

---

# PyTwitch Documentation

*Release v0.1.0-beta*

**Daniel Hyldebrandt Hemmingsen**

September 17, 2014



<b>1</b>	<b>Python Support</b>	<b>3</b>
<b>2</b>	<b>Twitch Authentication</b>	<b>5</b>
<b>3</b>	<b>Install</b>	<b>7</b>
<b>4</b>	<b>Structure</b>	<b>9</b>
<b>5</b>	<b>JSON Responses?</b>	<b>11</b>
5.1	Example Error Response . . . . .	11
5.2	Status Codes . . . . .	11
<b>6</b>	<b>Contents</b>	<b>13</b>
6.1	Channels . . . . .	13
6.2	StreamTip . . . . .	15



**PyTwitch** makes it easy to integrate the *Twitch API (v2)* into your own PyThon project. It also has the added bonus of being able to integrate with the **StreamTip API**.

**Warning:**

The project is still **under heavy development** as of now, so only a **limited functionality**.

Want to help? Fork the GitHub repository today at <https://github.com/dhh-hss/pytwitch>



---

## Python Support

---

As of now **PyTorch** only supports (*officially*) **Python 3.2>**.

It **might** work on *prior versions* of **Python**, but not **guaranteed**.

(There are plans to make **PyTorch officially** work with **Python 2.7>** in the future.)





---

# Twitch Authentication

---

You can read more about **scopes** and how to obtain an **OAuth token** in the *Twitch API (v2) documentation*.

Link to authentication documentation: <https://github.com/justintv/Twitch-API/blob/master/authentication.md>

Direct link to scopes section: <https://github.com/justintv/Twitch-API/blob/master/authentication.md#scope>



---

### Install

---

```
$ pip install pytwitch
```

---

**Note:**

This will also **install requests**. **Requests** is an awesome **Python Package** for making **HTTP(S) requests**.

You can read more about **requests** at <http://requests.readthedocs.org/>

---



---

### Structure

---

The structure used in this project, is heavily reflected on the actual *Twitch API* (v2) documentations. You can learn more about the *Twitch API* (v2) here: <https://github.com/justintv/twitch-api>. This **may or may not** change over time, since some of the Twitch API (v2) is rather obnoxiously named.



---

## JSON Responses?

---

All of the PyTwitch functions, even for error handling returns a JSON object right out of the box. (It's planned that PyTwitch in the future will support more than just a JSON object to be returned.)

### 5.1 Example Error Response

```
# Example get_channel() JSON response
{
  "error": "Fatal Error"
  "message": "get_channel: See the PyTwitch documentation for useage."
  "status": 101
}
```

### 5.2 Status Codes

503 = Service Unavailable

404 = Not Found

401 = Unauthorized

200 = OK

101 = Fatal Error





---

## Contents

---

### 6.1 Channels

When it comes to channel data, like getting your **stream key**, **game** and channel **title**.  
(Or even update them.) The **order** of the **arguments given**, does **NOT** matter.

---

#### Note:

You can read more about **scopes** and how to obtain an **OAuth token** in the *Twitch API (v2) documentation*.

Link to authentication section: <https://github.com/justintv/Twitch-API/blob/master/authentication.md>

Direct link to scopes: <https://github.com/justintv/Twitch-API/blob/master/authentication.md#scope>

---

#### 6.1.1 get\_channel()

With the `get_channel()` function, you can get the channel object.

#### Example

```
import pytwitch # Import of PyTwitch.

# Get the channel object either with OAuth or the name of the channel.
data = pytwitch.channels.get_channel(oauth="<oauth>", name="<test_user1>")
# Pretty printing JSON for debugging purposes using utils.pretty_json()
pytwitch.utils.pretty_json(data)
```

#### Warning:

You **can't** use both *OAuth* and *name* arguments at the same time.

*required scope(s):* channel\_read (only for **OAuth** call.)

## Arguments

```
# If you provide an oauth access token,  
# it will get you the authorized channel object back.  
oauth=""  
  
# If you provide a name, it will get you the public channel object back.  
name=""
```

### 6.1.2 set\_channel()

With the `set_channel()` function, you can set the **title** of your channel and/or the **game** of your channel.

**Warning:**  
**OAuth token is required** for the `set_channel()` function.  
*required scope(s):* channel\_editor

## Example

```
import pytwitch # Import of PyTwitch.  
  
# Set the channels title and/or game, with OAuth  
data = pytwitch.channels.set(oauth="<oauth token>", title="<title>", game="<game>")  
# Pretty printing JSON for debugging purposes using utils.pretty_json()  
pytwitch.utils.pretty_json(data)
```

---

## Note:

You are **not required** to use **both** the **title** and **game** argument\*.  
Example: You only want to **update** the **title** of the stream.  
You simply only provide the **oauth** argument and the **title** argument.

---

## Arguments

```
# Set the oauth token.  
oauth=""  
  
# Set the stream title.  
title=""  
  
# Set the game title.  
game=""
```

### 6.1.3 get\_editors()

With the `get_editors()`, you can return all the editors of a channel

**Warning:**

**OAuth token is required** for the `get_editors()` function.

*required scope(s):* channel\_read

#### Example

```
import pytwitch # Import of PyTwitch.

# Get all the editors of channel with OAuth.
data = pytwitch.channels.get_editors(oauth="<ouath>")
# Pretty printing JSON for debugging purposes using utils.pretty_json()
pytwitch.utils.pretty_json(data)
```

#### Arguments

```
# Set the oauth token.
oauth=""
```

## 6.2 StreamTip

Easily get your latest tips (donations) through the *StreamTip API*.

You can learn more about the *StreamTip* API at (<http://streamtip.com>)

(You **have** to be **logged in** to see the *API documentation*.)

### 6.2.1 get\_tips()

With the `get_tips()` function, you can retrieve tips (donations) with a **client\_id** and **access\_token**.

**Note:**

You can find your **client\_id** and **access\_token** by logging into <http://streamtip.com> and go to your *account*. (<https://streamtip.com/account/billing>)

---

**Danger:**

(Taken from the *StreamTip API* documentation)

**Do not place calls to these APIs on a publicly accessible website!**

Your **client\_id** and **access\_token** must remain confidential to ensure the security of your account.

**An opt-in, publicly accessible solution with more limited data (personal info removed) will be coming sometime in the future for those who need it.**

## Example

```
import pytwitch # Import of PyTwitch.

# Getting 25 latest tips (default) sorted by date and in desc (descending) order.
# If you only provide required client_id and access_token.
data = pytwitch.streamtip.get_tips(client_id="<client id>", access_token="<access token>")
# Pretty printing JSON for debugging purposes using utils.pretty_json
pytwitch.utils.pretty_json(data)
```

## Arguments

```
# OAuth (required)
client_id=""
access_token=""

# Shortcut argument, ommits all other arguments other than the required client_id and access_token
# top - Gets all time top tipper/donator
# recent - Gets the most recent tipper/donator
get=""

# The direction tips are returned. (**asc** or **desc**, default value is **asc**)
direction=""

# The number of tips to be returned. (default value is **25**)
limit=""

# The number of tips to skip. (default value is **0**)
offset=""

# The field to sort tips by. (**amount** and **date** are supported, default value is **date**)
sort_by=""
```