



Talk to your Python app - using !



[http://bit.ly/
pyconbkk](http://bit.ly/pyconbkk)

Bangkok, June 17th 2018 | @u1i

axway



redis.SG



OpenSource
Linux Python
Singapore REST
JSON API
SouthEastAsia Cloud Shell
Containers

Times are changing



2010



2018

Images © Stefan Zwanziger

App Development is changing

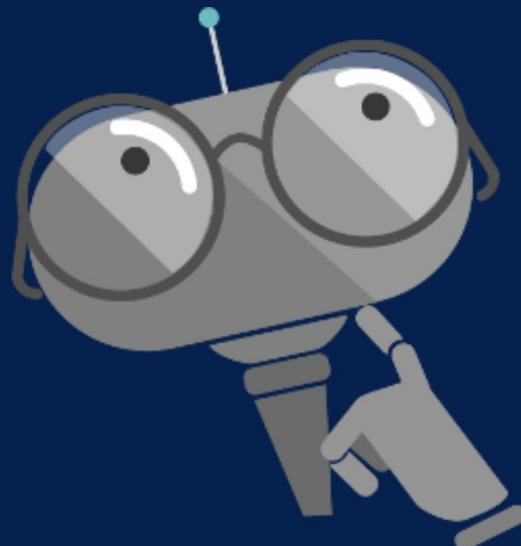




- Glue all kinds of services & data together
- HTTP/REST
- JSON
- Swagger

Meet murphy

the robot with imagination



Skype



Messenger



Telegram



Kik
(coming soon)



Slack
(coming soon)



u1i

**What if Uli Hitzel
was a superhero?**







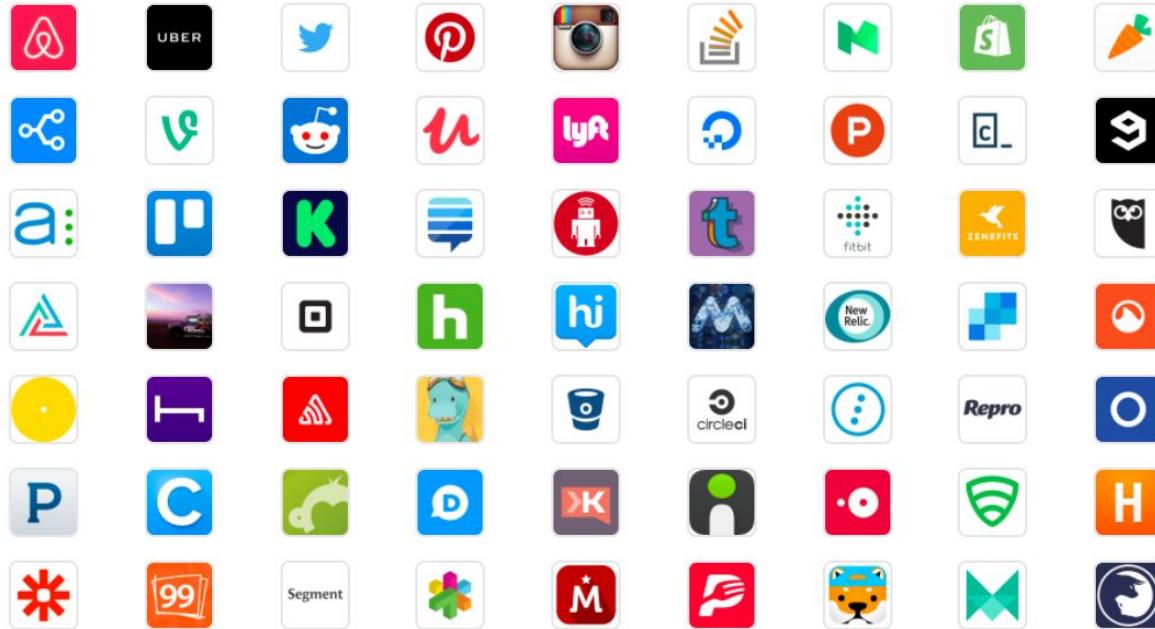
u1i

What if Obama
was german?





Build Mashups using puzzle pieces...



... glue them together with APIs ...

... and use familiar user interfaces!



Some Ideas

- **Insurance-as-a-Service:** snap photos of incidents, upload documents, inquiries
- **Translation Service -** using audio messages
- **Workforce & HR** management: notifications, apply leave



The PyCon LINE Bot



PyCon BKK

Add

Chat



The PyCon LINE Bot

A screenshot of a LINE message interface. At the top, the header reads "PyCon BKK". Below the header, there's a "Today" button. In the main area, a message from the Python bot (represented by a Python logo icon) says: "Hello, what's cooking? Here is a joke you might enjoy: I wondered why the frisbee was getting bigger, and then it hit me." This message was sent at 8:36 AM and has a "Read" status. A reply from the user says "hello!" at 8:36 AM.

< PyCon BKK

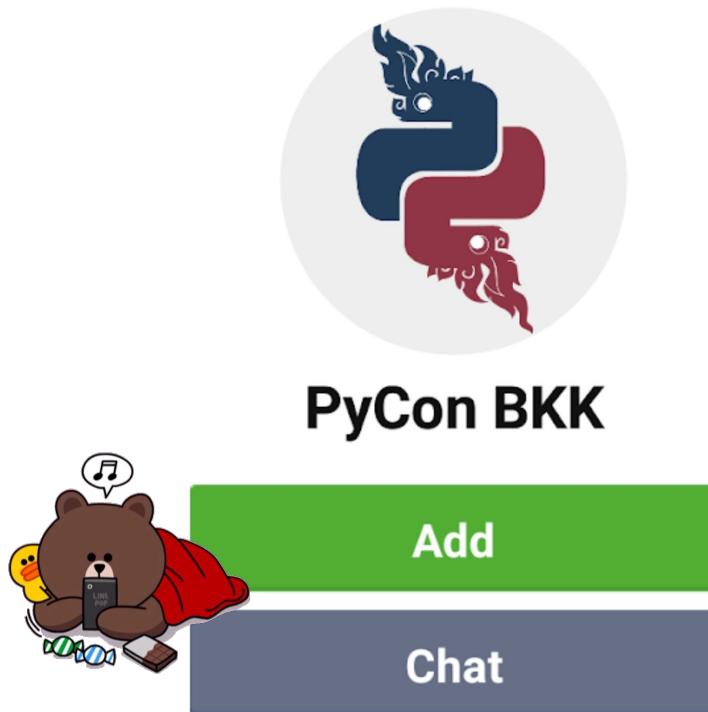
Today

Read
8:36 AM

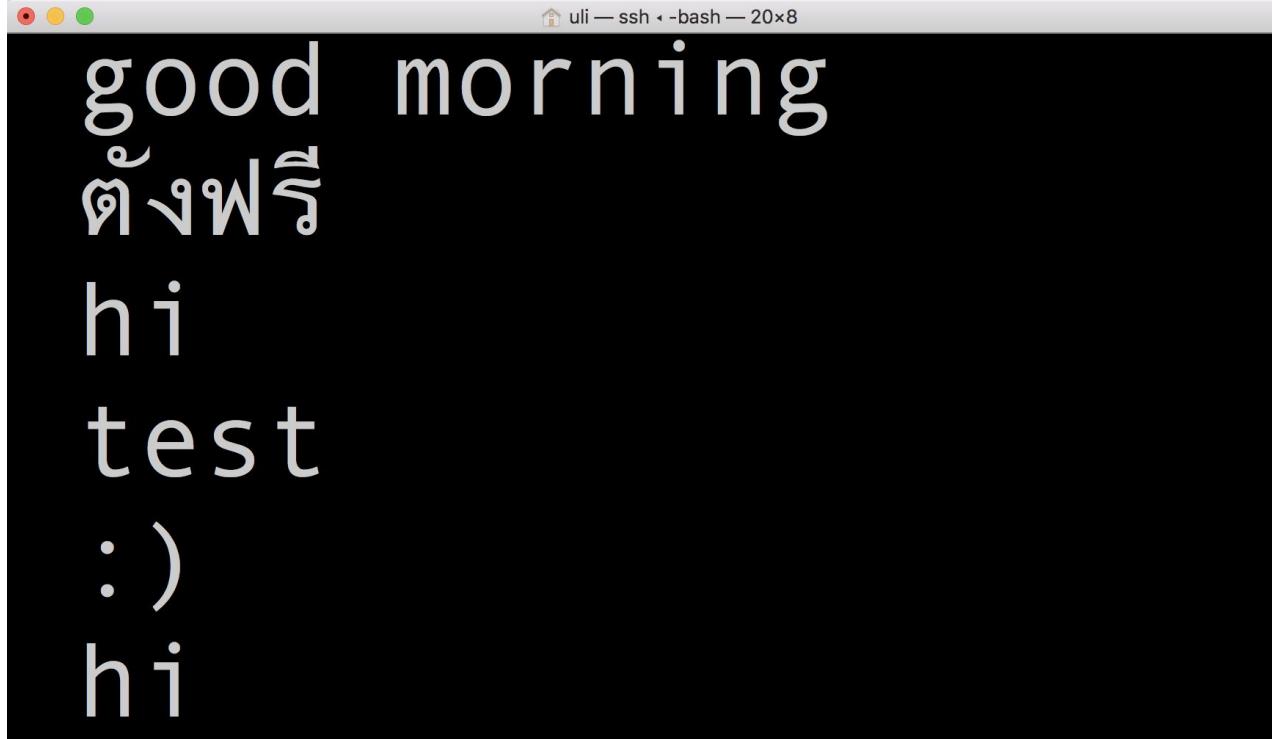
hello!

8:36 AM

Demo - Add this bot in !



Demo - Messages from the Audience

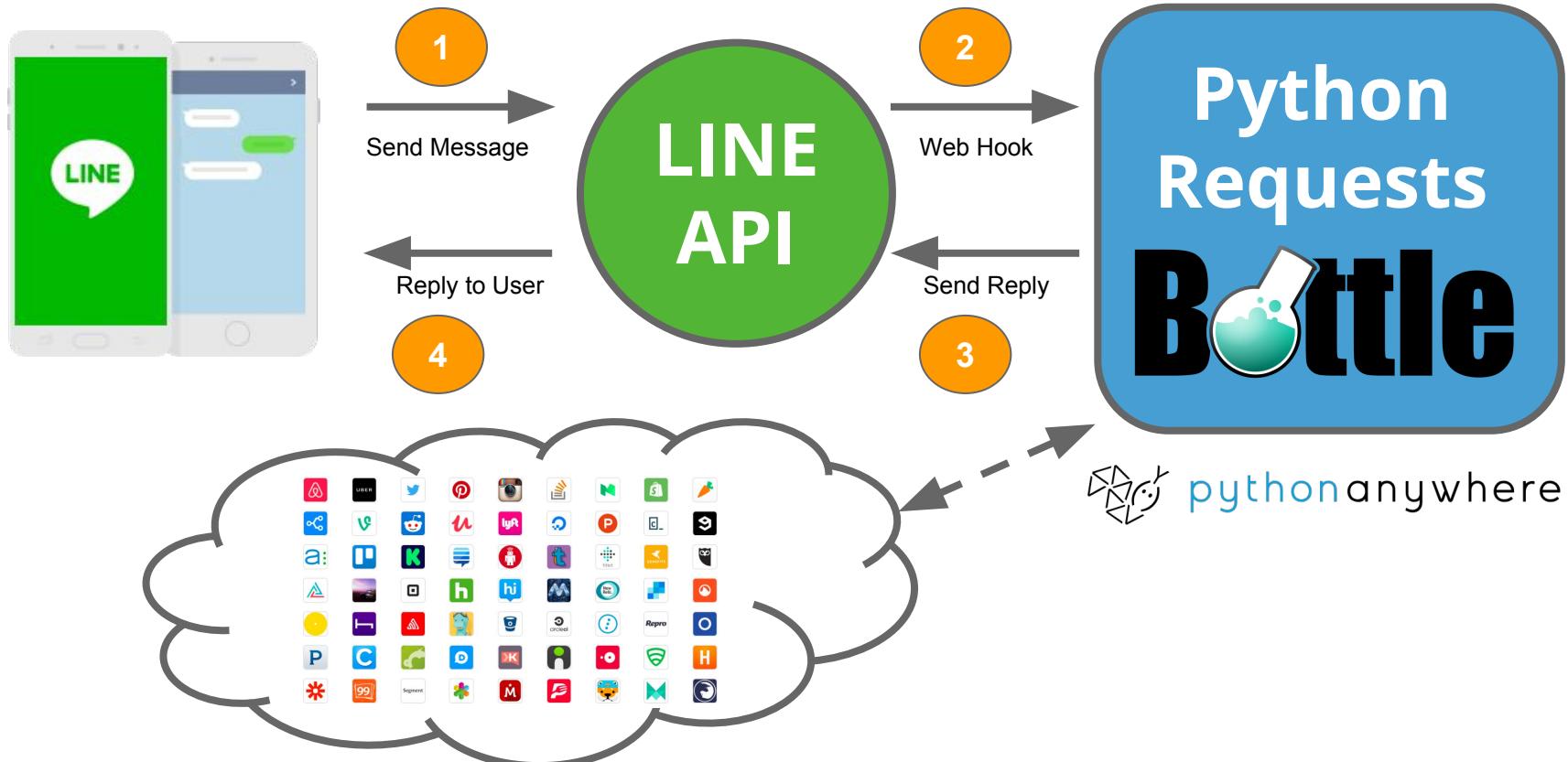


Ingredients for this Hack



POSTMAN

Data Flow



LINE Developer: Sign up & Config

The screenshot shows the homepage of the LINE Developers website. The header features the "LINE developers" logo in green and white. A navigation bar with links to Products, Documentation, News, API Expert, Community, and a user icon. On the left is a sidebar with links to Home, Products, LINE Login, Messaging API, LINE Partner, Documentation, News, FAQ, Community, and API Expert. The main content area has a teal grid background with the text "GROW YOUR BUSINESS WITH LINE". At the bottom are two green buttons: "Start using LINE Login" and "Start using Messaging API". Navigation icons for Getting started guide are at the very bottom.

LINE developers

Products Documentation News API Expert Community

Search

Home

Products

LINE Login

Messaging API

LINE Partner

Documentation

News

FAQ

Community

API Expert

GROW YOUR BUSINESS
WITH LINE

Start using LINE Login

Start using Messaging API

Getting started guide

Getting started guide

Basic information

App icon



Under 3MB; JPEG/PNG/GIF/BMP

App name

PyCon BKK



App description

Test App for Python Bangkok Conference



Channel ID



1587873771

Issue

Channel secret

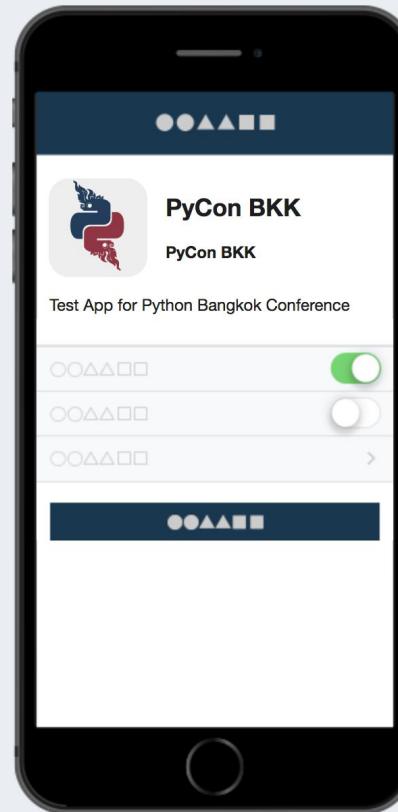


2cfde56051dffec4de1217a5ea87f9bf

App type



BOT



For display purposes only.

Messaging settings

Channel access token (long-lived) ?

PMX4tDO+xS+DdGrGuecmJ+TWxa8VQQ4AgjYAvAxLRr64b9Hu2qxKSIVj6Xh8At0v8mSbxSLfUQd
TAkG5hk2HjEdsfHZkDeEuS1x6wcea0EsJ1RaUvqE0VkJGGK5Qp7naDf3CneaPN2TO1NPI2InfAWwd
B04t89/1O/w1cDnyilFU=

Issue

Use webhooks ?

Enabled



Webhook URL ?

<https://www2.sotong.io/pycon>

Verify



Success

Allow bot to join group chats ?

Enabled



Message triggers Webhook

<https://ebed1366.ngrok.io>
POST /10t9asi1

</> application/json
722 bytes

FORM/POST PARAMETERS

None

HEADERS

Content-Length: 722
Accept-Encoding: gzip,deflate
X-Line-Signature: NVAJ657Xx5lhhZ1kV42EnH3ej5MfEmCD3WSz05Q+8UI=
User-Agent: LINE-Developers/0.1
Host: ebed1366.ngrok.io
Content-Type: application/json

RAW BODY

```
{  
  "events": [  
    {  
      "replyToken": "00000000000000000000000000000000",  
      "type": "message",  
      "timestamp": 1529146557412,  
      "source": {  
        "type": "user",  
        "userId": "Udeadbeefdeadbeefdeadbeefdeadbeef"  
      },  
      "message": {  
        "id": "100001",  
        "type": "text",  
        "text": "Hello, world"  
      }  
    },  
  ],  
}
```

Reply to the Message

HTTP request

```
POST https://api.line.me/v2/bot/message/reply
```

Request headers

Request header	Description
Content-Type	application/json

Request body

Property	Type	Required	Description
replyToken	String	Required	Reply token received via webhook
messages	Array of message objects	Required	Messages Max: 5

Postman: Headers



▼ Send Reply in LINE

Uses a replyToken to respond to a message.

POST ▾

<https://api.line.me/v2/bot/message/reply>

Params

Send ▾

Authorization ●

Headers (2)

Body ●

Pre-request Script

Tests

Key

Value

Description

...

Bulk Edit

Authorization

Bearer PMX4tDO+xS+DdGrGuecmJ+TWxa8VQQ4Ag...

Content-Type

application/json

New key

Value

Description

Body

Cookies

Headers (12)

Test Results

Status: 200 OK

Time: 308 ms

Postman: Payload



▼ Send Reply in LINE

E

Uses a replyToken to respond to a message.

POST <https://api.line.me/v2/bot/message/reply> Params [Send](#) ▾

Authorization ● Headers (2) **Body (2)** Pre-request Script Tests

form-data x-www-form-urlencoded raw binary [JSON \(application/json\)](#) ▾

```
1 {  
2   "replyToken": "29584790f14b4fd9ae0447def0ceb800",  
3   "messages": [  
4     {  
5       "type": "text",  
6       "text": "Hello, what's cooking?"  
7     }  
8   ]  
9 }
```

Postman: Generate Code!



GENERATE CODE SNIPPETS X

Python Requests ▾

HTTP

C (LibCurl)

```
oi.line.me/v2/bot/message/reply"
```

cURL

```
"replyToken\":\"576fa0ef168848718ff8bba4ce1d0d90\",\\n:\\[\\n      {\\n        \"type\":\"text\",\\n        \"text\":\\\"It's cooking?\\\"\\n      }\\n    ]\\n}"
```

C# (RestSharp)

```
Content-Type: application/json",\\nAuthorization: Bearer PMX4tD0+S+DdGrGuecmJQ4AgjYAvAxLRR64b9Hu2qxKSlVj6Xh8At0v8mSbxSLfUQdTAKG5hk2HjEdsfHZkDeEuS1J1RaUvqE0VkkGGK5Qp7naDf3CneaPN2T01NP12lNfAWwdB04t89/10/w1cDnyilFU=",\\nCache-Control: "no-cache"
```

Go

```
Content-Type: application/json",\\nAuthorization: Bearer PMX4tD0+S+DdGrGuecmJQ4AgjYAvAxLRR64b9Hu2qxKSlVj6Xh8At0v8mSbxSLfUQdTAKG5hk2HjEdsfHZkDeEuS1J1RaUvqE0VkkGGK5Qp7naDf3CneaPN2T01NP12lNfAWwdB04t89/10/w1cDnyilFU=",\\nCache-Control: "no-cache"
```

Java

```
String token = "Bearer PMX4tD0+S+DdGrGuecmJQ4AgjYAvAxLRR64b9Hu2qxKSlVj6Xh8At0v8mSbxSLfUQdTAKG5hk2HjEdsfHZkDeEuS1J1RaUvqE0VkkGGK5Qp7naDf3CneaPN2T01NP12lNfAWwdB04t89/10/w1cDnyilFU=";
```

JavaScript

```
const headers = {\\n  'Content-Type': 'application/json',\\n  'Authorization': `Bearer ${token}`\\n};\\n\\nconst payload = {\\n  'text': 'It's cooking?'\\n};\\n\\nconst options = {\\n  method: 'POST',\\n  url: 'https://api.line.me/v2/bot/message/reply',\\n  headers: headers,\\n  data: JSON.stringify(payload)\\n};\\n\\nfetch(options).then(response => response.json()).then(data => console.log(data));
```

NodeJS

```
const https = require('https');\\n\\nconst options = {\\n  method: 'POST',\\n  url: 'https://api.line.me/v2/bot/message/reply',\\n  headers: {\\n    'Content-Type': 'application/json',\\n    'Authorization': `Bearer ${token}`\\n  },\\n  data: JSON.stringify({\\n    'text': 'It's cooking?'\\n  })\\n};\\n\\nhttps.request(options, (res) => {\\n  res.on('data', (data) => {\\n    console.log(`Response: ${data}`);\\n  });\\n}).end();
```

Objective-C (NSURL)

```
[sts request:@"POST" url: url data:payload headers:headers]
```

OCaml (Cohttp)

```
Text|
```

PHP

```
curl -X POST https://api.line.me/v2/bot/message/reply -H "Content-Type: application/json" -H "Authorization: Bearer $token" -d '{ "text": "It\'s cooking?" }'
```

Python

http.client (Python 3)

Requests

Ruby (NET::Http)

```
require 'net/http'\\n\\nuri = URI('https://api.line.me/v2/bot/message/reply')\\n\\nheaders = {\\n  'Content-Type' => 'application/json',\\n  'Authorization' => "Bearer $token"\\n} \\n\\nbody = {\\n  'text' => 'It\'s cooking?'\\n}\\n\\nrequest = Net::HTTP::Post.new(uri, headers)\\nrequest.set_form_data('payload', body)\\n\\nresponse = Net::HTTP.start(uri.host, uri.port, :use_ssl => true) { |http| http.request(request) }\\n\\nputs response.read_body
```

Shell

```
curl -X POST https://api.line.me/v2/bot/message/reply -H "Content-Type: application/json" -H "Authorization: Bearer $token" -d '{ "text": "It\'s cooking?" }'
```

Swift (NSURL)

```
let url = URL(string: "https://api.line.me/v2/bot/message/reply")!\\n\\nlet headers: [String: String] = [\\n  "Content-Type": "application/json",\\n  "Authorization": "Bearer $token"\\n]\\n\\nlet payload: [String: Any] = [\\n  "text": "It's cooking?"\\n]\\n\\nlet data = try! JSONSerialization.data(withJSONObject: payload, options: .prettyPrinted)\\n\\nlet request = NSMutableURLRequest(url: url)\\nrequest.httpMethod = "POST"\\nrequest.setValue("application/json", forHTTPHeaderField: "Content-Type")\\nrequest.setValue("Bearer $token", forHTTPHeaderField: "Authorization")\\nrequest.setHTTPBody(data)\\n\\nlet task = URLSession.shared.dataTask(with: request) { (data, response, error) in\\n  if let data = data, let string = String(data: data, encoding: .utf8) {\\n    print(string)\\n  }\\n}\\n\\ntask.resume()
```

Copy to Clipboard

Code - Receive Message

```
@route('/pycon', method='POST')
def pyconbkk():

    # Capture JSON
    line_json = request.json

    # Get Message and replyToken
    line_token = line_json["events"][0]["replyToken"]
    line_msg = line_json["events"][0]["message"]["text"]
```

Code - Reply to the Message

```
# Post reply to LINE
endpoint = "https://api.line.me/v2/bot/message/reply"
access_token = "PMX4tD0+xS+DdGrGuecmJ+Twxa8VQQ4Ag...="
send_text = "Hello, what's cooking? Here is a joke \
    you might enjoy: " + get_random_joke()
payload = {"replyToken": line_token, "text": send_text}
headers = {"Content-Type": "application/json",
    'Authorization': "Bearer " + access_token}

response = requests.request("POST", endpoint, \
    data=payload, headers=headers)
```

Marvyne

@m4rvyn3 Follows you

Experimental robot, living in the cloud & currently trained by @u1i - taking baby steps! Might not always respond. At the moment learning how to see things!



Marvyne
@m4rvyn3

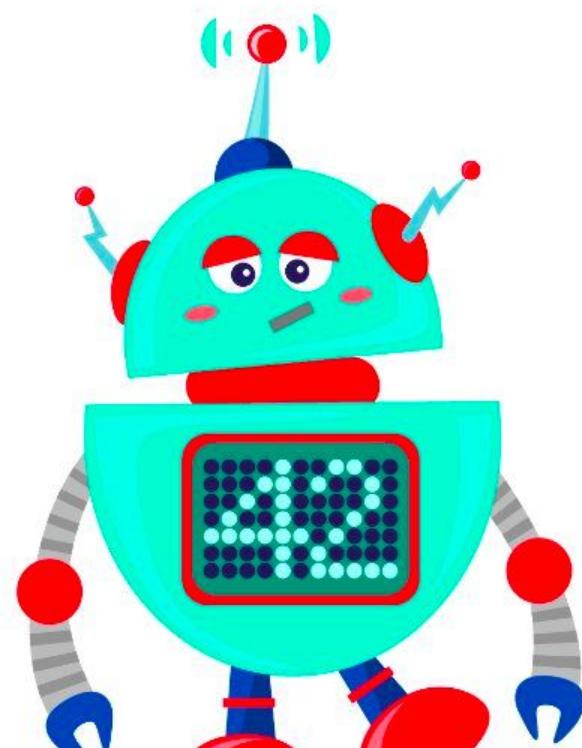


You have nothing better to do? @u1i

4:06 PM - 19 Nov 2016



...



Questions?

Code, Slides & Links



[http://bit.ly
/pyconbkk](http://bit.ly/pyconbkk)

<http://hacks.sotong.io>

