

---

# **smite-python Documentation**

***Release 1.0<sub>r.c2</sub>***

**Jayden Bailey**

February 06, 2017



<b>1</b>	<b>API Reference</b>	<b>3</b>
1.1	Main Functions . . . . .	3
1.2	Exceptions . . . . .	6
1.3	Enums . . . . .	6
<b>2</b>	<b>Examples</b>	<b>7</b>
2.1	Getting a list of all gods . . . . .	7
<b>3</b>	<b>FAQ</b>	<b>9</b>
3.1	How do I get a devId and authKey? . . . . .	9
3.2	Why am I getting a null dataset for a player that exists? . . . . .	9
3.3	When using methods such as <code>SmiteClient.get_match()</code> , some player info is missing. Why? .	9
3.4	Does this library support older versions of Python? . . . . .	9
3.5	Do all methods work with all endpoints? . . . . .	9
<b>4</b>	<b>Indices and tables</b>	<b>11</b>



Contents:



---

## API Reference

---

The following section outlines the API of smite-python.

### 1.1 Main Functions

**class** `smite.SmiteClient` (*dev\_id, auth\_key, lang=1*)

Represents a connection to the Smite API. This class is used to interact with the API and retrieve information in JSON.

---

**Note:** Any player with Privacy Mode enabled in-game will return a null dataset from methods that require a player name

---

**get\_data\_used**()

**Returns** Returns a dictionary of daily usage limits and the stats against those limits

---

**Note:** Getting your data usage does contribute to your daily API limits

---

**get\_demo\_details** (*match\_id*)

**Parameters** *match\_id* – ID of the match

**Returns** Returns information regarding a match

---

**Note:** It is better practice to use `get_match_details()`

---

**get\_esports\_proleague\_details**()

**Returns** Returns the matchup information for each matchup of the current eSports pro league session.

**get\_friends** (*player*)

**Parameters** *player* – The player name or a player ID

**Returns** Returns a list of friends

**get\_god\_ranks** (*player*)

**Parameters** *player* – The player name or player ID

**Returns** Returns the rank and worshippers value for each God the player has played

**get\_god\_recommended\_items** (*god\_id*)

**Parameters** **god\_id** – ID of god you are querying. Can be found in `get_gods` return result.

**Returns** Returns a dictionary of recommended items for a particular god

**get\_god\_skins** (*god\_id*)

**Param** **god\_id**: ID of god you are querying. Can be found in `get_gods` return result.

**Returns** Returnss all skin information for a particular god

**get\_gods** ()

**Returns** Returns all smite Gods and their various attributes

**get\_items** ()

**Returns** Returns all Smite items and their various attributes

**get\_league\_leaderboard** (*queue, tier, season*)

**Parameters**

- **queue** – The queue to obtain data from
- **tier** – The tier to obtain data from
- **season** – The season to obtain data from

**Returns** Returns the top players for a particular league

**get\_league\_seasons** (*queue*)

**Parameters** **queue** – The queue to obtain data from

**Returns** Returns a list of seasons for a match queue

**get\_match\_details** (*match\_id*)

**Parameters** **match\_id** – The id of the match

**Returns** Returns a dictionary of the match and it's attributes.

**get\_match\_history** (*player*)

**Parameters** **player** – The player name or player ID

**Returns** Returns the recent matches and high level match statistics for a particular player.

**get\_match\_ids\_by\_queue** (*queue, date, hour=-1*)

**Parameters**

- **queue** – The queue to obtain data from
- **date** – The date to obtain data from
- **hour** – The hour to obtain data from (0-23, -1 = all day)

**Returns** Returns a list of all match IDs for a specific match queue for given time frame

**get\_match\_player\_details** (*match\_id*)

**Parameters** **match\_id** – The ID of the match

**Returns** Returns player information for a live match

**get\_motd** ()



**Returns** Returns information about the most recent Match of the Days

**get\_player** (*player\_name*)

**Parameters** **player\_name** – the string name of a player

**Returns** Returns league and non-league high level data for a given player name

**get\_player\_achievements** (*player\_id*)

**Parameters** **player\_id** – ID of a player

**Returns** Returns a select number of achievement totals for the specified player ID

**get\_player\_status** (*player\_name*)

**Parameters** **player\_name** – the string name of a player

**Returns** Returns the current online status of a player

**get\_queue\_stats** (*player, queue*)

**Parameters**

- **player** – The player name or player ID
- **queue** – The id of the game mode

**Returns** Returns match summary statistics for a player and queue

**get\_team\_details** (*clan\_id*)

**Parameters** **clan\_id** – The id of the clan

**Returns** Returns the details of the clan in a python dictionary

**get\_team\_match\_history** (*clan\_id*)

**Parameters** **clan\_id** – The ID of the clan.

**Returns** Returns a history of matches from the given clan.

**Warning:** This method is deprecated and will return a null dataset

**get\_team\_players** (*clan\_id*)

**Parameters** **clan\_id** – The ID of the clan

**Returns** Returns a list of players for the given clan.

**get\_top\_matches** ()

**Returns** Returns the 50 most watch or most recent recorded matches

**ping** ()

**Returns** Indicates whether the request was successful

---

**Note:** Pinging the Smite API is used to establish connectivity. You do not need to authenticate your ID or key to do this.

---

**search\_teams** (*search\_team*)

**Parameters** **search\_team** – The string search term to search against

**Returns** Returns high level information for clan names containing search\_team string

## 1.2 Exceptions

```
class smite.SmiteError(*args, **kwargs)
```

```
class smite.NoResultError(*args, **kwargs)
```

## 1.3 Enums

```
class smite.Endpoint
```

Valid enums: PC, PS4, XBOX

---

## Examples

---

### 2.1 Getting a list of all gods

```
from smite import SmiteClient
from smite import Endpoint

# Create a new instance of the client
smite = SmiteClient(1700, '2djsa8231jlsad92ka9d2jkad912j')
# Print JSON data for all of the gods in the game on PC
print(smite.get_gods())

# Make the library use the Xbox endpoint for future requests
smite._switch_endpoint(Endpoint.XBOX)
# Print JSON data for all of the gods in the game on Xbox
print(smite.get_gods())
```



### 3.1 How do I get a devId and authKey?

If you do not already have a developer ID and API key, you can obtain one by filling out [this form](#). Hi-Rez Studios usually responds within a few days.

### 3.2 Why am I getting a null dataset for a player that exists?

If a player has Hide My Profile enabled in-game, methods will return a null dataset.

### 3.3 When using methods such as `SmiteClient.get_match()`, some player info is missing. Why?

If a player has Hide My Profile enabled in-game, some of their data will be unavailable for match history etc.

### 3.4 Does this library support older versions of Python?

No. smite-python only supports Python 3.5 or higher. However, if you are using Python 3.4, you can use [smython](#), though it may be outdated. It does not contain all possible API calls, and doesn't support the console endpoints.

### 3.5 Do all methods work with all endpoints?

No. PC is the only endpoint guaranteed to work with all methods.



---

## Indices and tables

---

- `genindex`
- `modindex`





## E

Endpoint (class in smite), 6

SmiteClient (class in smite), 3

SmiteError (class in smite), 6

## G

get\_data\_used() (smite.SmiteClient method), 3

get\_demo\_details() (smite.SmiteClient method), 3

get\_esports\_proleague\_details() (smite.SmiteClient method), 3

get\_friends() (smite.SmiteClient method), 3

get\_god\_ranks() (smite.SmiteClient method), 3

get\_god\_recommended\_items() (smite.SmiteClient method), 4

get\_god\_skins() (smite.SmiteClient method), 4

get\_gods() (smite.SmiteClient method), 4

get\_items() (smite.SmiteClient method), 4

get\_league\_leaderboard() (smite.SmiteClient method), 4

get\_league\_seasons() (smite.SmiteClient method), 4

get\_match\_details() (smite.SmiteClient method), 4

get\_match\_history() (smite.SmiteClient method), 4

get\_match\_ids\_by\_queue() (smite.SmiteClient method), 4

get\_match\_player\_details() (smite.SmiteClient method), 4

get\_motd() (smite.SmiteClient method), 4

get\_player() (smite.SmiteClient method), 5

get\_player\_achievements() (smite.SmiteClient method), 5

get\_player\_status() (smite.SmiteClient method), 5

get\_queue\_stats() (smite.SmiteClient method), 5

get\_team\_details() (smite.SmiteClient method), 5

get\_team\_match\_history() (smite.SmiteClient method), 5

get\_team\_players() (smite.SmiteClient method), 5

get\_top\_matches() (smite.SmiteClient method), 5

## N

NoResultError (class in smite), 6

## P

ping() (smite.SmiteClient method), 5

## S

search\_teams() (smite.SmiteClient method), 5