



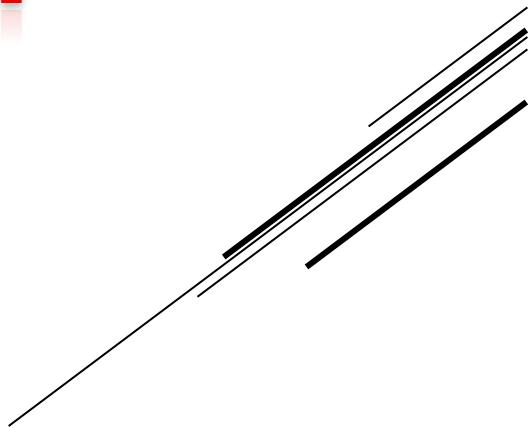
UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA



DIPARTIMENTO  
DI INGEGNERIA  
DELL'INFORMAZIONE



Developed by:  
chriSTIAN Marchiori  
FAbio Zanini





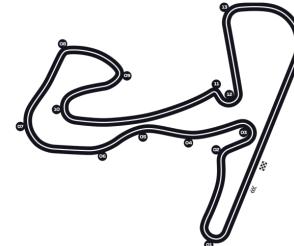
# Introduction

This project aims to create a graph database representing the world of Formula 1.

Formula 1 (a.k.a. F1 or Formula One) is the highest category of single-seat auto racing sanctioned by the Fédération Internationale de l'Automobile (FIA) and owned by the Formula One Group.

The datasets used to create this database contain information regarding races, drivers, driver ratings, constructors, qualifying, circuits, lap times, pit stops, and championships from 1950 until the latest 2023 season.

The project also consists on some SPARQL queries to show how the database works.





# Datasets



Formula 1 data:

<https://ergast.com/mrd/> or

<https://www.kaggle.com/datasets/rohanrao/formula-1-world-championship-1950-2020>

The dataset consists of all information on the Formula 1 races, drivers, constructors, qualifying, circuits, lap times, pit stops, championships from 1950 till the latest 2023 season.



Ratings data:

- ▶ 2023: <https://ratings-api.ea.com/v2/entities/f1-23-drivers-ratings>
- ▶ Older: <https://github.com/toUpperCase78/formula1-datasets>

Driver Ratings from EA & Codemasters F1 2021 / F1 22 / F1 23 Official Video Game



Country to Nationality csv:

<https://github.com/Imagin-io/country-nationality-list/tree/master>

Csv file that maps nationalities to countries.



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA



# Development process

Graph schema

Ontology

Serialization

Queries

Graph model  
designed to fit  
the F1 data

Developed with  
**arrows.app**

Ontology  
according to  
the Graph  
model

Developed with  
**Protégé**

Serialization  
process to  
produce turtle  
files

Developed with  
**Jupyter  
Notebook**

Queries writing to  
test the database

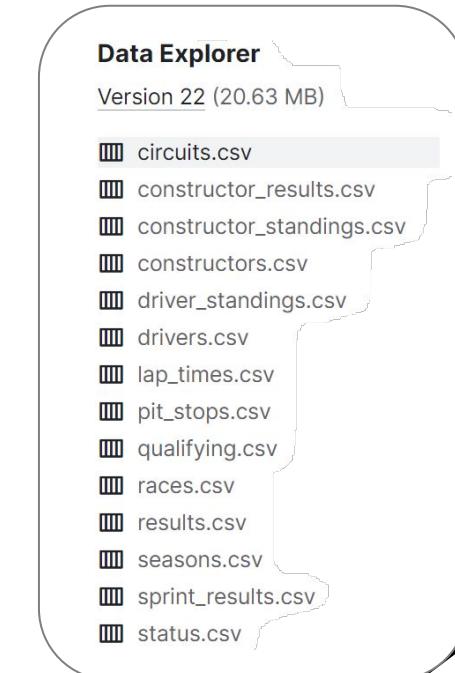
Developed with  
**SPARQL**



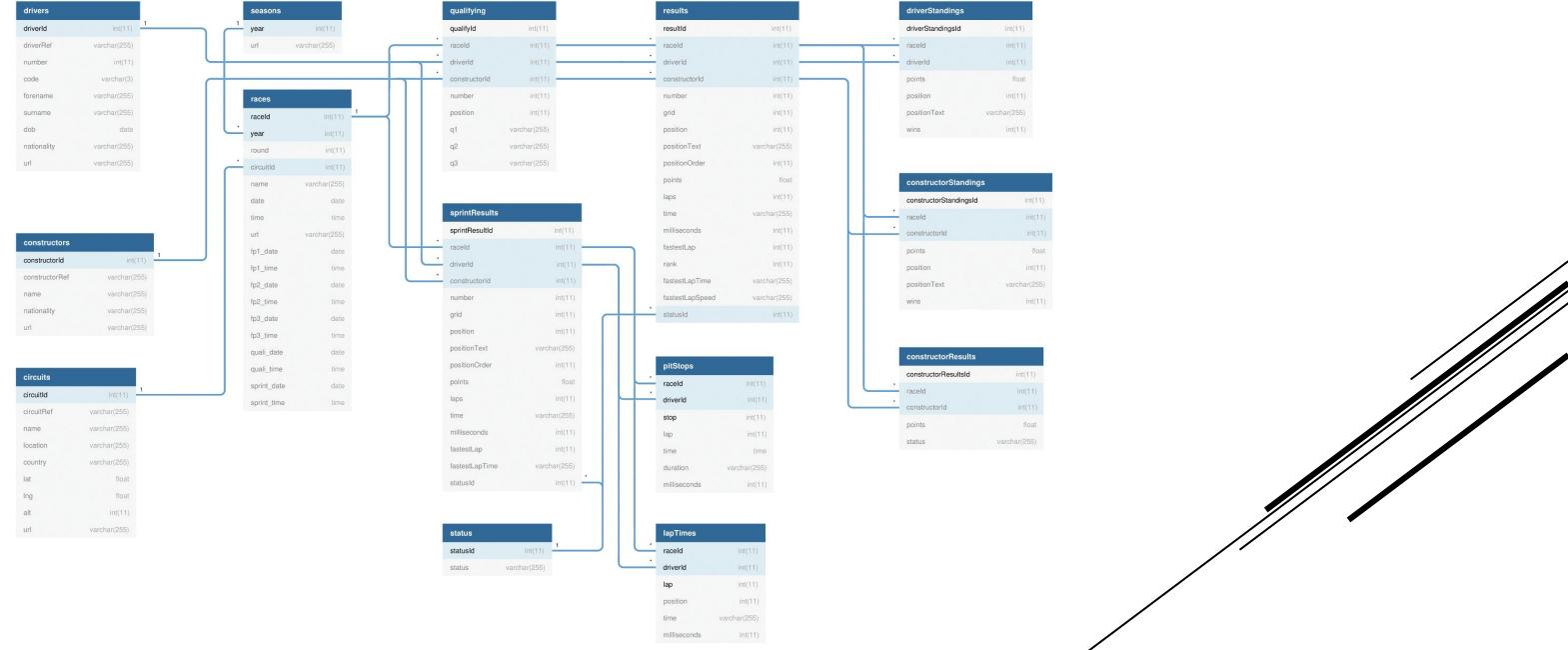
## Dataset

# CSV files list

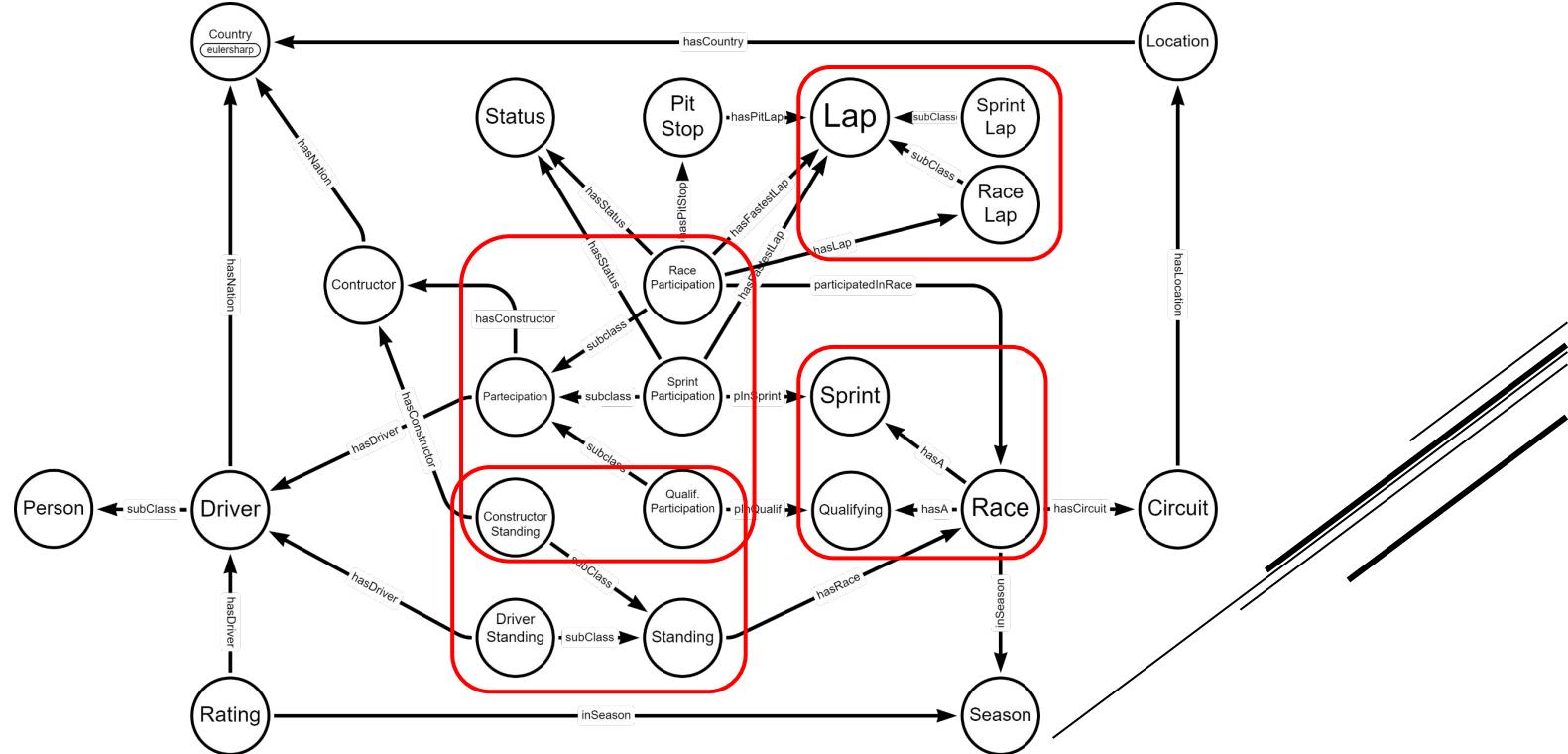
- **circuits.csv** Circuits where F1 races are held
- **constructor\_results.csv** Race results of the constructors' championship
- **constructor\_standings.csv** Final standings of the constructors' championship
- **constructors.csv** Constructor teams in F1
- **driver\_standings.csv** Final standings of the driver's championship
- **drivers.csv** Drivers in F1
- **lap\_times.csv** Lap times in F1
- **nationalities.csv** Map nationalities to countries
- **pit\_stops.csv** Pit stops in F1
- **qualifying.csv** Qualifying in F1
- **races.csv** Races in F1
- **ratings.csv** Ratings of F1 drivers
- **results.csv** Results of F1 races for each driver
- **seasons.csv** Seasons of F1
- **sprint\_results.csv** Results of F1 sprint races for each driver
- **status.csv** Various status related to drivers' results



# E-R Schema



# Graph Schema





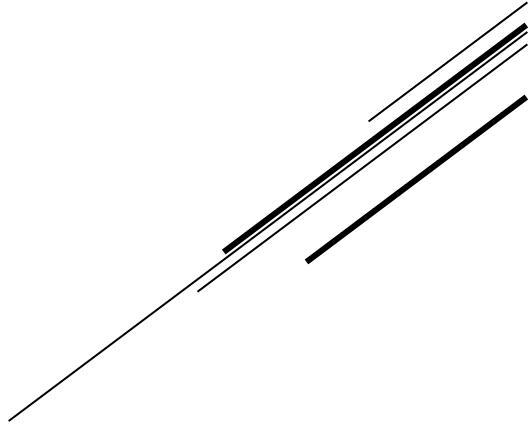
UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA

DIPARTIMENTO  
DI INGEGNERIA  
DELL'INFORMAZIONE



# Ontology

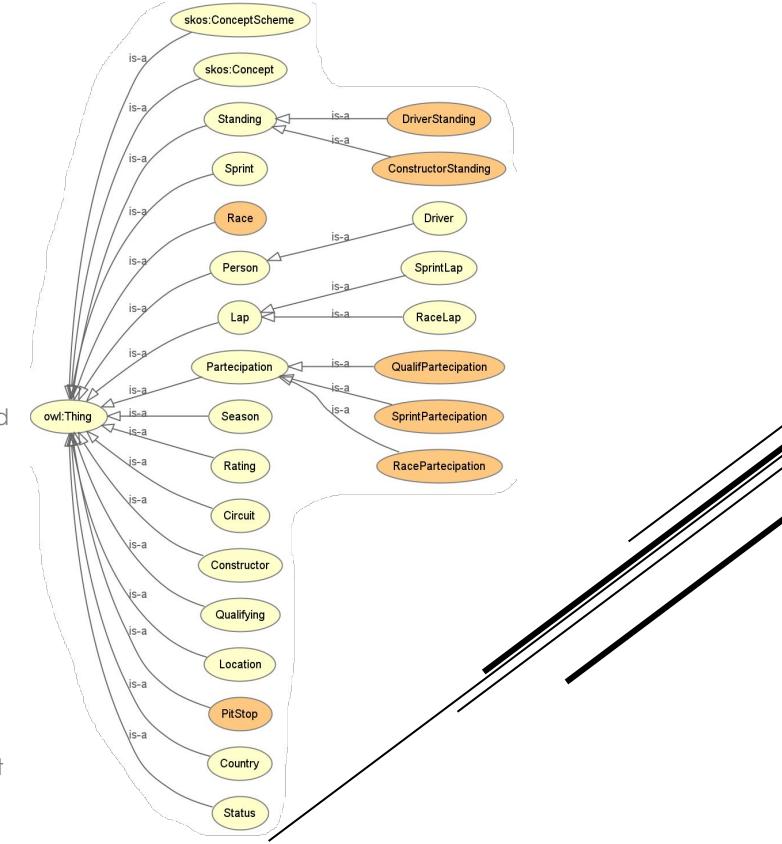
ontology



# Ontology

# Class hierarchy

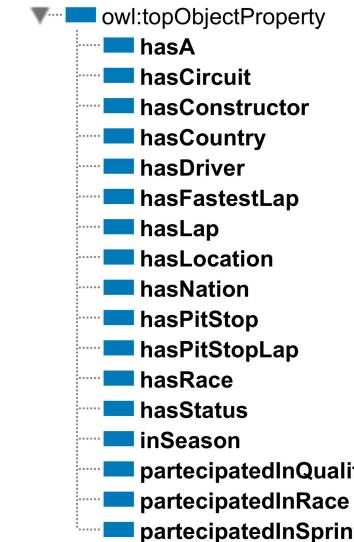
- **Standing**
    - **DriverStanding**
    - **ConstructorStanding**
  - **Sprint**
  - **Race**
  - **Person**
    - **Driver**
  - **Lap**
    - **SprintLap**
    - **RaceLap**
  - **Participation**
    - **QualifParticipation**
    - **SprintParticipation**
    - **RaceParticipation**
  - **Season**
  - **Rating**
  - **Circuit**
  - **Constructor**
  - **Qualifying**
  - **Location**
  - **PitStop**
  - **Country**
  - **Status**
- Represents championship standings  
...related to drivers  
...related to constructors
- Represents a sprint race related to a specific race
- Represents a specific F1 race
- Represents a person in general  
Represents a F1 driver
- Represents a single lap in a generic race  
Represents a single lap in a sprint  
Represents a single lap in a race
- Represents the participation of a driver and his related constructor team  
...to a qualifying  
...to a sprint  
...to a race
- Represents a F1 season
- Represents a rating tab for a driver
- Represents the circuits on which F1 races are done
- Represents F1 teams
- Represents a qualifying related to a particular race
- Represents the location/city of a circuit
- Represents a single pitstop in a F1 race
- Represents world countries
- Represents a status of a driver or a car in a race/sprint





# Ontology Object properties

- **hasA** (Race  $\mapsto$  Qualifying or Sprint)
- **hasCircuit** (Race  $\mapsto$  Circuit) **F**
- **hasConstructor** (ConstructorStanding or Participation  $\mapsto$  Constructor) **F**
- **hasCountry** (Location  $\mapsto$  Country) **F**
- **hasDriver** (DriverStanding or Participation or Rating  $\mapsto$  Driver) **F**
- **hasFastestLap** (RaceParticipation or SprintParticipation  $\mapsto$  Lap) **F**
- **hasLap** (RaceParticipation  $\mapsto$  Lap)
- **hasLocation** (Circuit  $\mapsto$  Location) **F**
- **hasNation** (Constructor or Driver  $\mapsto$  Country)
- **hasPitStop** (RaceParticipation  $\mapsto$  PitStop)
- **hasPitStopLap** (PitStop  $\mapsto$  Lap)**F**
- **hasRace** (Standing  $\mapsto$  Race) **F**
- **hasStatus** (RaceParticipation or SprintParticipation  $\mapsto$  Status) **F**
- **inSeason** (Race or Rating  $\mapsto$  Season) **F**
- **participatedInQualif** (QualifParticipation  $\mapsto$  Qualifying) **F**
- **participatedInRace** (RaceParticipation  $\mapsto$  Race) **F**
- **participatedInSprint** (SprintParticipation  $\mapsto$  Sprint) **F**





# Ontology Data properties

▼ owl:topDataProperty

- hasAlt
- hasAwareness
- hasBuyout
- hasCarNumber
- hasCircuitRef
- hasCode
- hasConstructorPoints
- hasConstructorRef
- hasContractCost
- hasDate
- hasDateOfBirth
- hasDriverNumber
- hasDriverRef
- hasDuration
- hasExperience
- hasFastestLapRank
- hasFastestLapSpeed
- hasFastestLapTime
- hasForename
- hasFp1Date
- hasFp1Time
- hasFp2Date
- hasFp2Time
- hasFp3Date
- hasFp3Time
- hasLapNumber
- hasLapPosition
- hasLaps
- hasLapTime

- hasLat
- hasLng
- hasMillisecondsResultTime
- hasMillisecondsTime
- hasName
- hasPace
- hasPeriod
- hasPitStopTimeOfDay
- hasPoints
- hasPosition
- hasPositionOrder
- hasPositionText
- hasQ1Time
- hasQ2Time
- hasQ3Time
- hasRaceCraft
- hasRating
- hasResultGap
- hasResultTime
- hasRound
- hasSalary
- hasStartingGridPosition
- hasStopNumber
- hasSurname
- hasTime
- hasTotalPoints
- hasTotalPosition
- hasTotalPositionText
- hasURL
- hasWins
- hasYear



# Ontology Data properties

## RACE PARTECIPATION:

| Property                  | Datatype    |
|---------------------------|-------------|
| hasConstructorPoints      | xsd:integer |
| hasFastestLapRank         | xsd:integer |
| hasFastestLapSpeed        | xsd:decimal |
| hasFastestLapTime         | xsd:time    |
| hasLaps                   | xsd:integer |
| hasMillisecondsResultTime | xsd:integer |
| hasPoints                 | xsd:integer |
| hasPositionOrder          | xsd:integer |
| hasPositionText           | xsd:string  |
| hasPositionGap            | xsd:time    |
| hasResultTime             | xsd:time    |
| hasStartingGridPosition   | xsd:integer |

## PARTECIPATION:

| Property     | Datatype    |
|--------------|-------------|
| hasCarNumber | xsd:integer |
| hasPosition  | xsd:integer |

## SPRINT PARTECIPATION:

| Property                  | Datatype    |
|---------------------------|-------------|
| hasFastestLap             | xsd:integer |
| hasFastestLapTime         | xsd:time    |
| hasLaps                   | xsd:integer |
| hasMillisecondsResultTime | xsd:integer |
| hasPoints                 | xsd:integer |
| hasPositionOrder          | xsd:integer |
| hasPositionText           | xsd:string  |
| hasResultGap              | xsd:time    |
| hasResultTime             | xsd:time    |
| hasStartingGridPosition   | xsd:integer |

## STANDING:

| Property              | Datatype    |
|-----------------------|-------------|
| hasTotalPoints        | xsd:integer |
| hasTotalPositionOrder | xsd:integer |
| hasTotalPositionText  | xsd:string  |
| hasWins               | xsd:integer |

## QUALIFYING PARTECIPATION: CIRCUIT:

| Property  | Datatype |
|-----------|----------|
| hasQ1Time | xsd:time |
| hasQ2Time | xsd:time |
| hasQ3Time | xsd:time |

## STATUS:

| Property | Datatype   |
|----------|------------|
| hasName  | xsd:string |

## QUALIFYING and SPRINT:

| Property | Datatype |
|----------|----------|
| hasDate  | xsd:date |
| hasTime  | xsd:time |

## SEASON:

| Property | Datatype   |
|----------|------------|
| hasURL   | xsd:string |
| hasYear  | xsd:gYear  |

## LAP:

| Property     | Datatype    |
|--------------|-------------|
| hasLapNumber | xsd:integer |
| hasLapTime   | xsd:time    |

## RACE LAP:

| Property            | Datatype    |
|---------------------|-------------|
| hasLapPosition      | xsd:integer |
| hasMillisecondsTime | xsd:integer |

## CONSTRUCTOR:

| Property          | Datatype   |
|-------------------|------------|
| hasConstructorRef | xsd:string |
| hasName           | xsd:string |
| hasUrl            | xsd:string |

## RACE:

| Property   | Datatype |
|------------|----------|
| hasDate    | xsd:date |
| hasTime    | xsd:time |
| hasFp1Date | xsd:date |
| hasFp2Date | xsd:date |
| hasFp3Date | xsd:date |
| hasFp1Time | xsd:time |
| hasFp2Time | xsd:time |
| hasFp3Time | xsd:time |

| Property        | Datatype    |
|-----------------|-------------|
| hasDriverNumber | xsd:integer |
| hasDriverRef    | xsd:string  |
| hasForename     | xsd:string  |
| hasSurname      | xsd:string  |
| hasURL          | xsd:string  |

## RATING:

| Property        | Datatype    |
|-----------------|-------------|
| hasAwareness    | xsd:integer |
| hasBuyout       | xsd:long    |
| hasContractCost | xsd:long    |
| hasExperience   | xsd:integer |
| hasPace         | xsd:integer |
| hasPeriod       | xsd:date    |
| hasRaceCraft    | xsd:integer |
| hasRating       | xsd:integer |
| hasSalary       | xsd:long    |

## PIT STOP:

| Property            | Datatype    |
|---------------------|-------------|
| hasDuration         | xsd:time    |
| hasMillisecondsTime | xsd:integer |
| hasPitStopTimeOfDay | xsd:time    |
| hasStopNumber       | xsd:integer |



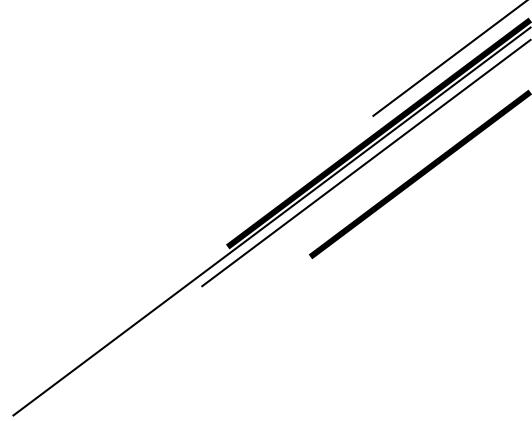
UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA

DIPARTIMENTO  
DI INGEGNERIA  
DELL'INFORMAZIONE



# Serialization

Serializzazione





# Serialization Introduction

Turtle files produced by the serialization process:

| Turtle files            | Csv files from which we extracted the data for the turtle files    |
|-------------------------|--------------------------------------------------------------------|
| circuits.ttl            | circuits.csv, countries.csv                                        |
| constructors.ttl        | constructors.csv, nationalities.csv                                |
| drivers.ttl             | drivers.csv, nationalities.csv                                     |
| laps1.ttl and laps2.ttl | lap_times.csv, results.csv, constructor_results.csv                |
| qualifying.ttl          | qualifying.csv, races.csv                                          |
| race.ttl                | races.csv                                                          |
| race_participation.ttl  | results.csv, constructor_results.csv, lap_times.csv                |
| ratings.ttl             | ratings.csv, drivers.csv                                           |
| sprint.ttl              | sprint_results.csv, races.csv                                      |
| standings.ttl           | driver_standings.csv, constructor_standings.csv                    |
| status_season.ttl       | status.csv, seasons.csv                                            |
| stops.ttl               | pit_stops.csv, results.csv, constructor_results.csv, lap_times.csv |

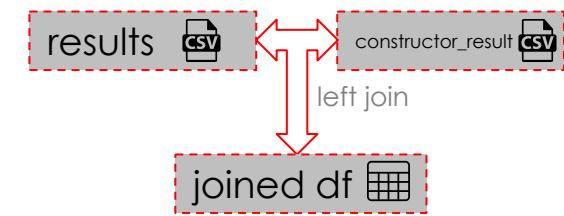


## Serialization

# Matching columns

### 1. Join

We joined *results.csv* and *constructor\_results.csv* in a single Pandas data frame. Then, all the results concerning the constructor team were matched with the driver results before managing the data frame.



### 2. Search

For each current row evaluated, we searched for a matching row in the second table in order to match some constraints, such as some keys.



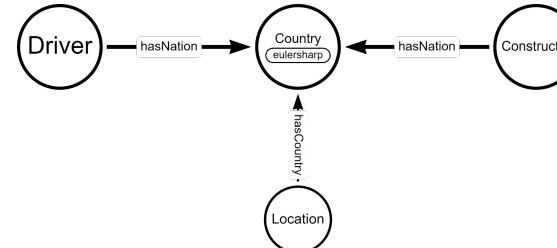
## Serialization

# Nationality to country mapping

We used this approach in `drivers.ttl` and `constructors.ttl` to maintain a single class `country` taken by an external source.

In this way, we avoided creating an additional nationality class and it was not necessary to link each individual in the nationality class to the country class via an object property.

The mapping was done directly in the serialization step.



| num_code | alpha_2_code | alpha_3_code | en_short_name       | nationality          |
|----------|--------------|--------------|---------------------|----------------------|
| 4        | AF           | AFG          | Afghanistan         | Afghan               |
| 248      | AX           | ALA          | Åland Islands       | Åland Island         |
| 8        | AL           | ALB          | Albania             | Albanian             |
| 12       | DZ           | DZA          | Algeria             | Algerian             |
| 16       | AS           | ASM          | American Samoa      | American Samoan      |
| 20       | AD           | AND          | Andorra             | Andorran             |
| 24       | AO           | AGO          | Angola              | Angolan              |
| 660      | AI           | AIA          | Anguilla            | Anguillan            |
| 10       | AQ           | ATA          | Antarctica          | Antarctic            |
| 28       | AG           | ATG          | Antigua and Barbuda | Antiguan or Barbudan |
| 32       | AR           | ARG          | Argentina           | Argentine            |

⋮

## Serialization

# Time format

We used 3 functions to manage time:

## 1. `time_formatter`

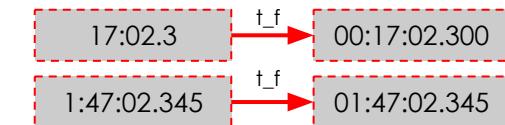
Function that transforms a time value to the standard %H:%M:%S.%f format, adds the zeros and the missing colon to the beginning and end of the string.

## 2. `gap_formatter`

Function that transforms a gap time value to the standard %H:%M:%S.%f format, adds the zeros and the missing colon to the beginning and end of the string.

## 3. `time_converter`

Function that calculates a driver's actual arrival time using the winner's arrival time and the time distance from it.



$$\begin{array}{l} \text{Driver gap time:} \\ + \\ \text{Winner arrival time:} \\ = \\ \text{Driver arrival time:} \end{array}$$

|              |
|--------------|
| +122.345     |
| 1:00:00.000  |
| 01:02:02.345 |

## Serialization

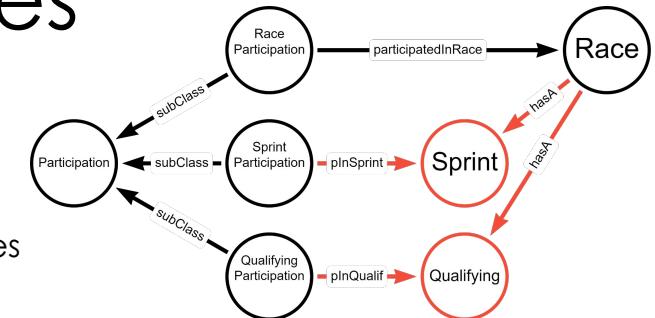
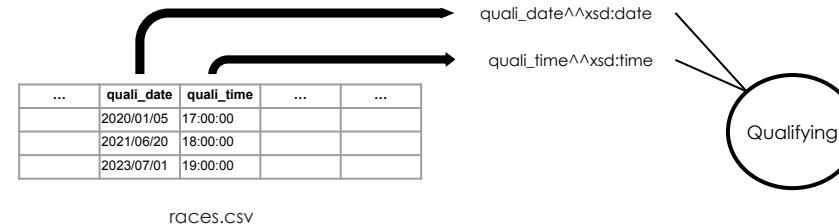
# Qualifying and Sprint classes

In the source dataset, we have two csv files concerning the races:

- the one for the generic race (`races.csv`) and
- the one for each driver's participation in the race (`results.csv`).

For qualifying and sprints this does not happen and we only have files representing the participations (`qualifying.csv`, `sprint_results.csv`).

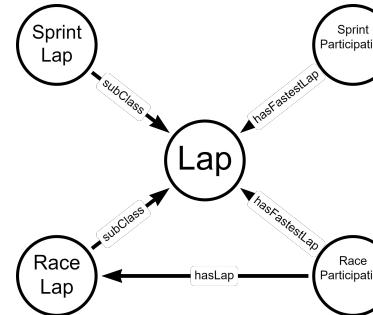
We decided to **add two classes**, Sprint and Qualifying, to represent the generic sprints and qualifications, not related to the driver.



## Serialization

# Race and sprint laps

In the source dataset, we have the *lap\_times.csv* file that contains all the laps of each driver in each race, with their positions and times.



Consequently, we decided to **create a subclass of Lap**, called SprintLap, to include the fastest laps related to sprints, with related time.

We also chose to keep this information in SprintParticipation so that it would be faster to extract it in SPARQL.



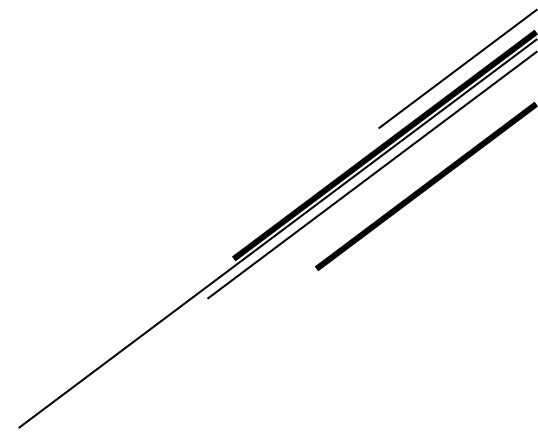
UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA

DIPARTIMENTO  
DI INGEGNERIA  
DELL'INFORMAZIONE



# Queries

docs100





## Query 8.1

# The 5 most winning drivers in F1 history, ordered by number of grand prix wins

```
PREFIX : <https://www.dei.unipd.it/db2/groupProject/FASTianF1#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?name (COUNT(?name) as ?numOfWins) WHERE {
    ?race :hasCircuit ?circuit;
    a :Race.
    ?racePart :partecipatedInRace ?race;
    :hasPositionOrder "1"^^xsd:integer;
    :hasDriver ?driver.
    ?driver a :Driver;
    :hasForename ?fname;
    :hasSurname ?surname.
    BIND(CONCAT(?fname, " ", ?surname) AS ?name).
} GROUP BY ?name
ORDER BY DESC (?numOfWins)
LIMIT 5
```

**SPARQL RESULT:**

|   | name                 | numOfWins          |
|---|----------------------|--------------------|
| 1 | "Lewis Hamilton"     | "103"^^xsd:integer |
| 2 | "Michael Schumacher" | "91"^^xsd:integer  |
| 3 | "Max Verstappen"     | "54"^^xsd:integer  |
| 4 | "Sebastian Vettel"   | "53"^^xsd:integer  |
| 5 | "Alain Prost"        | "51"^^xsd:integer  |

**WIKIPEDIA:**

| Rank | Country        | Driver              | Wins | Seasons active       | First win                | Last win                      |
|------|----------------|---------------------|------|----------------------|--------------------------|-------------------------------|
| 1    | United Kingdom | Lewis Hamilton†     | 103  | 2007–                | 2007 Canadian Grand Prix | 2021 Saudi Arabian Grand Prix |
| 2    | Germany        | Michael Schumacher‡ | 91   | 1991–2006, 2010–2012 | 1992 Belgian Grand Prix  | 2006 Chinese Grand Prix       |
| 3    | Netherlands    | Max Verstappen‡     | 54   | 2015–                | 2016 Spanish Grand Prix  | 2023 Abu Dhabi Grand Prix     |
| 4    | Germany        | Sebastian Vettel‡   | 53   | 2007–2022            | 2008 Italian Grand Prix  | 2019 Singapore Grand Prix     |
| 5    | France         | Alain Prost‡        | 51   | 1980–1991, 1993      | 1981 French Grand Prix   | 1993 German Grand Prix        |



## Query 11.3

## Data about fastest qualifying laps in F1 history (both q1, q2 and q3)

```
PREFIX : <https://www.dei.unipd.it/db2/groupProject/FASTianF1#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?driversName ?qTime ?circuitName ?raceName WHERE {
    { ?qualPart :hasQ3Time ?qTime.
    }
    UNION
    {
        ?qualPart :hasQ2Time ?qTime.
    }
    UNION
    {
        ?qualPart :hasQ1Time ?qTime.
    }
    ?qualPart :hasDriver ?driver;
        :participatedInQualif ?quali;
        a :QualifParticipation.
    ?quali a :Qualifying.
    ?race a :Race;
        :hasA ?quali;
        :hasName ?raceName;
        :hasCircuit ?circ.
    ?circ a :Circuit;
        :hasName ?circuitName.
    ?driver a :Driver;
        :hasForename ?fname;
        :hasSurname ?sname.
    BIND(CONCAT(?fname, " ", ?sname) AS ?driversName).
    FILTER(?qTime != "00:00:00"^^xsd:time)
}
ORDER BY (?qTime)
LIMIT 100
```

## SPARQL RESULT:

|    | driversName        | qTime                       | circuitName                     | raceName            |
|----|--------------------|-----------------------------|---------------------------------|---------------------|
| 1  | "Valtteri Bottas"  | "00:00:53.377000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 2  | "George Russell"   | "00:00:53.403000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 3  | "Max Verstappen"   | "00:00:53.433000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 4  | "Charles Leclerc"  | "00:00:53.613000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 5  | "Max Verstappen"   | "00:00:53.647000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 6  | "Sergio Pérez"     | "00:00:53.787000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 7  | "Sergio Pérez"     | "00:00:53.790000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 8  | "Valtteri Bottas"  | "00:00:53.803000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 9  | "Carlos Sainz"     | "00:00:53.818000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 10 | "George Russell"   | "00:00:53.819000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 11 | "Charles Leclerc"  | "00:00:53.825000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 12 | "Lance Stroll"     | "00:00:53.840000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 13 | "Daniil Kvyat"     | "00:00:53.856000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 14 | "Daniel Ricciardo" | "00:00:53.871000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 15 | "Valtteri Bottas"  | "00:00:53.904000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 16 | "Daniil Kvyat"     | "00:00:53.906000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 17 | "Pierre Gasly"     | "00:00:53.941000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 18 | "Daniel Ricciardo" | "00:00:53.957000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |
| 19 | "Esteban Ocon"     | "00:00:53.985000"^^xsd:time | "Bahrain International Circuit" | "Sakhir Grand Prix" |



## Query 12

# Data about 10 fastest pit stops in F1 history

```
PREFIX : <https://www.dei.unipd.it/db2/groupProject/FASTianF1#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?driversName ?pitDuration ?circuitName ?raceName ?raceDate WHERE {
    ?racePart :hasPitStop ?pit;
                :hasDriver ?driver;
                :partecipatedInRace ?race;
                a :RacePartecipation
    ?race a :Race;
           :hasName ?raceName;
           :hasDate ?raceDate;
           :hasCircuit ?circ
    ?circ a :Circuit;
           :hasName ?circuitName
    ?pit a :PitStop;
           :hasDuration ?pitDuration
    ?driver a :Driver;
           :hasForename ?fname;
           :hasSurname ?sname
    BIND(CONCAT(?fname, " ", ?sname) AS ?driversName)
}
ORDER BY (?pitDuration)
LIMIT 10
```

**SPARQL RESULT:**

|    | driversName          | pitDuration                | circuitName                      | raceName               | raceDate              |
|----|----------------------|----------------------------|----------------------------------|------------------------|-----------------------|
| 1  | "Pastor Maldonado"   | "00:00:12.897000""xsd:time | "Yas Marina Circuit"             | "Abu Dhabi Grand Prix" | "2011-11-13""xsd:date |
| 2  | "Bruno Senna"        | "00:00:12.959000""xsd:time | "Yas Marina Circuit"             | "Abu Dhabi Grand Prix" | "2011-11-13""xsd:date |
| 3  | "Lewis Hamilton"     | "00:00:13.173000""xsd:time | "Hungaroring"                    | "Hungarian Grand Prix" | "2011-07-31""xsd:date |
| 4  | "Pastor Maldonado"   | "00:00:13.186000""xsd:time | "Hungaroring"                    | "Hungarian Grand Prix" | "2011-07-31""xsd:date |
| 5  | "Michael Schumacher" | "00:00:13.199000""xsd:time | "Hungaroring"                    | "Hungarian Grand Prix" | "2012-07-29""xsd:date |
| 6  | "Sergio Pérez"       | "00:00:13.201000""xsd:time | "Hungaroring"                    | "Hungarian Grand Prix" | "2011-07-31""xsd:date |
| 7  | "Pastor Maldonado"   | "00:00:13.206000""xsd:time | "Hungaroring"                    | "Hungarian Grand Prix" | "2012-07-29""xsd:date |
| 8  | "Felipe Massa"       | "00:00:13.259000""xsd:time | "Circuit de Barcelona-Catalunya" | "Spanish Grand Prix"   | "2012-05-13""xsd:date |
| 9  | "Pastor Maldonado"   | "00:00:13.266000""xsd:time | "Circuit de Barcelona-Catalunya" | "Spanish Grand Prix"   | "2013-05-12""xsd:date |
| 10 | "Sebastian Vettel"   | "00:00:13.335000""xsd:time | "Circuit de Barcelona-Catalunya" | "Spanish Grand Prix"   | "2012-05-13""xsd:date |



## Query 14

# Constructors with most wins in the constructors' championship

```
PREFIX : <https://www.dei.unipd.it/db2/groupProject/FASTianF1#>
SELECT ?name (COUNT(?name) AS ?totalWins) (GROUP_CONCAT(DISTINCT ?outerYear ; separator=",") AS ?years) WHERE{
{
    SELECT ?name ?race (AVG(?pos) AS ?finalPosition) ?outerYear WHERE{
        ?stand :hasRace ?race ;
        :hasConstructor ?cons ;
        :hasTotalPosition ?pos
        ?race :hasRound ?outerRound ;
        :inSeason ?outerSeason
        ?outerSeason :hasYear ?outerYear.
        ?cons :hasName ?name.
        FILTER(?outerRound >= ?maxRound && ?outerYear = ?year && ?pos = 1)
    {
        SELECT ?year (MAX(?round) AS ?maxRound) WHERE {
            ?race :inSeason ?season ;
            :hasRound ?round
            ?season :hasYear ?year.
        }
        GROUP BY ?year
    }
    GROUP BY ?name ?race ?outerYear
    ORDER BY (?outerYear)
}
}
GROUP BY ?name
ORDER BY DESC (?totalWins)
```

## SPARQL RESULT:

| name            | totalWins         | years                                                                             |
|-----------------|-------------------|-----------------------------------------------------------------------------------|
| 'Ferrari'       | "16"^^xsd:integer | "1961,1964,1975,1976,1977,1979,1982,1983,1999,2000,2001,2002,2003,2004,2007,2008" |
| 'Williams'      | "9"^^xsd:integer  | "1980,1981,1986,1987,1992,1993,1994,1996,1997"                                    |
| 'McLaren'       | "9"^^xsd:integer  | "1974,1984,1985,1988,1989,1990,1991,1998"                                         |
| 'Mercedes'      | "8"^^xsd:integer  | "2014,2015,2016,2017,2018,2019,2020,2021"                                         |
| 'Red Bull'      | "6"^^xsd:integer  | "2010,2011,2012,2013,2022,2023"                                                   |
| 'Team Lotus'    | "4"^^xsd:integer  | "1970,1972,1973,1978"                                                             |
| 'Cooper-Climax' | "2"^^xsd:integer  | "1959,1960"                                                                       |
| 'Lotus-Climax'  | "2"^^xsd:integer  | "1963,1965"                                                                       |
| 'Brahham-Repco' | "2"^^xsd:integer  | "1966,1967"                                                                       |
| 'Renault'       | "2"^^xsd:integer  | "2005,2006"                                                                       |
| 'Vanwall'       | "1"^^xsd:integer  | "1958"                                                                            |
| 'BRM'           | "1"^^xsd:integer  | "1962"                                                                            |
| 'Lotus-Ford'    | "1"^^xsd:integer  | "1968"                                                                            |
| 'Matra-Ford'    | "1"^^xsd:integer  | "1969"                                                                            |
| 'Tyrrell'       | "1"^^xsd:integer  | "1971"                                                                            |
| 'Benetton'      | "1"^^xsd:integer  | "1995"                                                                            |
| 'Brawn'         | "1"^^xsd:integer  | "2009"                                                                            |

## WIKIPEDIA:

| Constructor | Titles | Seasons                                                                                        |
|-------------|--------|------------------------------------------------------------------------------------------------|
| Ferrari     | 16     | 1961, 1964, 1975, 1976, 1977, 1979, 1982, 1983, 1999, 2000, 2001, 2002, 2003, 2004, 2007, 2008 |
| Williams    | 9      | 1980, 1981, 1986, 1987, 1992, 1993, 1994, 1996, 1997                                           |
| McLaren     | 8      | 1974, 1984, 1985, 1988, 1989, 1990, 1991, 1998                                                 |
| Mercedes    |        | 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021                                                 |
| Lotus       | 7      | 1963, 1965, 1968, 1970, 1972, 1973, 1978                                                       |
| Red Bull    | 6      | 2010, 2011, 2012, 2013, 2022, 2023                                                             |
| Cooper      |        | 1959, 1960                                                                                     |
| Brabham     | 2      | 1966, 1967                                                                                     |
| Renault     |        | 2005, 2006                                                                                     |
| Vanwall     |        | 1958                                                                                           |
| BRM         |        | 1962                                                                                           |
| Matra       |        | 1969                                                                                           |
| Tyrrell     |        | 1971                                                                                           |
| Benetton    |        | 1995                                                                                           |
| Brawn       |        | 2009                                                                                           |

## Query 16

Are drivers who won the championship after 2000 older than those who won it before 2000, on average?

```

PREFIX : <https://www.dei.unipd.it/db2/groupProject/FASTianFl#>
ASK WHERE{
  {
    SELECT (AVG(?age)as ?avgOldAge) WHERE{
      ?stand :hasRace ?race ;
      :hasDriver ?driv ;
      :hasTotalPosition ?pos
      ?driv :hasDateOfBirth ?dob.
      ?race :hasDate ?raceDate ;
      :inSeason ?outerSeason ;
      :hasRound ?outerRound
      ?outerSeason :hasYear ?outerYear.
      BIND((YEAR(?raceDate)- YEAR(?dob)) AS ?age)
      FILTER(?outerRound = ?maxRound && ?outerYear = ?year && ?pos = 1)
    {
      SELECT ?year (MAX(?round) AS ?maxRound) WHERE {
        ?race :inSeason ?season ;
        :hasRound ?round
        ?season :hasYear ?year.
        FILTER(?year >= 2000)
      }
      GROUP BY ?year
    }
    GROUP BY ?year
  }
}
  {
    SELECT (AVG(?age)as ?avgRecentAge) WHERE{
      ?stand :hasRace ?race ;
      :hasDriver ?driv ;
      :hasTotalPosition ?pos
      ?driv :hasDateOfBirth ?dob.
      ?race :hasDate ?raceDate ;
      :inSeason ?outerSeason ;
      :hasRound ?outerRound
      ?outerSeason :hasYear ?outerYear.
      BIND((YEAR(?raceDate)- YEAR(?dob)) AS ?age)
      FILTER(?outerRound = ?maxRound && ?outerYear = ?year && ?pos = 1)
    {
      SELECT ?year (MAX(?round) AS ?maxRound) WHERE {
        ?race :inSeason ?season ;
        :hasRound ?round
        ?season :hasYear ?year.
        FILTER(?year >= 2000)
      }
      GROUP BY ?year
    }
    FILTER(?avgRecentAge > ?avgOldAge)
  }
}
}
  
```

**SPARQL RESULT:**

NO

Before 2000 avg age:

33.0

After 2000 avg age:

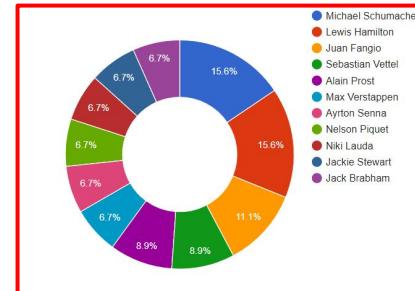
28.8



## Query 19

# Championship winning drivers who won more than 2 championships, ordered by number of championships won

```
PREFIX : <https://www.dei.unipd.it/db2/groupProject/FASTianFl#>
SELECT ?name (COUNT(?name) AS ?numOfWins) (GROUP_CONCAT(DISTINCT ?outerYear ; separator=",") AS ?years) WHERE{
    ?stand :hasRace ?race ;
        :hasDriver ?driv ;
        :hasTotalPosition ?pos.
    ?driv a :Driver;
        :hasForename ?fname;
        :hasSurname ?surname
    BIND(CONCAT(?fname, " ", ?surname) AS ?name)
    ?race :hasDate ?raceDate ;
        :inSeason ?outerSeason ;
        :hasRound ?outerRound.
    ?outerSeason :hasYear ?outerYear.
    FILTER(?outerRound = ?maxRound && ?outerYear = ?year && ?pos = 1)
    {
        SELECT ?year (MAX(?round) AS ?maxRound) WHERE {
            ?race :inSeason ?season ;
                :hasRound ?round
                ?season :hasYear ?year.
        }
        GROUP BY ?year
        ORDER BY ?year
    }
}
GROUP BY ?name
HAVING (?numOfWins > 2)
ORDER BY DESC (?numOfWins)
```



## SPARQL RESULT:

|    | name                 | numOfWins         | years                                |
|----|----------------------|-------------------|--------------------------------------|
| 1  | "Michael Schumacher" | "7***xsd:integer" | "1994,1995,2000,2001,2002,2003,2004" |
| 2  | "Lewis Hamilton"     | "5***xsd:integer" | "2008,2014,2015,2017,2018,2019,2020" |
| 3  | "Juan Fangio"        | "5***xsd:integer" | "1951,1954,1955,1956,1957"           |
| 4  | "Alain Prost"        | "4***xsd:integer" | "1985,1986,1989,1993"                |
| 5  | "Sebastian Vettel"   | "4***xsd:integer" | "2010,2011,2012,2013"                |
| 6  | "Jack Brabham"       | "3***xsd:integer" | "1959,1960,1966"                     |
| 7  | "Jackie Stewart"     | "3***xsd:integer" | "1969,1971,1973"                     |
| 8  | "Niki Lauda"         | "3***xsd:integer" | "1975,1977,1984"                     |
| 9  | "Nelson Piquet"      | "3***xsd:integer" | "1981,1983,1987"                     |
| 10 | "Ayrton Senna"       | "3***xsd:integer" | "1988,1990,1991"                     |
| 11 | "Max Verstappen"     | "3***xsd:integer" | "2021,2022,2023"                     |

## WIKIPEDIA:

| Driver             | Titles | Season(s)                                |
|--------------------|--------|------------------------------------------|
| Michael Schumacher | 7      | 1994, 1995, 2000, 2001, 2002, 2003, 2004 |
| Lewis Hamilton     | 5      | 2008, 2014, 2015, 2017, 2018, 2019, 2020 |
| Juan Manuel Fangio | 5      | 1951, 1954, 1955, 1956, 1957             |
| Alain Prost        | 4      | 1985, 1986, 1989, 1993                   |
| Sebastian Vettel   | 4      | 2010, 2011, 2012, 2013                   |
| Jack Brabham       | 3      | 1959, 1960, 1966                         |
| Jackie Stewart     | 3      | 1969, 1971, 1973                         |
| Niki Lauda         | 3      | 1975, 1977, 1984                         |
| Nelson Piquet      | 3      | 1981, 1983, 1987                         |
| Ayrton Senna       | 3      | 1988, 1990, 1991                         |
| Max Verstappen     | 3      | 2021, 2022, 2023                         |



## Query 22

# Winners of drivers' championship and constructors' championship for each season.

```
PREFIX : <https://www.dei.unipd.it/db2/groupProject/FASTianF1#>
SELECT ?outerYear ?driverName ?constructorName WHERE{
    ?driverStand :hasTotalPosition ?pos ;
        :hasRace ?race ;
        :hasDriver ?driver .
    ?constStand :hasTotalPosition ?pos ;
        :hasRace ?race ;
        :hasConstructor ?const .
    ?race :hasRound ?outerRound ;
        :inSeason ?season .
    ?season :hasYear ?outerYear .
    ?driver :hasForename ?fname ;
        :hasSurname ?sname .
    ?const :hasName ?constructorName .
    BIND(CONCAT(?fname, " ", ?sname) AS ?driverName).
    FILTER(?outerRound = ?maxRound && ?outerYear = ?year && ?pos = 1)
    {
        SELECT ?year (MAX(?round) AS ?maxRound) WHERE {
            ?race :inSeason ?season ;
                :hasRound ?round .
            ?season :hasYear ?year .
        }
        GROUP BY ?year
    }
}
ORDER BY DESC (?outerYear)
```

**SPARQL RESULT:**

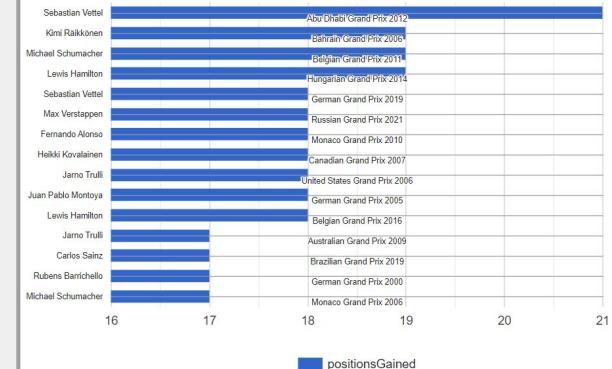
|    | outerYear           | driverName         | constructorName |
|----|---------------------|--------------------|-----------------|
| 1  | "2023"^^xsd:integer | 'Max Verstappen'   | 'Red Bull'      |
| 2  | "2022"^^xsd:integer | 'Max Verstappen'   | 'Red Bull'      |
| 3  | "2021"^^xsd:integer | 'Max Verstappen'   | 'Mercedes'      |
| 4  | "2020"^^xsd:integer | 'Lewis Hamilton'   | 'Mercedes'      |
| 5  | "2019"^^xsd:integer | 'Lewis Hamilton'   | 'Mercedes'      |
| 6  | "2018"^^xsd:integer | 'Lewis Hamilton'   | 'Mercedes'      |
| 7  | "2017"^^xsd:integer | 'Lewis Hamilton'   | 'Mercedes'      |
| 8  | "2016"^^xsd:integer | 'Nico Rosberg'     | 'Mercedes'      |
| 9  | "2015"^^xsd:integer | 'Lewis Hamilton'   | 'Mercedes'      |
| 10 | "2014"^^xsd:integer | 'Lewis Hamilton'   | 'Mercedes'      |
| 11 | "2013"^^xsd:integer | 'Sebastian Vettel' | 'Red Bull'      |

## Query 23

# Drivers who gained the most positions during a race after 2000

```
PREFIX : <https://www.dei.unipd.it/db2/groupProject/FASTianF1#>
SELECT ?driverName ?raceSeason ?positionsGained WHERE {
    ?part :hasStartingGridPosition ?grid ;
        :participatedInRace ?race ;
        :hasDriver ?driver ;
        :hasPosition ?pos .
    ?driver :hasForename ?fname ;
        :hasSurname ?lname .
    ?race :inSeason ?season ;
        :hasName ?raceName .
    ?season :hasYear ?year .
    BIND(CONCAT(?fname, " ", ?lname) AS ?driverName)
    BIND((?grid - ?pos) AS ?positionsGained)
    FILTER(?year >= 2000)
    BIND(CONCAT(?raceName, " ", STR(?year)) AS ?raceSeason)
}
ORDER BY DESC (?positionsGained)
LIMIT 15
```

**BAR CHART:**



**SPARQL RESULT:**

|    | driverName           | raceSeason                      | positionsGained  |
|----|----------------------|---------------------------------|------------------|
| 1  | "Sebastian Vettel"   | 'Abu Dhabi Grand Prix 2012'     | 21***xsd integer |
| 2  | "Kimi Räikkönen"     | 'Bahrain Grand Prix 2006'       | 19***xsd integer |
| 3  | "Michael Schumacher" | 'Belgian Grand Prix 2011'       | 19***xsd integer |
| 4  | "Lewis Hamilton"     | 'Hungarian Grand Prix 2014'     | 19***xsd integer |
| 5  | "Sebastian Vettel"   | 'German Grand Prix 2019'        | 18***xsd integer |
| 6  | "Max Verstappen"     | 'Russian Grand Prix 2021'       | 18***xsd integer |
| 7  | "Fernando Alonso"    | 'Monaco Grand Prix 2010'        | 18***xsd integer |
| 8  | "Heikki Kovalainen"  | 'Canadian Grand Prix 2007'      | 18***xsd integer |
| 9  | "Jarno Trulli"       | 'United States Grand Prix 2006' | 18***xsd integer |
| 10 | "Juan Pablo Montoya" | 'German Grand Prix 2005'        | 18***xsd integer |
| 11 | "Lewis Hamilton"     | 'Belgian Grand Prix 2016'       | 18***xsd integer |
| 12 | "Jarno Trulli"       | 'Australian Grand Prix 2009'    | 17***xsd integer |
| 13 | "Carlos Sainz"       | 'Brazilian Grand Prix 2019'     | 17***xsd integer |
| 14 | "Rubens Barrichello" | 'German Grand Prix 2000'        | 17***xsd integer |
| 15 | "Michael Schumacher" | 'Monaco Grand Prix 2006'        | 17***xsd integer |

## Query 24

Drivers ordered by pole-to-win percentage  
(percentage of races won starting from the first position in grid)

```

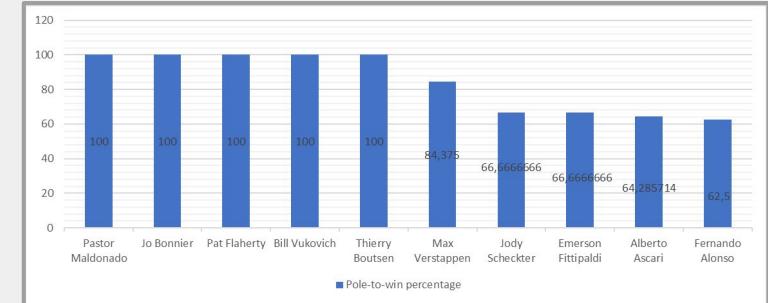
PREFIX : <https://www.dei.unipd.it/db2/groupProject/FASTianF1#>
SELECT ?name ?numOfPoleWins ?numOfPoles ?winPercentage WHERE{
    {
        SELECT ?name (COUNT(?name) AS ?numOfPoleWins) WHERE{
            ?racePart :participatedInRace ?race ;
                :hasDriver ?driv ;
                :hasStartingGridPosition ?startPos ;
                :hasPosition ?pos .
            ?driv a :Driver;
                :hasForename ?fname;
                :hasSurname ?surname.
            BIND(CONCAT(?fname, " ", ?surname) AS ?name).
            FILTER(?startPos = 1 && ?pos = 1)
        }
        GROUP BY ?name
    }
    {
        SELECT ?name (COUNT(?name) AS ?numOfPoles) WHERE{
            ?racePart :participatedInRace ?race ;
                :hasDriver ?driv ;
                :hasStartingGridPosition ?startPos .
            ?driv a :Driver;
                :hasForename ?fname;
                :hasSurname ?surname.
            BIND(CONCAT(?fname, " ", ?surname) AS ?name).
            FILTER(?startPos = 1)
        }
        GROUP BY ?name
    }
    BIND((?numOfPoleWins / ?numOfPoles) * 100 AS ?winPercentage)
}
ORDER BY DESC (?winPercentage)

```

### SPARQL RESULT:

| name                 | numOfPoleWins | numOfPoles | winPercentage                     |
|----------------------|---------------|------------|-----------------------------------|
| "Pastor Maldonado"   | "1"           | "100"      | "100"                             |
| "Jo Bonnier"         | "1"           | "100"      | "100"                             |
| "Pat Flaherty"       | "1"           | "100"      | "100"                             |
| "Bill Vukovich"      | "1"           | "100"      | "100"                             |
| "Thierry Boutsen"    | "1"           | "100"      | "100"                             |
| "Max Verstappen"     | "27"          | "32"       | "84.37500000000000"               |
| "Jody Scheckter"     | "2"           | "3"        | "66.6666666666666666666666666700" |
| "Emerson Fittipaldi" | "4"           | "6"        | "66.6666666666666666666666666700" |
| "Alberto Ascari"     | "9"           | "14"       | "64.285714285714285700"           |
| "Fernando Alonso"    | "14"          | "22"       | "63.636363636363636400"           |

### BAR CHART:





## Query 25

Drivers who finished first in a race and were placed in the last quarter in the standing at the previous race

```
PREFIX : <https://www.dei.unipd.it/db2/groupProject/FASTianF1#>
SELECT ?driverName ?raceName ?round ?year WHERE{
    ?race :hasRound ?round ;
        :inSeason ?season ;
        :hasName ?raceName.
    ?part :participatedInRace ?race ;
        :hasPosition ?pos ;
        :hasDriver ?driver.
    ?driver :hasForename ?fname ;
        :hasSurname ?lname.
    ?season :hasYear ?year.
    BIND((?round+1) AS ?prevRound)
    BIND(CONCAT(?fname," ",?lname) AS ?driverName)
    FILTER(?pos = 1 && EXISTS{
        ?prevRace :hasRound ?prevRound ;
            :inSeason ?season
        ?stand :hasRace ?prevRace ;
            :hasDriver ?driver ;
            :hasTotalPosition ?totPos
        FILTER (?totPos>= ((?maxPos/4)*3) && ?prevRace = ?raceMax)
        {
            SELECT ?raceMax (MAX(?totPos) AS ?maxPos) WHERE {
                ?standing a :DriverStanding ;
                    :hasRace ?raceMax ;
                    :hasTotalPosition ?totPos
            }
            GROUP BY ?raceMax
        }
    })
    ORDER BY (?year)
```

## SPARQL RESULT:

|    | driverName              | raceName                   | round            | year                |
|----|-------------------------|----------------------------|------------------|---------------------|
| 1  | "Alberto Ascari"        | "Belgian Grand Prix"       | "3"^^xsd:integer | "1952"^^xsd:integer |
| 2  | "Jack Brabham"          | "Dutch Grand Prix"         | "4"^^xsd:integer | "1960"^^xsd:integer |
| 3  | "Dan Gurney"            | "French Grand Prix"        | "4"^^xsd:integer | "1962"^^xsd:integer |
| 4  | "John Surtees"          | "Belgian Grand Prix"       | "2"^^xsd:integer | "1966"^^xsd:integer |
| 5  | "Ludovico Scarfiotti"   | "Italian Grand Prix"       | "7"^^xsd:integer | "1966"^^xsd:integer |
| 6  | "Dan Gurney"            | "Belgian Grand Prix"       | "4"^^xsd:integer | "1967"^^xsd:integer |
| 7  | "Jean-Pierre Beltoise"  | "Monaco Grand Prix"        | "4"^^xsd:integer | "1972"^^xsd:integer |
| 8  | "Jean-Pierre Jabouille" | "French Grand Prix"        | "8"^^xsd:integer | "1979"^^xsd:integer |
| 9  | "René Arnoux"           | "Brazilian Grand Prix"     | "2"^^xsd:integer | "1980"^^xsd:integer |
| 10 | "Kimi Räikkönen"        | "Malaysian Grand Prix"     | "2"^^xsd:integer | "2008"^^xsd:integer |
| 11 | "Sebastian Vettel"      | "Chinese Grand Prix"       | "3"^^xsd:integer | "2009"^^xsd:integer |
| 12 | "Lewis Hamilton"        | "Malaysian Grand Prix"     | "2"^^xsd:integer | "2014"^^xsd:integer |
| 13 | "Max Verstappen"        | "Saudi Arabian Grand Prix" | "2"^^xsd:integer | "2022"^^xsd:integer |



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA

DIPARTIMENTO  
DI INGEGNERIA  
DELL'INFORMAZIONE

FASTlan™

Thank you!

