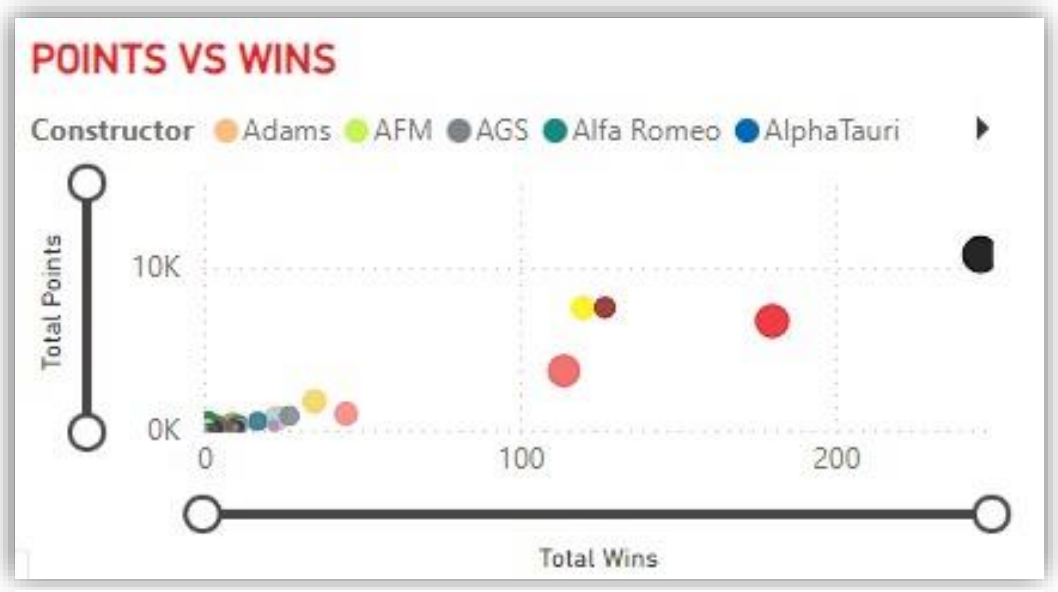

Figure 3.2 10

Figure 3.2.10: (Bubble Chart) Points VS Wins

Breaksdown the total points vs total wins of the constructors. The size shows the participated races.

Chapter 4: Insights and Recommendations

4.1 Driver Insights

1. Top 10 Nations to Find Best Drivers:
   Figure 3.1.5 Shows that countries like the UK, Germany, and Finland have produced many successful drivers.

2. Total Drivers/Races Every Season:
   A significant increase in races and participating drivers is observed after the 1980s, reflecting the sport's globalization.

3. Reasons Drivers Couldn't Finish:
   Engine failures dominate the list, followed by accidents and disqualifications, highlighting reliability challenges in car engineering.

4. How Starting Position Affects Points:
   Drivers starting closer to the front (lower grid numbers) generally score more points, showing the importance of qualifying.

5. Count of Top 10 Finishes | Driver-Season:
   Highlights consistency and top performance for key drivers, with peaks in specific years reflecting dominant seasons by certain drivers.

6. Finished Race (%) and Accident (%):
   High finished race rates indicate improved reliability over the years. A small accident percentage highlights the safety measures implemented in recent seasons.

7. Top Racers (Table):
   Allows quick identification of the most successful drivers in Formula 1 history.

4.2 Constructors Insights

1. Total Win Counts | Constructors:
   Ferrari leads with the highest percentage of wins, followed by dominant teams like McLaren and Mercedes.

2. Total Points | Constructors:
   Consistent dominance by teams like Ferrari, McLaren, and Mercedes, with spikes reflecting successful seasons.

3. Points vs. Wins:
   Teams with more wins tend to have higher points, reflecting the scoring system's emphasis on victories. Outliers represent teams that performed well in points without winning many races.

4. Total Wins | Constructors:
   Shows periods of dominance for teams like Ferrari and Mercedes and highlights shifts in competitiveness over decades.

5. Total Points | Constructors - Names:
   Reflects the contribution of key drivers to the constructors' success and highlights standout performers within each team.

6. Average Pit Stop Time by Constructor:
   Teams with shorter pit stop times tend to be more competitive. Improvements in pit stop efficiency over time indicate advancements in teamwork and technology.

7. Top Constructors (Table):
   Highlights the dominance of teams like Ferrari, McLaren, and Mercedes and provides a quick comparison of top constructors.

Chapter 5: Conclusion

This report provides a data-driven analysis of Formula 1, highlighting the dominance of iconic drivers like Lewis Hamilton and constructors such as Ferrari. Key insights include the impact of grid position, pit stop efficiency, and race reliability on performance, alongside the sport's evolution with increasing race. The dashboards showcase how consistent performance, strategic excellence, and technological innovation drive success, offering valuable insights into the history and dynamics of one of the world’s most competitive sports.

Chapter 6: References

- https://learn.microsoft.com/en-us/dax/
- F1 Dashboard
- https://learn.microsoft.com/en-us/power-bi/