
nhl

Matt Hostetter

Feb 09, 2024

API REFERENCE

1	nhl	3
2	nhl.statsapi	7
3	Versioning	11
3.1	Alpha releases	11
3.2	Beta releases	11
	Python Module Index	13
	Index	15

The [nhl](#) library is a Python 3 package for NHL game and player stats.

NHL

A Python 3 API for NHL game and player stats.

```
class nhl.Conference(id: int, name: str, name_short: str, abbreviation: str)
```

NHL conference object.

abbreviation : str

Conference abbreviation

id : int

The NHL statsapi universal conference ID

name : str

Conference name

name_short : str

Conference name shortened

```
class nhl.Division(id: int, name: str, name_short: str, abbreviation: str)
```

NHL division object.

abbreviation : str

Division abbreviated name

id : int

The NHL statsapi universal division ID

name : str

Division name

name_short : str

Division short name

```
class nhl.Event(game_id: int, id: int, type: str, subtype: str, time: gameTime, location: Location, value: float,  
               score: tuple[int, int], by_player_id: int, on_player_id: int, by_team_id: int, on_team_id: int)
```

NHL event object.

```
class nhl.Franchise(id: int, name: str)
```

NHL franchise object.

id : int

The NHL statsapi universal franchise ID

name : str

Franchise name

class nhl.GameTime(*period: int, period_sec: int*)

NHL game time object.

This object represents a unique time of the game. There are convenience properties to convert the game time into convenient formats.

property min_sec : tuple[int, int]

Elapsed minutes and seconds of the game.

period : int

Game period. 1-3 for regulation. 4+ for overtime.

property period_min_sec : tuple[int, int]

Elapsed minutes and seconds of the period.

period_sec : int

Elapsed seconds of the period.

property period_str : str

Period number as string (i.e. "2nd")

property sec : int

Elapsed seconds of the game.

class nhl.Location(*x: int | None = None, y: int | None = None*)

NHL location object.

distance(*other*)

Measure distance between current location and another on-ice location.

Parameters

other : Location

location to measure distance from

Returns

distance (ft)

Return type

float

x : int | None = None

Rink position in ft along x-axis (length) [-100, 100]

y : int | None = None

Rink position in ft along y-axis (breadth) [-42, 42]

class nhl.Official(*id: int, name: str*)

NHL official object.

property first_name : str

Official's first name

id : int

The NHL statsapi universal official ID

property last_name : str

Official's last name

name : str

Official's full name

```
class nhl.Player(id: int, name: str, number: int, position: str, height: int, weight: int, shoots_catches: str,
                birth_date: date, birth_city: str, birth_country: str)
```

NHL player object.

property age : int

Current age in years

birth_city : str

Player's birth city

birth_country : str

Player's birth country

birth_date : date

Player's birth date

property first_name : str

Player's first name

height : int

Player's height in total inches

property height_ft_in : tuple[int, int]

Height in feet and inches (*height* // 12, *height* % 12)

id : int

The NHL statsapi universal player ID

property last_name : str

Player's last name

name : str

Player's full name

number : int

Player's primary number

position : str

Player's primary position ("LW", "C", "RW", "D", "G")

shoots_catches : str

Indication of whether the player shoots (skater)/catches (goalie) "L" or "R"

weight : int

Player's weight in lbs

```
class nhl.Shift(game_id: int, player_id: int, shift_id: int, on: gameTime, off: gameTime)
```

NHL shift object.

game_id : int

NHL statsapi unique game ID

length : int

Shift length in seconds

off : *GameTime*

Shift end game time

on : *GameTime*

Shift start game time

player_id : *int*

NHL statsapi unique player ID

shift_id : *int*

Shift number for specified game

class `nhl.Team`(*id*: *int*, *location*: *str*, *name*: *str*, *abbreviation*: *str*, *first_year*: *int*, *division*: *Division*, *conference*: *Conference*, *franchise*: *Franchise*)

NHL team object.

abbreviation : *str*

Team's name abbreviated

conference : *Conference*

The NHL conference the team is in

division : *Division*

The NHL division the team is in

first_year : *int*

First year of play

franchise : *Franchise*

The NHL franchise the team belongs to

property full_name : *str*

Team's full name

id : *int*

The NHL statsapi universal team ID

location : *str*

Team's location

name : *str*

Team's name

class `nhl.Venue`(*id*: *int*, *name*: *str*)

NHL venue object.

id : *int*

The NHL statsapi universal venue ID

name : *str*

Venue name

NHL.STATSAPI

A module containing functions for querying the NHL statsapi REST API <https://statsapi.web.nhl.com/api/v1>.

`nhl.statsapi.conference(id: int) → Conference`

Fetches a single conference by its ID.

Parameters

id: int

The NHL statsapi conference ID.

Returns

A *Conference* object.

`nhl.statsapi.conferences() → list[Conference]`

Fetches all conferences.

Returns

A list of *Conference* objects.

`nhl.statsapi.division(id: int) → Division`

Fetches a single division by its ID.

Parameters

id: int

The NHL statsapi division ID.

Returns

A *Division* object.

`nhl.statsapi.divisions() → list[Division]`

Fetches all divisions.

Returns

A list of *Division* objects.

`nhl.statsapi.franchise(id: int) → Franchise`

Fetches a single franchise by its ID.

Parameters

id: int

The NHL statsapi franchise ID.

Returns

A *Franchise* object.

`nhl.statsapi.franchises()` → `list[Franchise]`

Fetches all franchises.

Returns

A list of *Franchise* objects.

`nhl.statsapi.player(id: int)` → *Player*

Fetches a single player by its ID.

Parameters

id: int

The NHL statsapi player ID.

Returns

A *Player* object.

`nhl.statsapi.players(ids: Sequence[int])` → `list[Player]`

Fetches multiple players by their IDs.

Returns

A list of *Player* objects.

`nhl.statsapi.shifts(game_id: int, player_id: int)` → `list[Shift]`

Fetches all shifts of a player of a given game.

Parameters

game_id: int

The NHL statsapi game ID.

player_id: int

The NHL statsapi player ID.

Returns

A list of *Shift* objects.

Raises

ValueError – If the player is not found in the game.

`nhl.statsapi.team(id: int)` → *Team*

Fetches a single team by its ID.

Parameters

id: int

The NHL statsapi team ID.

Returns

A *Team* object.

`nhl.statsapi.teams()` → `list[Team]`

Fetches all teams.

Returns

A list of *Team* objects.

`nhl.statsapi.venue(id: int)` → *Venue*

Fetches a single venue by its ID.

Parameters

id: `int`

The NHL statsapi venue ID.

Returns

A `Venue` object.

`nhl.statsapi.venues()` \rightarrow `list[Venue]`

Fetches all venues.

Returns

A list of `Venue` objects.

VERSIONING

The *nhl* library uses [semantic versioning](#). Releases are versioned `major.minor.patch`.

Major versions introduce API-changing features. Minor versions add features that are backwards-compatible with other releases. Patch versions make backwards-compatible bug fixes.

3.1 Alpha releases

Versions before `0.1.0` are alpha releases. Alpha releases are versioned `0.0.alpha`. There is no API compatibility guarantee for them. They can be thought of as `0.0.alpha-major`.

3.2 Beta releases

Versions before `1.0.0` are beta releases. Beta releases are versioned `0.beta.x` and are API-compatible. They can be thought of as `0.beta-major.beta-minor`.

PYTHON MODULE INDEX

n

`nhl`, [3](#)

`nhl.statsapi`, [7](#)

A

abbreviation (*nhl.Conference attribute*), 3
 abbreviation (*nhl.Division attribute*), 3
 abbreviation (*nhl.Team attribute*), 6
 age (*nhl.Player property*), 5

B

birth_city (*nhl.Player attribute*), 5
 birth_country (*nhl.Player attribute*), 5
 birth_date (*nhl.Player attribute*), 5

C

Conference (*class in nhl*), 3
 conference (*nhl.Team attribute*), 6
 conference() (*in module nhl.statsapi*), 7
 conferences() (*in module nhl.statsapi*), 7

D

distance() (*nhl.Location method*), 4
 Division (*class in nhl*), 3
 division (*nhl.Team attribute*), 6
 division() (*in module nhl.statsapi*), 7
 divisions() (*in module nhl.statsapi*), 7

E

Event (*class in nhl*), 3

F

first_name (*nhl.Official property*), 4
 first_name (*nhl.Player property*), 5
 first_year (*nhl.Team attribute*), 6
 Franchise (*class in nhl*), 3
 franchise (*nhl.Team attribute*), 6
 franchise() (*in module nhl.statsapi*), 7
 franchises() (*in module nhl.statsapi*), 7
 full_name (*nhl.Team property*), 6

G

game_id (*nhl.Shift attribute*), 5
 GameTime (*class in nhl*), 3

H

height (*nhl.Player attribute*), 5
 height_ft_in (*nhl.Player property*), 5

I

id (*nhl.Conference attribute*), 3
 id (*nhl.Division attribute*), 3
 id (*nhl.Franchise attribute*), 3
 id (*nhl.Official attribute*), 4
 id (*nhl.Player attribute*), 5
 id (*nhl.Team attribute*), 6
 id (*nhl.Venue attribute*), 6

L

last_name (*nhl.Official property*), 4
 last_name (*nhl.Player property*), 5
 length (*nhl.Shift attribute*), 5
 Location (*class in nhl*), 4
 location (*nhl.Team attribute*), 6

M

min_sec (*nhl.GameTime property*), 4
 module
 nhl, 3
 nhl.statsapi, 7

N

name (*nhl.Conference attribute*), 3
 name (*nhl.Division attribute*), 3
 name (*nhl.Franchise attribute*), 3
 name (*nhl.Official attribute*), 4
 name (*nhl.Player attribute*), 5
 name (*nhl.Team attribute*), 6
 name (*nhl.Venue attribute*), 6
 name_short (*nhl.Conference attribute*), 3
 name_short (*nhl.Division attribute*), 3
 nhl
 module, 3
 nhl.statsapi
 module, 7
 number (*nhl.Player attribute*), 5

O

`off` (*nhl.Shift* attribute), 5

`Official` (class in *nhl*), 4

`on` (*nhl.Shift* attribute), 6

P

`period` (*nhl.GameTime* attribute), 4

`period_min_sec` (*nhl.GameTime* property), 4

`period_sec` (*nhl.GameTime* attribute), 4

`period_str` (*nhl.GameTime* property), 4

`Player` (class in *nhl*), 5

`player()` (in module *nhl.statsapi*), 8

`player_id` (*nhl.Shift* attribute), 6

`players()` (in module *nhl.statsapi*), 8

`position` (*nhl.Player* attribute), 5

S

`sec` (*nhl.GameTime* property), 4

`Shift` (class in *nhl*), 5

`shift_id` (*nhl.Shift* attribute), 6

`shifts()` (in module *nhl.statsapi*), 8

`shoots_catches` (*nhl.Player* attribute), 5

T

`Team` (class in *nhl*), 6

`team()` (in module *nhl.statsapi*), 8

`teams()` (in module *nhl.statsapi*), 8

V

`Venue` (class in *nhl*), 6

`venue()` (in module *nhl.statsapi*), 8

`venues()` (in module *nhl.statsapi*), 9

W

`weight` (*nhl.Player* attribute), 5

X

`x` (*nhl.Location* attribute), 4

Y

`y` (*nhl.Location* attribute), 4