Formula1 with Databricks Delta Lake

SHEFALI BISHT

About Me

Shefali Bisht, Data Engineer

https://www.linkedin.com/in/shefali-bisht/

shefalibisht00@gmail.com

https://medium.com/@shefalibisht00

Formula1 Project Overview

Goal: Capture Formula1 race data periodically from an external API and store it so that we can analyze it in the future.

Technical Stack:

- **Delta Lake :** Build Lakehouse architecture on top of Data Lakes.
- **Azure Databricks :** Configure Delta Lake based on our workload patterns and perform ETL.
- **Azure Data Lake Storage Gen2 :** Store raw, refined and aggregates data (Storage layer).
- **Azure Data Factory :** Author, scheduled and monitor the ETL pipeline.
- **Power BI :** Dashboards and reports to derive insights.

Two Workflows

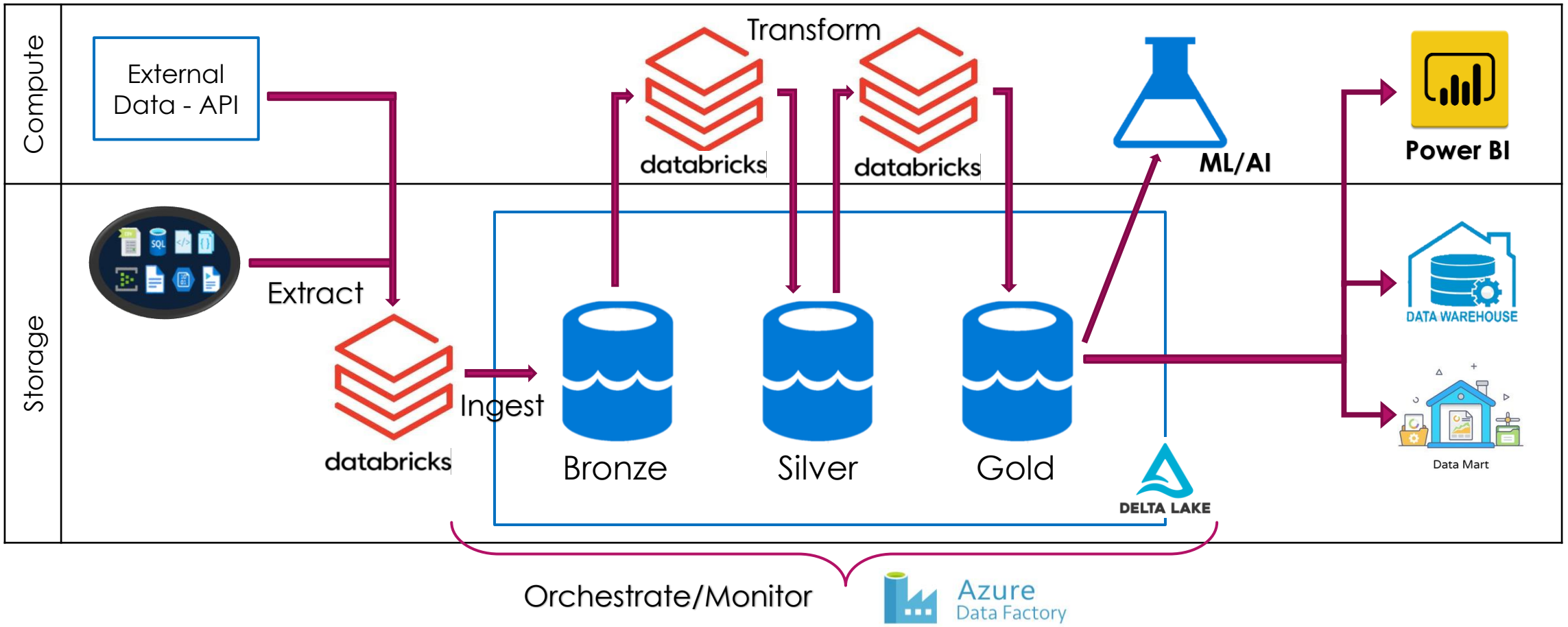
Provisioning Workflow

1. Provision Azure Data Lake Storage Gen2.
2. Provision Azure Databricks.
3. Provision Azure Key Vault
4. Provision Azure Data Factory.
5. Provision Power BI.

Holistic Workflow

1. Create a Azure Databricks notebook that will – ingest, clean, transform and analyze data from Ergast API
2. Store data as Delta tables in Delta Lake.
3. Orchestrate and automate the entire workflow via Azure Data Factory.
4. Connect Power BI to Databricks to consume Gold level data for visualization and insights.

Formula-1 Data Architecture

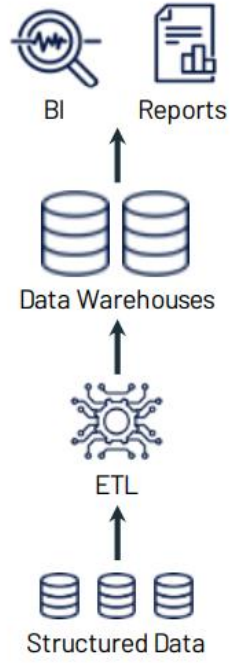


Why Delta Lake?

- Delta Lake is a project originally developed by databricks and then open sourced under the Linux Foundation license around late 1990.
- Before, diving into Delta Lake, lets first talk about Data warehouses and Data Lakes…

Data Warehouse

- Data warehouses came into existence in the 1980s when businesses wanted to make decisions based on all of the data available in an organization in one place, rather than looking at data in individual departments.
- A data warehouse mainly consisted of operational and external data available within the organization to make intelligent decisions.
- The data received in a data warehouse is mainly structured as SQL tables, or CSV files or semi-structured such as JSON or XML files. Data is cleaned, de-duped, validated, and augmented with business meaning before loading into a warehouse or data mart.
- They highly aggregated data is then consumed for BI reporting.



Data Warehouse Pitfalls

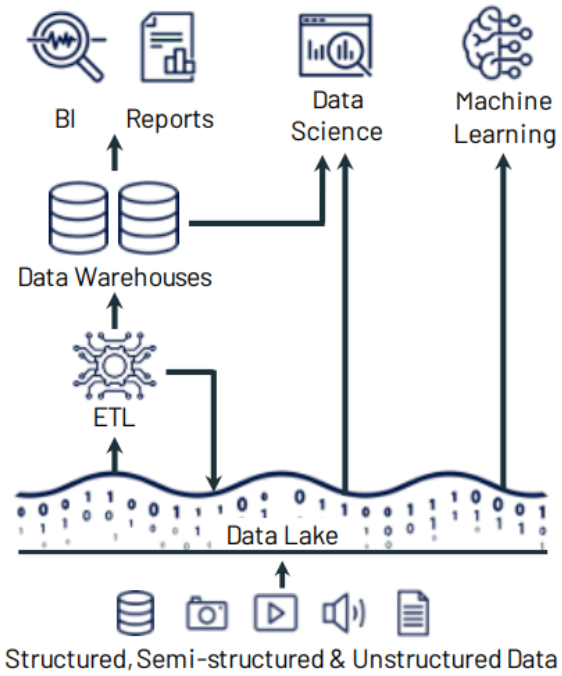
- By the early 2000s, due to the internet boom, there was a tremendous increase in volume and velocity of data as well as the variety of the data had started to change as well. We started to see unstructured data such as videos, images, text files, etc., which were valuable for making decisions.
- However, the data warehouses lacked the support for such unstructured data. Data in a data warehouse was only loaded after the data quality has been checked and also once it has been transformed. This meant it took longer to develop a solution to get new data into the warehouse.
- Data warehouses were built on traditional relational databases or MPP (Massive Parallel Processing) solutions, which meant that they used proprietary file formats and resulted in vendor logins. Also, the traditional on-premise data warehouses were very difficult to scale or at times impossible, which resulted in large data migration projects to scale up those databases. Storage was expensive with large vendors and also it wasn't possible to scale up storage without computing.
- Finally, data warehouses didn't provide sufficient support for data science or ML/AI workloads.

Data Warehouse Pitfalls - Summary

- Inability to handle unstructured data
- Longer to ingest new data
- Propriety data formats and vendor lock-ins
- Difficult to scale
- Expensive storage
- Lack of ML/AI or data science support

Data Lake

- Date lake architecture was aimed at solving the data warehouse pitfalls.
- They can handle structured, semi structured, and unstructured data (~90% of today's data). The data received is ingested into a data lake without any kind of cleansing or transformation, resulting in quicker timescales to develop solutions and faster injection times.
- Storage like HDFS, ADLS or S3 were really cheap, enabling organizations to just dump their enormous data at a very low cost.
- Data lakes were built on open source file formats such as Parquet or Avro, enabling us to use a wide variety of software and libraries to process and analyze the data. Data science and the machine learning workloads could use either the raw or transformed data in the data lake.



Data Lake Pitfalls

- Data Lakes were however, very slow in servicing interactive BI reports, and there was a lack of data governance.
- This made data to be first ETLed into lakes, and then again ELTed into warehouses, creating complexity, delays, and new failure modes.
- It resulted in a clunky architecture with too many moving parts.

Data Lake Pitfalls - Summary

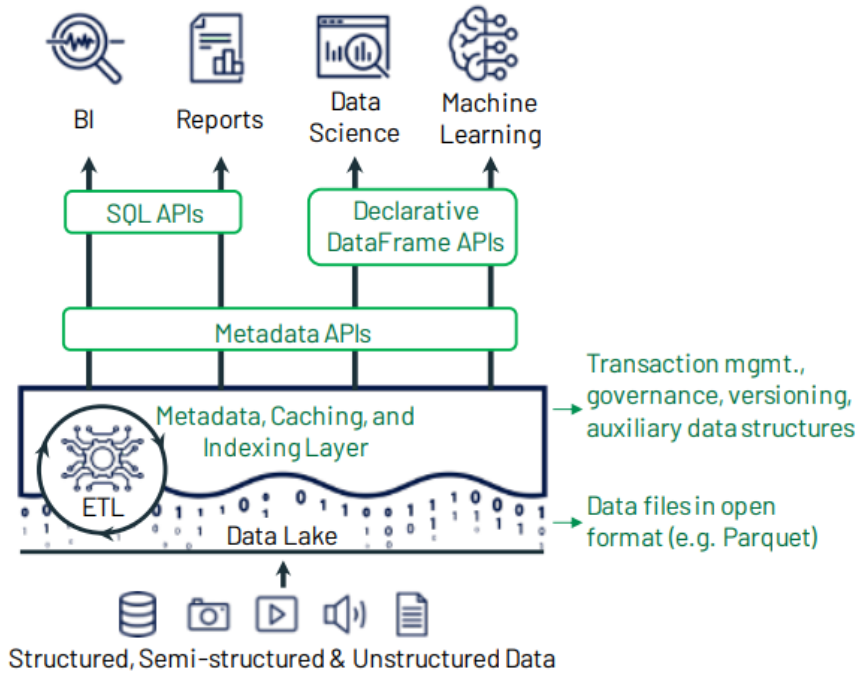
- No support for ACID transactions. Due to this failed jobs left partially loaded files which had to be cleaned up in separate processes during each rerun.
- Inconsistent reads making user consume unreliable data.
- Extremely difficult to handle data corrections as data lake offer no support for updates. So developers had to partition the data and rewrite the entire partition. This resulted in increased development time.
- No data rollback feature. Either we had to rewrite a partition or rewrite an entire table.
- Lack of ability to delete data for regulations like GDPR.
- No history or versioning, which often creating data swamps.
- Poor performance in terms of BI support, interactive query, security or data governance.
- We needed to process streaming data and batch data separately, resulting in complex Lambda architecture.

Delta Lake

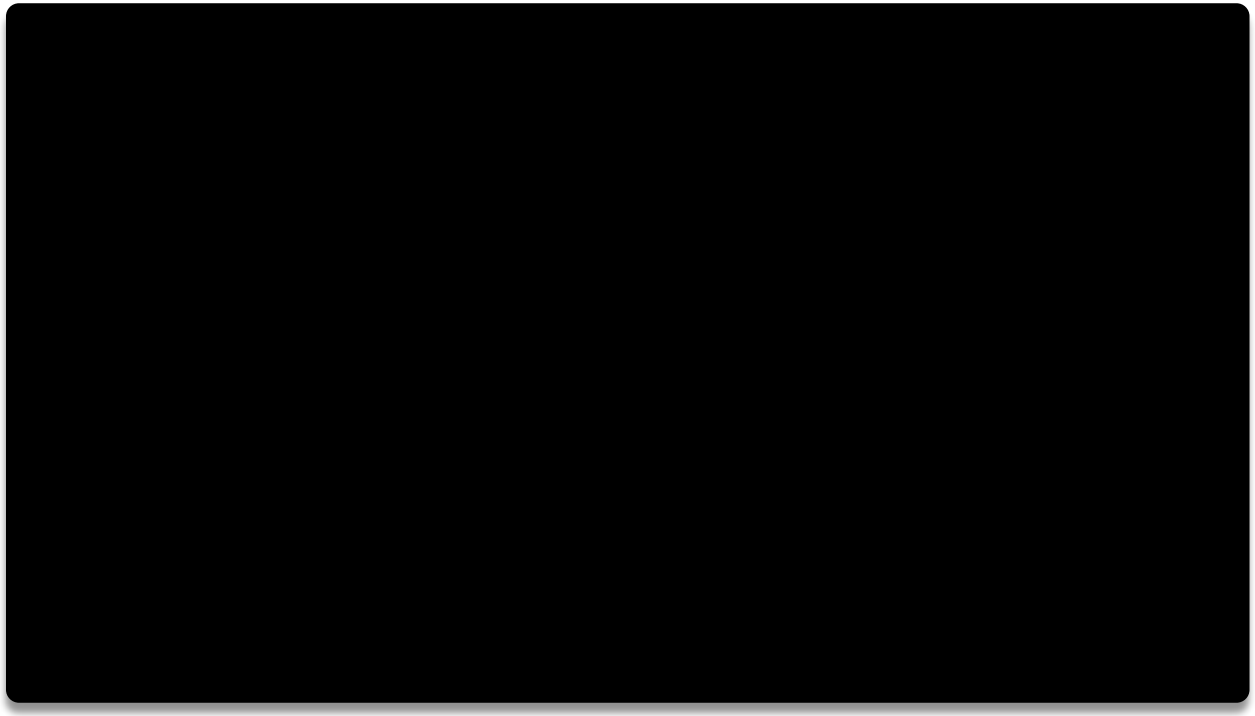
- Data Lakehouse architecture is aimed at bringing the best of both data warehouse as well as data lake. They are designed to provide BI, data science and machine learning support all together.
- In a nutshell, Delta Lake is an open-source storage layer that brings reliability to data lakes, providing ACID transactions, scalable metadata handling, auditing, indexing, and unifying streaming and Batch Workloads, thus eliminating the complex lambda architecture.
- Delta Lakes runs on top of data lakes, and they are fully compatible with Apache Spark APIs.
- The ingested data could then be transformed more efficiently without the need to rewrite an entire partitions in cases of reprocessing data or rewriting data lakes in case of processing GDP requests.

Delta Lake - Pros

- Handles all kinds of data
- Uses open source format
- Inexpensive cloud object storage
- ACID support
- Supports all kinds of workloads (BI or ML/AI)
- History and version control
- Simple architecture
- Better performance



Azure ADF pipeline

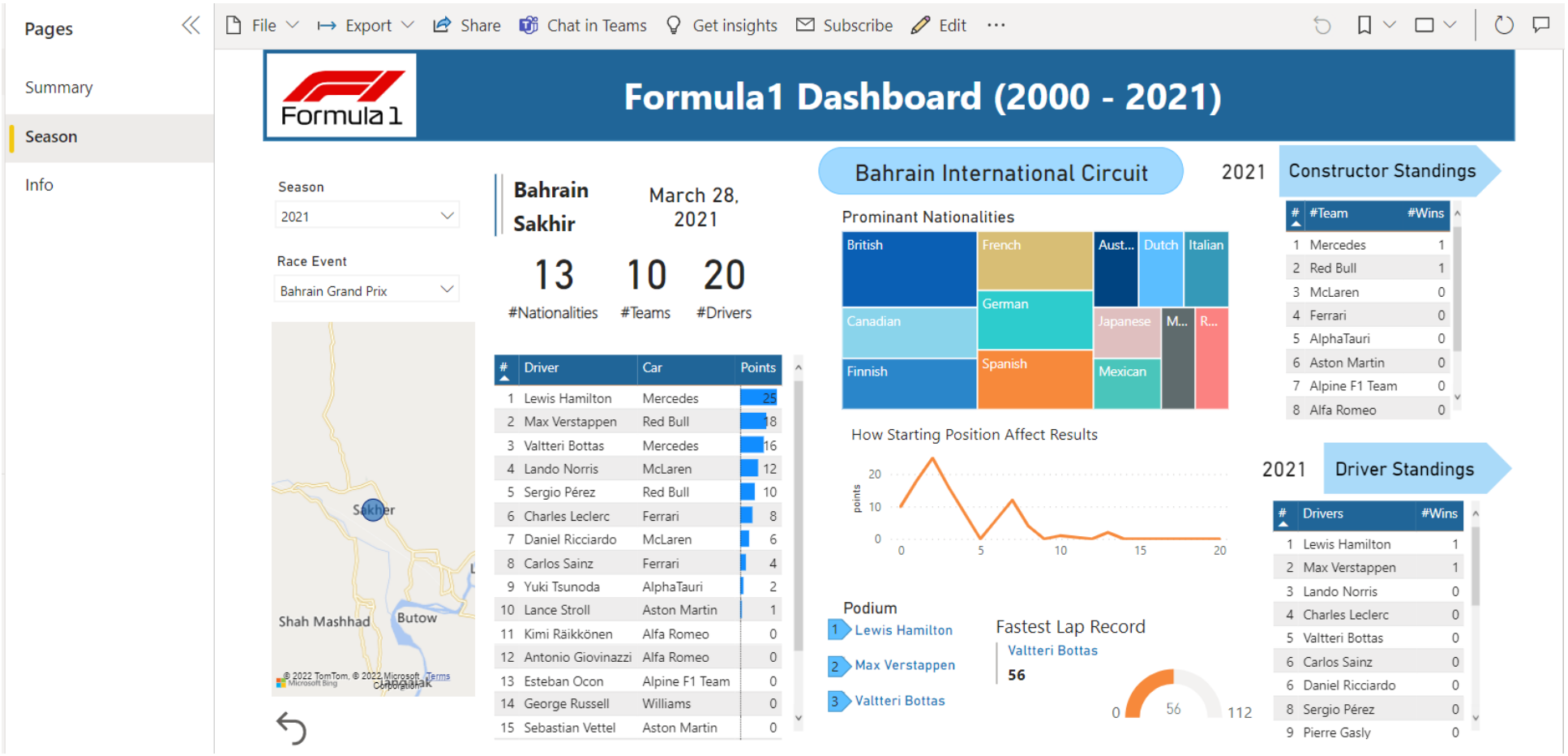


- Automate databricks notebooks from Azure data factory.
- Schedule and monitor from the portal

Power BI


Pages
Summary
Season
Info
File
Export
Share
Chat in Teams
Get insights
Subscribe
Edit
Formula 1
Formula1 Dashboard (2000 - 2021)
Year
2000
2021
Racer
All
Race
All
Highest Win Rate
Lewis Hamilton
35.82%
Most Winning Racer
Lewis Hamilton
96
26
#Country
33
#Nationalities
391
#Races
33
#Location
37
#Teams
118
#Drivers
Team
Wins
Points
Points by Year
Ferrari 113 6066
Mercedes 107 5745
Red Bull 65 5095
McLaren 59 3487
Williams 11 1571
Renault 20 1465
Force India 1098
Lotus F1 2 706
Toro Rosso 1 500
Sauber 473
BMW Sauber 1 308
Racing Point 1 283
Toyota 276
BAR 227
Wins by Racer
Lewis Hamilton
Michael Schumacher
Sebastian Vettel
Fernando Alonso
Nico Rosberg
Kimi Räikkönen
Jenson Button
Felipe Massa
0
50
100
#Drivers by Year
25
20
2000
2010
2020
Fastest Lap Record
Lance Stroll
85
Lando Norris
85
0
85
170
Win% by Nationality
Australian 3.70%
German 8.64%
Dutch 15.38%
Finnish 10.53%
British 14.67%
Top Circuit Location
Bahrain International C...
Autodrom...
Circuit de ...
Circuit de ...
Red Bull Ring
Hungaroring
Yas Marina...
Silverstone Circuit
Sochi Autodrom
Developed by Shefali Bisht

Power BI


Pages
Summary
Season
Info
File
Export
Share
Chat in Teams
Get insights
Subscribe
Edit
Formula 1
Formula1 Dashboard (2000 - 2021)
Season
2021
Race Event
Bahrain Grand Prix
Bahrain
Sakhir
March 28, 2021
13
#Nationalities
10
#Teams
20
#Drivers
Bahrain International Circuit
2021
Prominant Nationalities
British
French
Aust...
Dutch
Italian
German
Canadian
Japanese
M...
R...
Finnish
Spanish
Mexican
How Starting Position Affect Results
points
Podium
1 Lewis Hamilton
2 Max Verstappen
3 Valtteri Bottas
Fastest Lap Record
Valtteri Bottas
56
0
56
112
Constructor Standings
#Team #Wins
1 Mercedes 1
2 Red Bull 1
3 McLaren 0
4 Ferrari 0
5 AlphaTauri 0
6 Aston Martin 0
7 Alpine F1 Team 0
8 Alfa Romeo 0
2021 Driver Standings
Drivers #Wins
1 Lewis Hamilton 1
2 Max Verstappen 1
3 Lando Norris 0
4 Charles Leclerc 0
5 Valtteri Bottas 0
6 Carlos Sainz 0
6 Daniel Ricciardo 0
8 Sergio Pérez 0
9 Pierre Gasly 0
Driver Car Points
1 Lewis Hamilton Mercedes 25
2 Max Verstappen Red Bull 18
3 Valtteri Bottas Mercedes 16
4 Lando Norris McLaren 12
5 Sergio Pérez Red Bull 10
6 Charles Leclerc Ferrari 8
7 Daniel Ricciardo McLaren 6
8 Carlos Sainz Ferrari 4
9 Yuki Tsunoda AlphaTauri 2
10 Lance Stroll Aston Martin 1
11 Kimi Räikkönen Alfa Romeo 0
12 Antonio Giovinazzi Alfa Romeo 0
13 Esteban Ocon Alpine F1 Team 0
14 George Russell Williams 0
15 Sebastian Vettel Aston Martin 0
Sakhir
Shah Mashhad
Butow

Thank You!