Discord ヘビーユーザによる

discord.py 解説

おまえ誰よ

- 塚田 貴史
- Discord 歴 4 年(2016 年から)
  - (歴が長いだけ....とはいわせない!!)
- お仕事・趣味で Python に触れている
- **重度(重度)**のゲーマー
- github/takapdayon

LT で持って帰ってもらいたいもの

✌bot開発は簡単だよ!✌

突然ですが!

Discord 使ってますか?


DISCORD

Discord

元々はゲーマー向けコミュニケーションツールとしてデビュー
今では、大学・企業・ゲーム以外のコミュニティでも幅広く活躍!!!

- Python.jp
- discord.py
- etc...

そして、なんとサーバ費は驚きの ZERO!(ブーストはあります)
そんな Discord サーバ...

豪華に...したくないですか?

便利にしたく...ないですか?

その願い、discord.pyで叶うかもしれません!

discord.py

Discord API をラップして Python から使えるようにしたライブラリ
中で Discord との認証等ごにょってくれているためとてもありがたい

https://discordpy.readthedocs.io/ja/latest/index.html

始め方

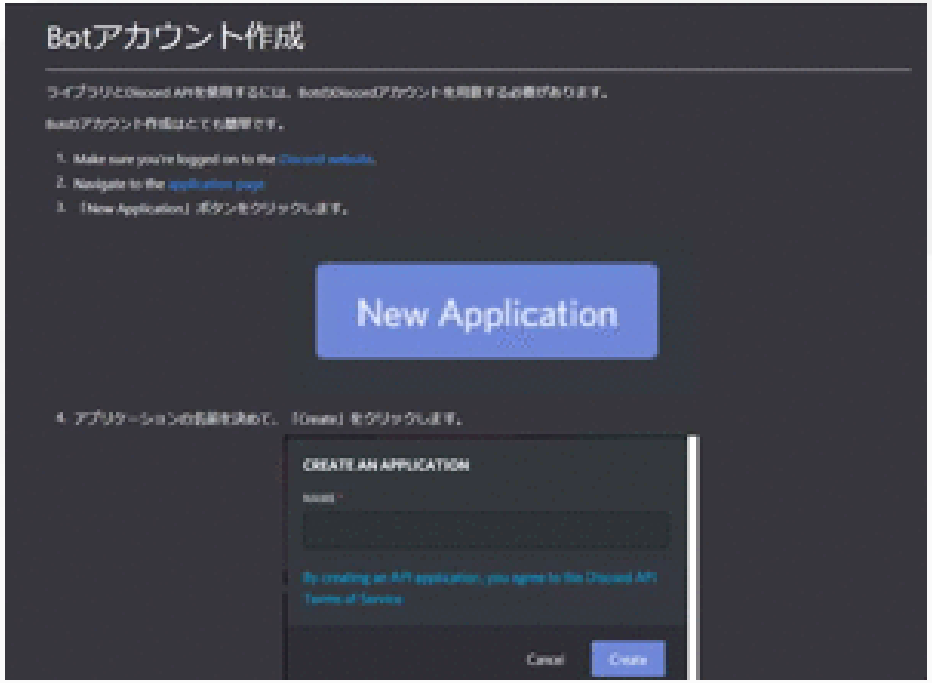
- discord bot を作成する
- pip で discord.py を入れる

これだけです!

1: discord bot を作成する

discord.py で紹介されています
https://discordpy.readthedocs.io/ja/latest/discord.html

見てわかる通り、やることは凄く少ないです



1. botの名前を決めて作成
2. botの**トークン**を入手
3. botをDiscordサーバに招待する

2: pip で discord.py を入れる

おなじみパッケージ管理ツール pip を使います

```
$ pip install discord.py

# 音声系を使う場合
$ pip install discord.py[voice]
```

Hello World!

公式からサンプルで出されている最小限コードです
https://discordpy.readthedocs.io/en/latest/quickstart.html

```
import discord

client = discord.Client()

@client.event
async def on_ready():
    print('We have logged in as {0.user}'.format(client))

@client.event
async def on_message(message):
    if message.author == client.user:
        return

    if message.content.startswith('$hello'):
        await message.channel.send('Hello!')

client.run('Botトークン')
```

Hello World!

公式からサンプルで出されている最小限コードです
https://discordpy.readthedocs.io/en/latest/quickstart.html

```
import discord

client = discord.Client()

@client.event
async def on_ready():
    print('We have logged in as {0.user}'.format(client))

@client.event
async def on_message(message):
    if message.author == client.user:
        return

    # ここ!
    if message.content.startswith('$hello'):
        await message.channel.send('Hello!')

# ここ!
client.run('Botトークン')
```

動かしてみよう

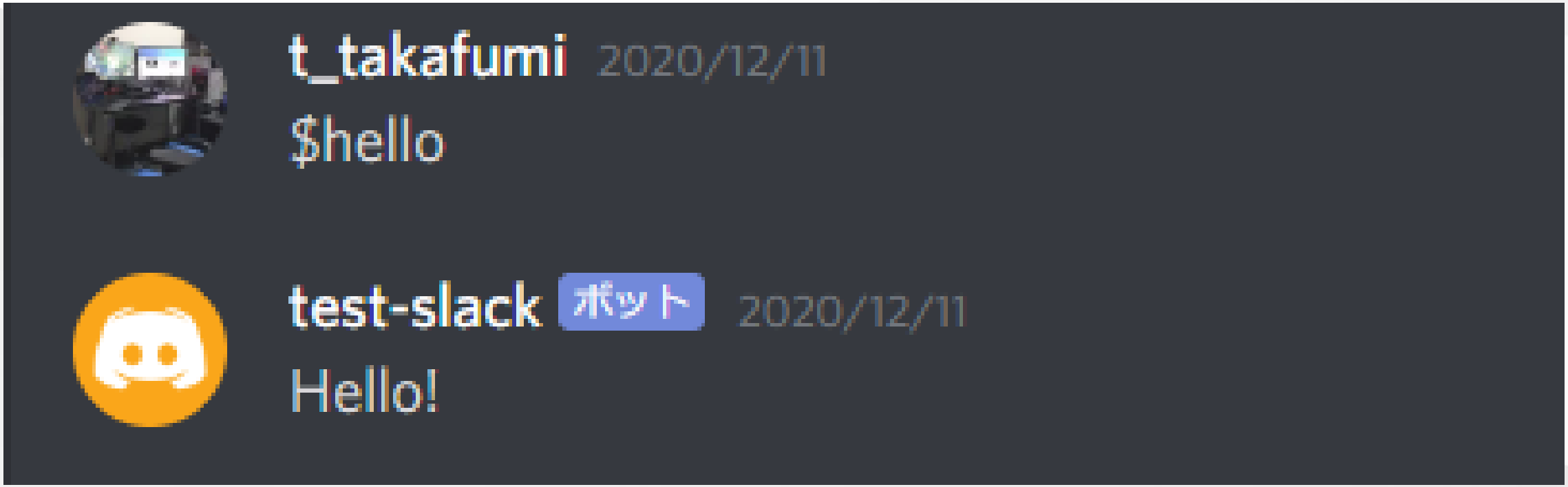
先ほどのコードを実行してみたいと思います

```
$ python Main.py
We have logged in as test-slack
```

これだけで、discord.py 側で、よしなにしてくれます

試す

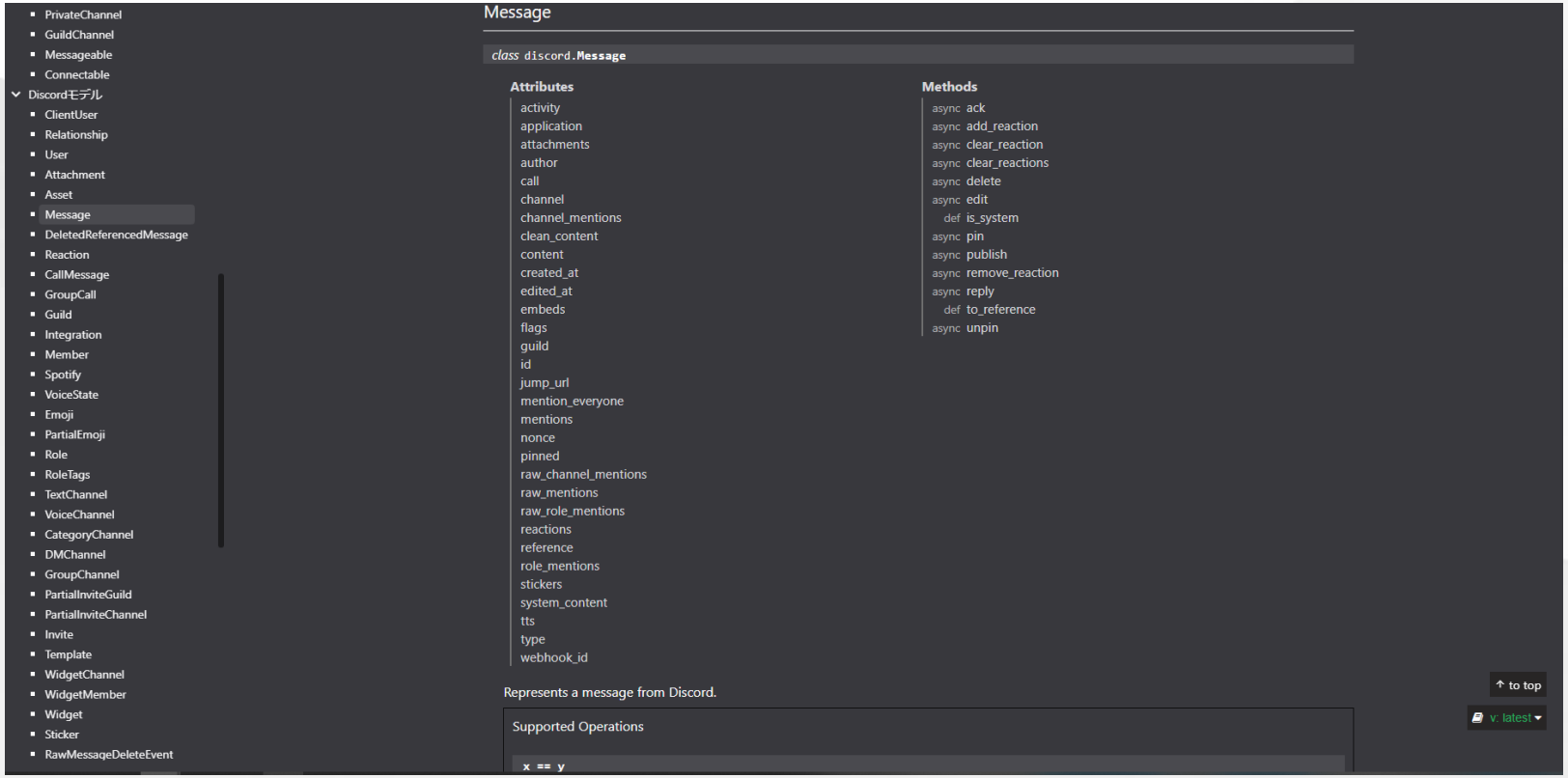
bot を導入した Discord サーバで$hello と投稿してみましょう
Bot が Hello!と返してくれれば成功です



何をトリガーにできるのか

一覧は discord.py の API で紹介されています

https://discordpy.readthedocs.io/ja/latest/api.html



ちらっと中身解説

```
client = discord.Client()

@client.event
async def on_message(message):
    if message.author == client.user:
        return
```

- @client.event
  - 対象の関数が**コルーチン関数**か判定し、コルーチンの場合は client にインスタンス変数として保持させます

ちらっと中身解説

```
client = discord.Client()

@client.event
async def on_message(message):
    if message.author == client.user:
        return
```

- async def on_message(message):
  - discord.py側で実行する関数です。messageの中にチャットしたサーバ等の情報が入ってます
- if message.author == client.user:
  - メッセージを発信した対象が bot 自身か判定(無限ループ防止)

ちらっと中身解説

```
client.run('Botトークン')
```

- client.run()
  - 中でstart関数(login関数とconnect関数)がTaskとして登録され、run_forever()で永続化されて動いています。
    asyncio の低レベルAPI()がゴリゴリ動いているので、興味ある方は見てみると面白いかもしれません!

まとめ

✌bot開発は簡単だよ!✌

You are the server customizer Champion!

(ご清聴ありがとうございました!)