



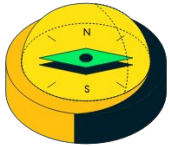
# F1 Base

Data Management project

Michele Nicoletti - 1886646

Lorenzo Pecorari - 1885161

## DM tool: MongoDB



- NoSQL DBMS
- collections and BSON documents
- flexible and fast
- advanced **query** operations
- Compass, mongosh, Atlas and pymongo

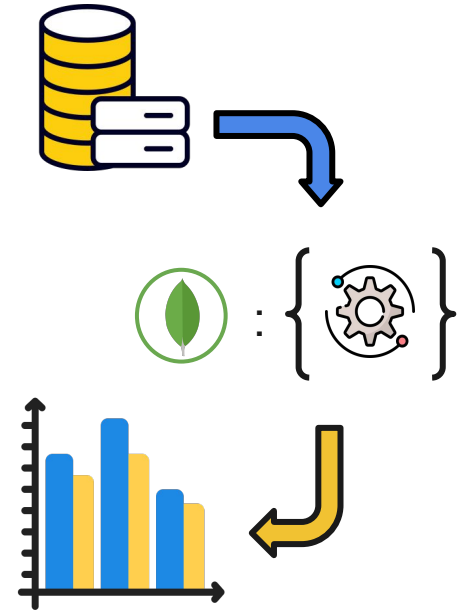
# Dataset and Data Modelling

- [Kaggle: Formula 1 World Championship \(1950 - 2024\)](#)
- circuits, drivers, constructors, races, standings, qualifyings, results and others
- csv files imported as collections with **referenced** model
- queries generating **embedded-modelled** data

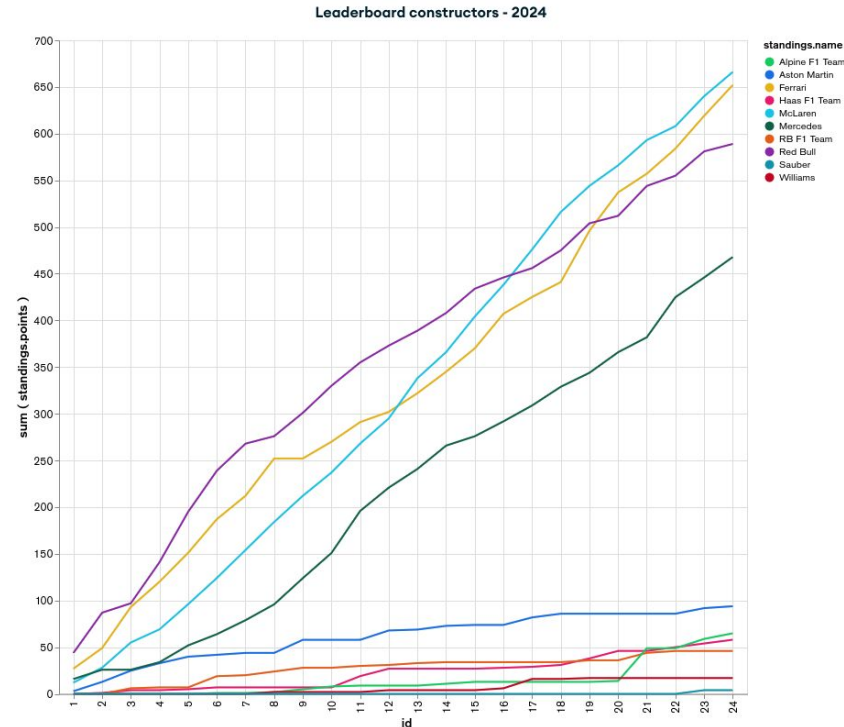


# Objectives of the project

- dataset **exploration** to investigate:
  - constructors' performances
  - driver career performances
  - WDC and WCC wins among the seasons
  - failures and retirements
- Acquire knowledge about a **new** DM tool

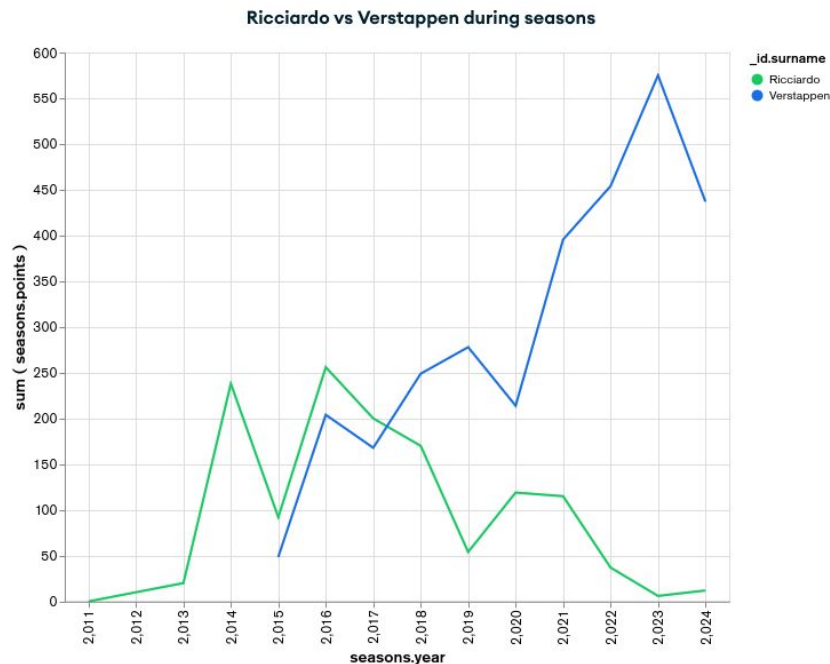
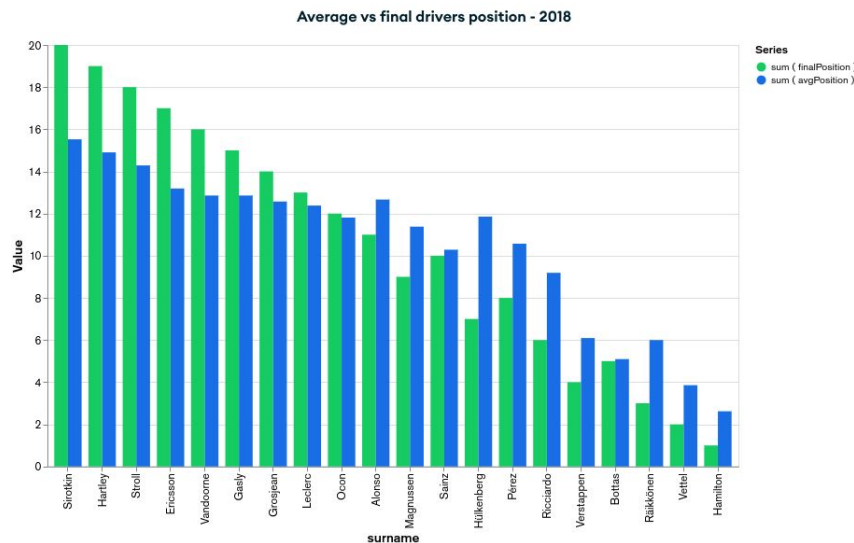


- 
- The bar chart displays the points scored by McLaren in each season from 1970 to 2023. The y-axis represents the number of points, ranging from 0 to 600. The x-axis represents the year, with labels every 10 years from 1970 to 2020. The chart shows a general upward trend in points over the years, with a significant peak in 2023, where McLaren scored over 600 points. There are also notable peaks around 1989 and 2010.
- | Year | Points |
|------|--------|
| 1970 | 5      |
| 1971 | 10     |
| 1972 | 50     |
| 1973 | 60     |
| 1974 | 70     |
| 1975 | 50     |
| 1976 | 70     |
| 1977 | 60     |
| 1978 | 20     |
| 1979 | 15     |
| 1980 | 15     |
| 1981 | 60     |
| 1982 | 30     |
| 1983 | 140    |
| 1984 | 90     |
| 1985 | 100    |
| 1986 | 80     |
| 1987 | 200    |
| 1988 | 140    |
| 1989 | 140    |
| 1990 | 130    |
| 1991 | 140    |
| 1992 | 120    |
| 1993 | 80     |
| 1994 | 30     |
| 1995 | 40     |
| 1996 | 60     |
| 1997 | 150    |
| 1998 | 120    |
| 1999 | 160    |
| 2000 | 100    |
| 2001 | 130    |
| 2002 | 160    |
| 2003 | 100    |
| 2004 | 180    |
| 2005 | 110    |
| 2006 | 180    |
| 2007 | 220    |
| 2008 | 150    |
| 2009 | 70     |
| 2010 | 450    |
| 2011 | 500    |
| 2012 | 380    |
| 2013 | 120    |
| 2014 | 180    |
| 2015 | 30     |
| 2016 | 80     |
| 2017 | 30     |
| 2018 | 140    |
| 2019 | 200    |
| 2020 | 270    |
| 2021 | 160    |
| 2022 | 300    |
| 2023 | 660    |



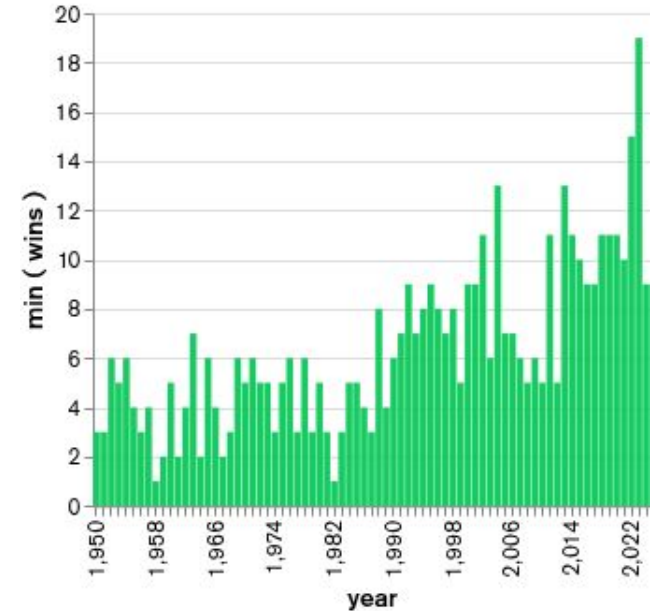
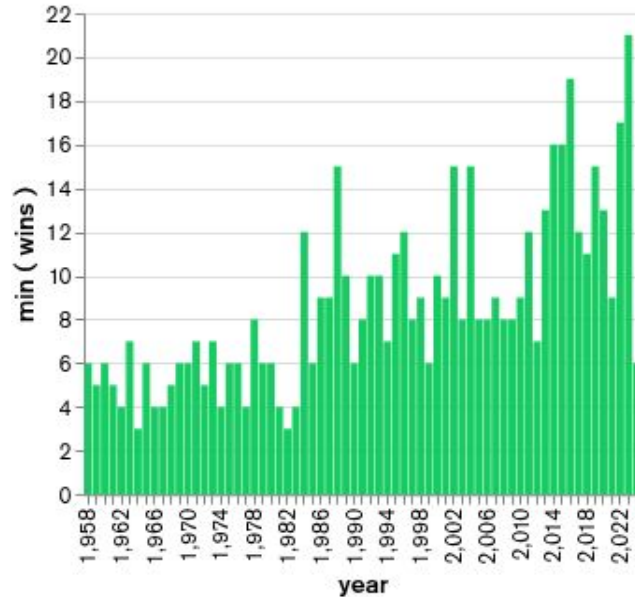
# Driver career performances

- average points scored in a **weekend**
- average points scored in a weekend per **constructors**
- **teammates** comparison in a season
- **final** position vs **average** position in a season



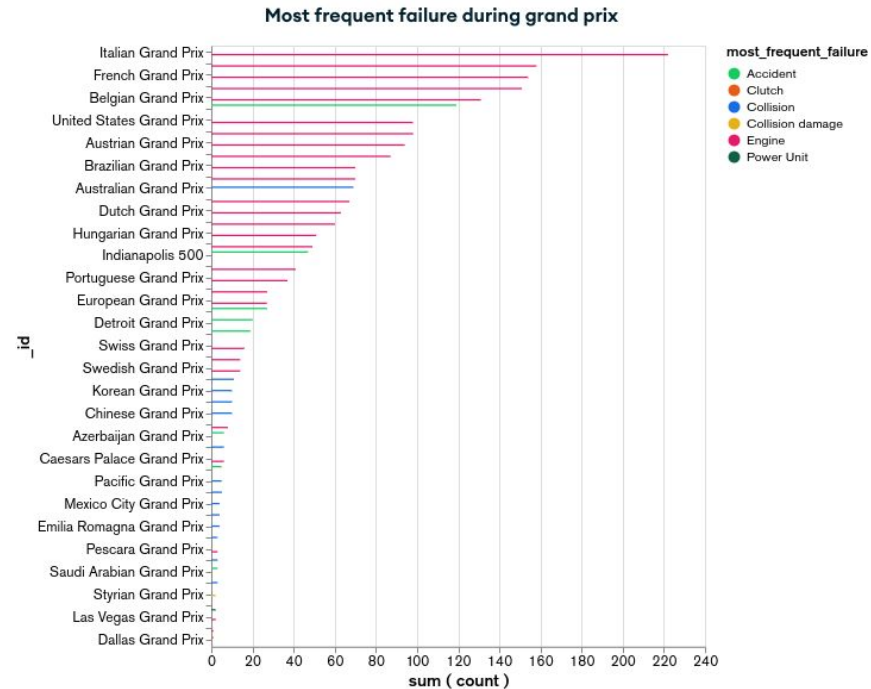
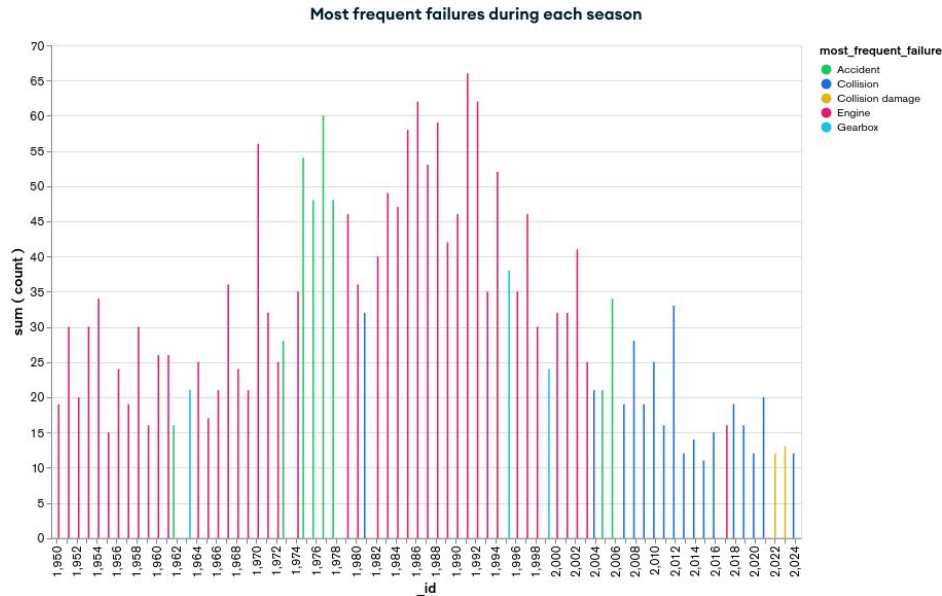
# WDC and WCC wins along the seasons

- number of wins of **constructors** champion along seasons
- number of wins of **drivers** champion during years




# Failures and retirements

- most frequent failures during all the seasons
- most frequent failure for each grand prix





# Query analysis: “Driver champion wins”



```
{
  $group: {
    _id: "$year",
    lastRound: {$max:"$round"}
  }
}
```

```
{ $lookup:
  { from: "races",
    let: {year: "$_id", round: "$lastRound"},
    pipeline: [
      { $match:
        { $expr: { $and:
          [ { $eq: ["$year", "$$year"] },
            { $eq: ["$round", "$$round"] } ] } } },
        { $project: { raceId: 1, year: 1 } }
      ],
      as: "lastRace"
    ]
  }
}
```

```
{ $unwind: "$lastRace" }
```

GROUP

LOOKUP

UNWIND

# Query analysis: *“Driver champion wins”*

```
{
  $project: {
    _id: 0,
    year: "$_id",
    driverId: "$standing.driverId",
    wins: "$standing.wins",
    name: "$driver.forename",
    surname: "$driver.surname"
  }
}
```

```
{
  $sort:{
    year:1
  }
}
```

PROJECT

SORT

# Indexes

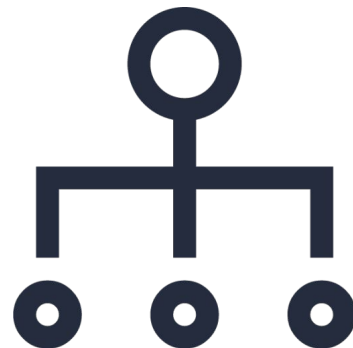
- adopted for improving performances
- single and combined
- defined for **most used** collections in queries

ex: indexes for collection “*racess*”

```
db.racess.createIndex({year: 1})
```

```
db.racess.createIndex({raceId: 1})
```

```
db.racess.createIndex({year: 1, round: 1})
```



# Indexes: feedback on “Driver champion wins”

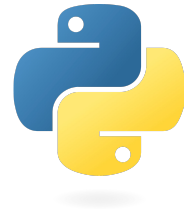
```
db.<COLLECTION>.aggregate(...).explain("executionStats")
```

```
{
  '$lookup': {
    from: 'driver_standings',
    as: 'standing',
    localField: 'lastRace.raceId',
    foreignField: 'raceId',
    let: {},
    pipeline: [
      { '$match': { position: { '$eq': 1 } } }
    ],
    unwinding: { preserveNullAndEmptyArrays: false }
  },
  totalDocsExamined: Long('2614725'),
  totalKeysExamined: Long('0'),
  collectionScans: Long('75'),
  indexesUsed: [],
  nReturned: Long('75'),
  executionTimeMillisEstimate: Long('715')
},
```

```
{
  '$lookup': {
    from: 'driver_standings',
    as: 'standing',
    localField: 'lastRace.raceId',
    foreignField: 'raceId',
    let: {},
    pipeline: [
      { '$match': { position: { '$eq': 1 } } }
    ],
    unwinding: { preserveNullAndEmptyArrays: false }
  },
  totalDocsExamined: Long('3192'),
  totalKeysExamined: Long('3192'),
  collectionScans: Long('0'),
  indexesUsed: [ 'raceId_1' ],
  nReturned: Long('75'),
  executionTimeMillisEstimate: Long('42')
},
```

# Data analysis through a script

- interaction between MongoDB and an application
- connection to the **cluster**
- querying a database of the cluster
- data analysis on JSON-like output



- command: `python3 main.py`