Package: openf1r (via r-universe)

May 17, 2026

Type Package

Title Retrieve Formula 1 Race Data from the OpenF1 API

Version 0.1.1

Maintainer Mike Cheng <mikefc@coolbutuseless.com>

URL https://github.com/coolbutuseless/openf1r

BugReports https://github.com/coolbutuseless/openf1r/issues

Description An interface to the OpenF1 API to retrieve Formula 1 race data such as race timings, lap speeds, driver information.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

Repository https://coolbutuseless.r-universe.dev

Date/Publication 2025-04-24 10:25:44 UTC

RemoteUrl https://github.com/coolbutuseless/openf1r

RemoteRef HEAD

RemoteSha 98a57e174caad127b81f001a65b225c54ff8f70d

Contents

`clear_cache` *Clear cached downloads*

Description

All data fetched from the OpenF1 API is cached, and subsequent calls with the same parameters just returns the cached value. Use this function to clear this cache so that the next download fetches data from the API

Usage

```
clear_cache()
```

Value

None.

Examples

```
clear_cache()
```

`f1_cars` *Some data about each car, at a sample rate of about 3.7 Hz.*

Description

Some data about each car, at a sample rate of about 3.7 Hz.

Usage

```
f1_cars(
  ...,
  session_key = "latest",
  driver_number,
  brake,
  drs,
  n_gear,
  rpm,
  speed,
  throttle,
```

```
  date,
  meeting_key
)
```

Arguments

...	Verbatim options added to API call. Use this to add filtering options e.g. `f1_sessions("date_start >= 2025-04-01")`. For filtering • most named arguments can be filtered • use <=, >=, <, > for specifying ranges • for date filtering, use date format "YYYY-MM-DD", and "YYYY-MM-DDTHH:MM:SS"
session_key	The unique identifier for the session. For `f1_sessions()` the default is `NULL` to return all sessions. For all other functions `session_key` defaults to 'latest' so as to only fetch the data for the latest/current session. Valid values are: 'latest', `NULL` or an integer session key
driver_number	The unique number assigned to an F1 driver
brake	Whether the brake pedal is pressed (100) or not (0).
drs	The Drag Reduction System (DRS) status. Possible codes and meanings: 0,1 = DRS off. 2,3,9 = Unknown. 8 = Detected, eligible once in activation zone. 10,12,14 = DRS on.
n_gear	Current gear selection, ranging from 1 to 8. 0 indicates neutral or no gear engaged.
rpm	Revolutions per minute of the engine.
speed	Velocity of the car in km/h.
throttle	Percentage of maximum engine power being used.
date	The UTC date and time, in ISO 8601 format.
meeting_key	The unique identifier for the meeting. Use 'latest' to identify the latest or current meeting.

Value

data.frame

Examples

```
f1_cars(driver_number = 44)
```

f1_drivers *Provides information about drivers for each session.*

Description

Provides information about drivers for each session.

Usage

```
f1_drivers(..., session_key = "latest", meeting_key)
```

Arguments

`...`	Verbatim options added to API call. Use this to add filtering options e.g. `f1_sessions("date_start >= 2025-04-01")`. For filtering • most named arguments can be filtered • use <=, >=, <, > for specifying ranges • for date filtering, use date format "YYYY-MM-DD", and "YYYY-MM-DDTHH:MM:SS"
`session_key`	The unique identifier for the session. For `f1_sessions()` the default is `NULL` to return all sessions. For all other functions `session_key` defaults to 'latest' so as to only fetch the data for the latest/current session. Valid values are: 'latest', `NULL` or an integer session key
`meeting_key`	The unique identifier for the meeting. Use 'latest' to identify the latest or current meeting.

Value

data.frame

Examples

```
f1_drivers()
```

f1_intervals *Fetches real-time interval data between drivers and their gap to the race leader. Available during races only, with updates approximately every 4 seconds.*

Description

Fetches real-time interval data between drivers and their gap to the race leader. Available during races only, with updates approximately every 4 seconds.

Usage

```
f1_intervals(
  ...,
  session_key = "latest",
  driver_number,
  gap_to_leader,
  interval,
  date,
  meeting_key
)
```

Arguments

`...`	Verbatim options added to API call. Use this to add filtering options e.g. `f1_sessions("date_start >= 2025-04-01")`. For filtering • most named arguments can be filtered • use <=, >=, <, > for specifying ranges • for date filtering, use date format "YYYY-MM-DD", and "YYYY-MM-DDTHH:MM:SS"
`session_key`	The unique identifier for the session. For `f1_sessions()` the default is `NULL` to return all sessions. For all other functions `session_key` defaults to 'latest' so as to only fetch the data for the latest/current session. Valid values are: 'latest', `NULL` or an integer session key
`driver_number`	The unique number assigned to an F1 driver
`gap_to_leader`	The time gap to the race leader in seconds, +1 LAP if lapped, or null for the race leader.
`interval`	The time gap to the car ahead in seconds, +1 LAP if lapped, or null for the race leader.
`date`	The UTC date and time, in ISO 8601 format.
`meeting_key`	The unique identifier for the meeting. Use 'latest' to identify the latest or current meeting.

Value

data.frame

Examples

```
f1_intervals()
```

f1_laps *Provides detailed information about individual laps.*

Description

Provides detailed information about individual laps.

Usage

```
f1_laps(
  ...,
  session_key = "latest",
  driver_number,
  duration_sector_1,
  duration_sector_2,
  duration_sector_3,
  i1_speed,
  i2_speed,
  is_pit_out_lap,
  lap_duration,
  lap_number,
  segments_sector_1,
  segments_sector_2,
  segments_sector_3,
  st_speed,
  date_start,
  meeting_key
)
```

Arguments

`...`	Verbatim options added to API call. Use this to add filtering options e.g. `f1_sessions("date_start >= 2025-04-01")`. For filtering • most named arguments can be filtered • use <=, >=, <, > for specifying ranges • for date filtering, use date format "YYYY-MM-DD", and "YYYY-MM-DDTHH:MM:SS"
`session_key`	The unique identifier for the session. For `f1_sessions()` the default is `NULL` to return all sessions. For all other functions `session_key` defaults to 'latest' so as to only fetch the data for the latest/current session. Valid values are: 'latest', `NULL` or an integer session key
`driver_number`	The unique number assigned to an F1 driver
`duration_sector_1`, `duration_sector_2`, `duration_sector_3`	The time taken, in seconds to complete the given sector of the lap

i1_speed, i2_speed
: The speed of the car, in km/h, at the first and second intermediate point on the track.

is_pit_out_lap
: A boolean value indicating whether the lap is an "out lap" from the pit (true if it is, false otherwise).

lap_duration
: The total time taken, in seconds, to complete the entire lap.

lap_number
: The sequential number of the lap within the session (starts at 1).

segments_sector_1, segments_sector_2, segments_sector_3
: A list of values representing the "mini-sectors" within the first sector

st_speed
: The speed of the car, in km/h, at the speed trap, which is a specific point on the track where the highest speeds are usually recorded.

date_start
: The UTC starting date and time, in ISO 8601 format.

meeting_key
: The unique identifier for the meeting. Use 'latest' to identify the latest or current meeting.

Value

data.frame

Examples

```
f1_laps(driver_number = 44, lap_number = 1)
```

---

f1_locations *The approximate location of the cars on the circuit, at a sample rate of about 3.7 Hz.*

Description

Useful for gauging their progress along the track, but lacks details about lateral placement — i.e. whether the car is on the left or right side of the track. The origin point (0, 0, 0) appears to be arbitrary and not tied to any specific location on the track.

Usage

```
f1_locations(
  ...,
  session_key = "latest",
  driver_number,
  x,
  y,
  z,
  date,
  meeting_key
)
```

Arguments

...	Verbatim options added to API call. Use this to add filtering options e.g. `f1_sessions("date_start >= 2025-04-01")`. For filtering • most named arguments can be filtered • use <=, >=, <, > for specifying ranges • for date filtering, use date format "YYYY-MM-DD", and "YYYY-MM-DDTHH:MM:SS"
session_key	The unique identifier for the session. For `f1_sessions()` the default is `NULL` to return all sessions. For all other functions `session_key` defaults to 'latest' so as to only fetch the data for the latest/current session. Valid values are: 'latest', `NULL` or an integer session key
driver_number	The unique number assigned to an F1 driver
x, y, z	3D Cartesian coordinate system representing the current approximate location of the car on the track.
date	The UTC date and time, in ISO 8601 format.
meeting_key	The unique identifier for the meeting. Use 'latest' to identify the latest or current meeting.

Value

data.frame

Examples

```
f1_locations(driver_number = 44)
```

---

f1_meetings

Provides information about meetings i.e. a Grand Prix or testing weekend. Usually includes multiple sessions (practice, qualifying, race, ...).

Description

Provides information about meetings i.e. a Grand Prix or testing weekend. Usually includes multiple sessions (practice, qualifying, race, ...).

Usage

```
f1_meetings(
  ...,
  circuit_key,
  circuit_short_name,
  country_code,
  country_key,
```

```
  country_name,
  date_start,
  gmt_offset,
  location,
  meeting_key,
  meeting_name,
  meeting_official_name,
  year
)
```

Arguments

`...`	Verbatim options added to API call. Use this to add filtering options e.g. `f1_sessions("date_start >= 2025-04-01")`. For filtering • most named arguments can be filtered • use <=, >=, <, > for specifying ranges • for date filtering, use date format "YYYY-MM-DD", and "YYYY-MM-DDTHH:MM:SS"
`circuit_key`	The unique identifier for the circuit where the event takes place.
`circuit_short_name`	The short or common name of the circuit where the event takes place.
`country_code`	A code that uniquely identifies the country. e.g. "AUS", "USA"
`country_key`	The unique identifier for the country where the event takes place.
`country_name`	The full name of the country where the event takes place.
`date_start`	The UTC starting date and time, in ISO 8601 format.
`gmt_offset`	The difference in hours and minutes between local time at the location of the event and Greenwich Mean Time (GMT).
`location`	The city or geographical location where the event takes place.
`meeting_key`	The unique identifier for the meeting. Use 'latest' to identify the latest or current meeting.
`meeting_name`	The name of the meeting.
`meeting_official_name`	The official name of the meeting.
`year`	The year the event takes place.

Value

data.frame

Examples

```
f1_meetings()
```

f1_pits *Provides information about cars going through the pit lane.*

Description

Provides information about cars going through the pit lane.

Usage

```
f1_pits(
  ...,
  session_key = "latest",
  pit_duration,
  driver_number,
  lap_number,
  date,
  meeting_key
)
```

Arguments

...	Verbatim options added to API call. Use this to add filtering options e.g. `f1_sessions("date_start >= 2025-04-01")`. For filtering • most named arguments can be filtered • use <=, >=, <, > for specifying ranges • for date filtering, use date format "YYYY-MM-DD", and "YYYY-MM-DDTHH:MM:SS"
session_key	The unique identifier for the session. For `f1_sessions()` the default is `NULL` to return all sessions. For all other functions `session_key` defaults to 'latest' so as to only fetch the data for the latest/current session. Valid values are: 'latest', `NULL` or an integer session key
pit_duration	The time spent in the pit, from entering to leaving the pit lane, in seconds.
driver_number	The unique number assigned to an F1 driver
lap_number	The sequential number of the lap within the session (starts at 1).
date	The UTC date and time, in ISO 8601 format.
meeting_key	The unique identifier for the meeting. Use 'latest' to identify the latest or current meeting.

Value

data.frame

Examples

```
f1_pits()
```

f1_positions *Provides driver positions throughout a session, including initial placement and subsequent changes.*

Description

Provides driver positions throughout a session, including initial placement and subsequent changes.

Usage

```
f1_positions(
  ...,
  session_key = "latest",
  driver_number,
  position,
  date,
  meeting_key
)
```

Arguments

...	Verbatim options added to API call. Use this to add filtering options e.g. `f1_sessions("date_start >= 2025-04-01")`. For filtering • most named arguments can be filtered • use <=, >=, <, > for specifying ranges • for date filtering, use date format "YYYY-MM-DD", and "YYYY-MM-DDTHH:MM:SS"
session_key	The unique identifier for the session. For `f1_sessions()` the default is `NULL` to return all sessions. For all other functions `session_key` defaults to 'latest' so as to only fetch the data for the latest/current session. Valid values are: 'latest', `NULL` or an integer session key
driver_number	The unique number assigned to an F1 driver
position	Position of the driver (starts at 1).
date	The UTC date and time, in ISO 8601 format.
meeting_key	The unique identifier for the meeting. Use 'latest' to identify the latest or current meeting.

Value

data.frame

Examples

```
f1_positions()
```

`f1_race_control` *Provides information about race control (racing incidents, flags, safety car, ...).*

Description

Provides information about race control (racing incidents, flags, safety car, ...).

Usage

```
f1_race_control(
  ...,
  session_key = "latest",
  category,
  driver_number,
  flag,
  lap_number,
  message,
  scope,
  sector,
  date,
  meeting_key
)
```

Arguments

`...`	Verbatim options added to API call. Use this to add filtering options e.g. `f1_sessions("date_start >= 2025-04-01")`. For filtering • most named arguments can be filtered • use <=, >=, <, > for specifying ranges • for date filtering, use date format "YYYY-MM-DD", and "YYYY-MM-DDTHH:MM:SS"
`session_key`	The unique identifier for the session. For `f1_sessions()` the default is `NULL` to return all sessions. For all other functions `session_key` defaults to 'latest' so as to only fetch the data for the latest/current session. Valid values are: 'latest', `NULL` or an integer session key
`category`	The category of the event ('CarEvent', 'Drs', 'Flag', 'SafetyCar', ...).
`driver_number`	The unique number assigned to an F1 driver
`flag`	Type of flag displayed ('GREEN', 'YELLOW', 'DOUBLE YELLOW', 'CHEQUERED', 'RED", 'CLEAR',
`lap_number`	The sequential number of the lap within the session (starts at 1).
`message`	Description of the event or action.
`scope`	The scope of the event ('Track', 'Driver', 'Sector', ...).
`sector`	Segment ("mini-sector") of the track where the event occurred? (starts at 1).

date	The UTC date and time, in ISO 8601 format.
meeting_key	The unique identifier for the meeting. Use 'latest' to identify the latest or current meeting.

Value

data.frame

Examples

```
f1_race_control()
```

`f1_radios` *Provides a collection of radio exchanges between Formula 1 drivers and their respective teams during sessions.*

Description

Please note that only a limited selection of communications are included, not the complete record of radio interactions.

Usage

```
f1_radios(
  ...,
  session_key = "latest",
  driver_number,
  recording_url,
  date,
  meeting_key
)
```

Arguments

...	Verbatim options added to API call. Use this to add filtering options e.g. `f1_sessions("date_start >= 2025-04-01")`. For filtering • most named arguments can be filtered • use <=, >=, <, > for specifying ranges • for date filtering, use date format "YYYY-MM-DD", and "YYYY-MM-DDTHH:MM:SS"
session_key	The unique identifier for the session. For `f1_sessions()` the default is `NULL` to return all sessions. For all other functions `session_key` defaults to 'latest' so as to only fetch the data for the latest/current session. Valid values are: 'latest', `NULL` or an integer session key
driver_number	The unique number assigned to an F1 driver

recording_url	URL of the radio recording.
date	The UTC date and time, in ISO 8601 format.
meeting_key	The unique identifier for the meeting. Use 'latest' to identify the latest or current meeting.

Value

data.frame

Examples

```
f1_radios(driver_number = 44)
```

`f1_sessions` *Provides information about sessions - i.e. a distinct period of track activity during a Grand Prix (practice, qualifying, sprint, race, ...).*

Description

Provides information about sessions - i.e. a distinct period of track activity during a Grand Prix (practice, qualifying, sprint, race, ...).

Usage

```
f1_sessions(
  ...,
  session_key = NULL,
  circuit_key,
  circuit_short_name,
  country_code,
  country_key,
  country_name,
  date_start,
  date_end,
  gmt_offset,
  location,
  meeting_key,
  session_name,
  session_type,
  year
)
```

Arguments

...	Verbatim options added to API call. Use this to add filtering options e.g. `f1_sessions("date_start >= 2025-04-01")`. For filtering • most named arguments can be filtered • use <=, >=, <, > for specifying ranges • for date filtering, use date format "YYYY-MM-DD", and "YYYY-MM-DDTHH:MM:SS"
session_key	The unique identifier for the session. For `f1_sessions()` the default is `NULL` to return all sessions. For all other functions `session_key` defaults to 'latest' so as to only fetch the data for the latest/current session. Valid values are: 'latest', `NULL` or an integer session key
circuit_key	The unique identifier for the circuit where the event takes place.
circuit_short_name	The short or common name of the circuit where the event takes place.
country_code	A code that uniquely identifies the country. e.g. "AUS", "USA"
country_key	The unique identifier for the country where the event takes place.
country_name	The full name of the country where the event takes place.
date_start	The UTC starting date and time, in ISO 8601 format.
date_end	The UTC ending date and time, in ISO 8601 format.
gmt_offset	The difference in hours and minutes between local time at the location of the event and Greenwich Mean Time (GMT).
location	The city or geographical location where the event takes place.
meeting_key	The unique identifier for the meeting. Use 'latest' to identify the latest or current meeting.
session_name	The name of the session 'Practice 1', 'Qualifying', 'Race' etc
session_type	The type of the session 'Practice', 'Qualifying', 'Race' etc.
year	The year the event takes place.

Value

data.frame

Examples

```
# Find all sessions in the OpenF1 database
f1_sessions()[1:5] |>
  head()

# Show all sessions in the last 50 days
# This will user filtering by passing in an unnamed character string
limit <- Sys.Date() - 50
limit <- sprintf("date_start > %s", limit)
limit
f1_sessions(limit, session_type = 'Race')[,1:5]
```

f1_stints *Provides information about individual stints i.e. a period of continuous driving by a driver during a session.*

Description

Provides information about individual stints i.e. a period of continuous driving by a driver during a session.

Usage

```
f1_stints(
  ...,
  session_key = "latest",
  driver_number,
  compound,
  lap_end,
  lap_start,
  stint_number,
  tyre_age_at_start,
  meeting_key
)
```

Arguments

`...`	Verbatim options added to API call. Use this to add filtering options e.g. `f1_sessions("date_start >= 2025-04-01")`. For filtering • most named arguments can be filtered • use <=, >=, <, > for specifying ranges • for date filtering, use date format "YYYY-MM-DD", and "YYYY-MM-DDTHH:MM:SS"
`session_key`	The unique identifier for the session. For `f1_sessions()` the default is `NULL` to return all sessions. For all other functions `session_key` defaults to 'latest' so as to only fetch the data for the latest/current session. Valid values are: 'latest', `NULL` or an integer session key
`driver_number`	The unique number assigned to an F1 driver
`compound`	The specific compound of tyre used during the stint (SOFT, MEDIUM, HARD, ...).
`lap_start`, `lap_end`	Number of the first/last completed lap in this stint.
`stint_number`	The sequential number of the stint within the session (starts at 1).
`tyre_age_at_start`	The age of the tyres at the start of the stint, in laps completed.
`meeting_key`	The unique identifier for the meeting. Use 'latest' to identify the latest or current meeting.

Value

data.frame

Examples

```
f1_stints(driver_number = 44)
```

---

f1_weather *The weather over the track, updated every minute.*

Description

The weather over the track, updated every minute.

Usage

```
f1_weather(
  ...,
  session_key = "latest",
  rainfall,
  air_temperature,
  track_temperature,
  wind_direction,
  wind_speed,
  humidity,
  pressure,
  date,
  meeting_key
)
```

Arguments

...	Verbatim options added to API call. Use this to add filtering options e.g. `f1_sessions("date_start >= 2025-04-01")`. For filtering • most named arguments can be filtered • use <=, >=, <, > for specifying ranges • for date filtering, use date format "YYYY-MM-DD", and "YYYY-MM-DDTHH:MM:SS"
session_key	The unique identifier for the session. For `f1_sessions()` the default is `NULL` to return all sessions. For all other functions `session_key` defaults to 'latest' so as to only fetch the data for the latest/current session. Valid values are: 'latest', `NULL` or an integer session key
rainfall	Whether there is rainfall.
air_temperature	Air temperature (Celsius).

track_temperature	Track temperature (Celsius).
wind_direction	Wind direction angle in degrees, from 0 to 359.
wind_speed	Wind speed (m/s).
humidity	Relative humidity (percent).
pressure	Air pressure (mbar).
date	The UTC date and time, in ISO 8601 format.
meeting_key	The unique identifier for the meeting. Use 'latest' to identify the latest or current meeting.

Value

data.frame

Examples

```
f1_weather()
```

---

segments_sector *Information about the codes for mini-segments*

Description

Information about the codes for mini-segments

Usage

```
segments_sector
```

Format

An object of class `data.frame` with 8 rows and 2 columns.

Index