* Each GP names are abbreviated as following below:

| BH | bahrain |
| --- | --- |
| SA | Saudi arabia |
| AU | australia |
| ER | Emilia romagna (imola) |
| MA | miami |
| ES | spain |
| MC | monaco |
| AZ | azerbaijan |
| CA | canada |
| GB | Great britain |
| AT | austria |
| FR | france |
| HU | hungary |
| BE | belgium |
| NL | netherland |
| IT | italy |
| SG | singapore |
| JP | japan |
| TX | texas |
| MX | mexico |
| BR | brazil |
| AE | Abu dhabi |

* There are five sessions: free practice 1, 2, and 3, qualifying, and race, and each of them are abbreviated as following below:

Free Practice 1: FP1, Free Practice 2: FP2, Free Practice 3: FP3, Qualifying: Q, Race: R

<https://theoehrly.github.io/Fast-F1/>

* There are two main api functions that we used: timing\_data and car\_data.
  + Timing data for each driver of the session of a race is broken down into two: timing\_data\_RaceName\_(0 or 1)\_Session\_CarNumber.
  + Timing data 0 (pandas dataframe) contains the data following:

'Unnamed: 0': maybe the number of laps? We can ignore this data.

'Time': the time each lap was completed, <str>

'Driver': car number for each driver, <str>

'LapTime': lap time of each lap, <str>

'NumberOfLaps': number of laps done in the end of this lap, <numpy.int64>

'NumberOfPitStops': number of pit stops done by the end of this lap, <numpy.int64>

'PitOutTime': session time when the driver exited the pit, <str>

'PitInTime': session time when the driver entered the pit, <str>

'Sector1Time': sector 1 time, <str>

'Sector2Time': sector 2 time, <str>

'Sector3Time': sector 3 time, <str>

'Sector1SessionTime’: Session time when the sector time was completed, <str>

'Sector2SessionTime': Session time when the sector time was completed, <str>

'Sector3SessionTime': Session time when the sector time was completed, <str>

'SpeedI1': speed trap in sector 1, < numpy.float64>

'SpeedI2': speed trap in sector 2, < numpy.float64>

'SpeedFL': speed trap speed, < numpy.float64>

'SpeedST': speed trap in the longest straight, < numpy.float64>

'IsPersonalBest': show if the lap was the personal best or not, <bool>

* + Timing data 1 (pandas dataframe) contains the data following:

'Unnamed: 0': maybe the number of laps? We can ignore this data.

'Time': the time when the lap was completed, <str>

'Driver': car number of the driver, <numpy.int64>

'Position': the position in the race for the lap, <numpy.int64>

'GapToLeader': time gap to leader in seconds, <numpy.int64>

'IntervalToPositionAhead': time gap to the car ahead, <numpy.int64>

* + Car data (pandas dataframe) contains the data following:
  + The data stream has a sample rate of 240ms.

'Unnamed: 0': maybe the number of laps? We can ignore this data.

'Time': maybe the number of laps? We can ignore this data.

'Date': timestamp for this sample as Date + Time; more or less exact, <str>

'RPM': the engine rev, <numpy.int64>

'Speed': speed at the time, <numpy.int64>

'nGear': the gear at which the car is running at the time, <numpy.int64>

'Throttle': the throttle opening at the time, <numpy.int64>

'Brake': if the car is braking or not, <bool>

'DRS': if the car is using the drag reduction system, <numpy.int64>

'Source': we can ignore this

['Unnamed: 0', 'Time', 'Driver', 'LapTime', 'NumberOfLaps',

'NumberOfPitStops', 'PitOutTime', 'PitInTime', 'Sector1Time',

'Sector2Time', 'Sector3Time', 'Sector1SessionTime',

'Sector2SessionTime', 'Sector3SessionTime', 'SpeedI1', 'SpeedI2',

'SpeedFL', 'SpeedST', 'IsPersonalBest']

label\_according\_to\_team = ["#00D2BE", "#00D2BE", "#0600EF", "#0600EF",

"#DC0000", "#DC0000", "#FF8700", "#FF8700",

"#0090FF", "#0090FF", "#2B4562", "#2B4562",

"#006F62", "#006F62", "#900000", "#900000",

"#005AFF", "#005AFF", "#FFFFFF", "#FFFFFF"]

label\_according\_to\_PU = ["#00D2BE", "#00D2BE", "#0600EF", "#0600EF",

"#DC0000", "#DC0000", "#00D2BE", "#00D2BE",

"#0090FF", "#0090FF", "#0600EF", "#0600EF",

"#00D2BE", "#00D2BE", "#DC0000", "#DC0000",

"#00D2BE", "#00D2BE", "#DC0000", "#DC0000"]