

# 파이썬을 활용한 업무자동화

바이트디그리 대상 세미나

# 목차

바이트디그리 대상 세미나

- 세미나의 목표
- 업무자동화 소개
- 코드로 보는 사례
  - ▶ 은행 서비스(입출금내역 확인) 자동화
  - ▶ 인터파크 공연 티켓팅 자동화(1자리!)
  - ▶ 유튜브 공개데이터 크롤링
- 마무리

# 세미나의 목표

1

## ■ 세미나의 목표

- 새로운 기술을 배워보자
  - ▶ 처음만나서 2시간만에 새로운 기술을 소개하면, [?]
- 꺾여가는 의지를 다시 세워보자
  - ▶ 재밌는(?) 업무자동화 사례로 코드 리뷰
  - ▶ 사실 업무자동화라고 포장했지만 그냥 자동화

# 업무자동화 소개

## ■ 업무자동화란

- 업무에 쓰이는 프로그램 혹은 서비스와 행위를 파이썬 코드로 작성
- 업무에 쓰이는 프로그램 혹은 서비스
  - ▶ 웹 브라우저
  - ▶ 엑셀
  - ▶ 구글 스프레드 시트
  - ▶ 이메일
  - ▶ 등 ..

# 코드로 보는 사례

은행서비스 (입출금내역 조회) 자동화

3

## CODEF 서비스 이용

- <https://codef.io/#/products>

CODEF를 이용한 서비스를 만들기 위해서 아래의 방법중 하나를 선택해 주세요.



API 제공량  
샌드박스  
서비스 상담  
실제 계정 연동  
기본 통계 도구 ⓘ  
고급 통계 도구 ⓘ  
온라인(메일) 상담  
원격/전화 기술상담  
엔지니어 교육(예정)  
분석 보고서(예정)  
CODEF 분석 API

	STARTER	DEVELOPER	BUSINESS <small>Popular</small>	PARTNER
	테스트 계정을 이용하여 CODEF 시스템을 구축하세요. 샌드박스 테스트 >	실제 금융기관 계정을 이용하여 시스템을 구축 및 테스트 가능합니다. (일 최대 100회 API 제공) 사용 중인 서비스 >	CODEF가 제공하는 모든 은행 및 기관을 무제한으로 사용할 수 있습니다. 정식 버전 >	맞춤형 서비스 및 CODEF 분석 API를 제공합니다. 서비스 문의 >
	무료	무료	50만원+/월	가격협의
API 제공량	×	일 최대 100회	무제한	무제한
샌드박스	✓	✓	✓	✓
서비스 상담	✓	✓	✓	✓
실제 계정 연동	×	✓	✓	✓
기본 통계 도구 ⓘ	×	✓	✓	✓
고급 통계 도구 ⓘ	×	×	✓	✓
온라인(메일) 상담	×	×	✓	✓
원격/전화 기술상담	×	×	✓	✓
엔지니어 교육(예정)	×	×	✓	✓
분석 보고서(예정)	×	×	×	✓
CODEF 분석 API	×	×	×	✓

# 코드로 보는 사례

은행서비스 (입출금내역 조회) 자동화

4

## ■ 어떻게 동작할까?

- Flask로 API 서버 개발
  - ▶ API로 은행 계좌 조회 기능을 만듦
- Slack 명령어와 API 연동
- 슬랙 특정 채널에서 명령어를 입력하면 결과를 전달
- 확인해보기

# 코드로 보는 사례

5

은행서비스 (입출금내역 조회) 자동화

## ■ Flask 서버 활용

```
import os
from flask import Flask
from api_v1 import api as api_v1

app = Flask(__name__)
app.register_blueprint(api_v1, url_prefix='/api/v1')

basedir = os.path.abspath(os.path.dirname(__file__))
app.config['SECRET_KEY'] = 'alghost-sadarifilm-HASDFKJZXVQW'

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8009)
```

# 코드로 보는 사례

6

```
from threading import Thread
from . import api

@api.route('/kb', methods=['POST'])
def bank():
    data = request.form

    if data.get('channel_id') != '<allowed_channel_id>':
        return '권한이 없는 채널입니다'

    args = data.get('text').split(' ')

    transaction_days = 1

    if len(args) >= 1:
        if args[0].isdigit():
            transaction_days = int(args[0])

    response_url = data.get('response_url')

    thr = Thread(target=realbank, args=[transaction_days, response_url])
    thr.start()

    return '준비가되면 보내드릴게요'
```



```
import requests
import json
from .kb import get_transactions

def realbank(days, url):
    account = '<계좌번호>'
    res, t = get_transactions(account, days)
    msg = ''
    if not res:
        msg = '잠시후에 다시 시도해주세요.'
    else:
        balance = format(int(t.get('resAccountBalance')), ',')
        msg += '현재잔액: %s 원\n' % balance

        trans = t.get('resTrHistoryList')
        for t in trans:
            _time = t.get('resAccountTrTime')
            _time = _time[:2] + '시' + _time[2:4] + '분' + _time[4:] + '초'
            ttype = '입금'
            price = t.get('resAccountIn')
            if t.get('resAccountOut') != '0':
                price = t.get('resAccountOut')
                ttype = '출금'
            price = format(int(price), ',')
            msg += '%s\t%s\t%s\t%s\t%s\n' % (ttype, t.get('resAccountTrDate'),
            _time, t.get('resAccountDesc3'), price)

    payload = {"text":msg, "username": "다리봇"}
    requests.post(url, data=json.dumps(payload))
```

# 코드로 보는 사례

8

은행서비스 (입출금내역 조회) 자동화

## CODEF 예제코드 그대로 활용

- CODEF 예제코드: <https://github.com/codef-io/codef-python>

```
def get_transactions(account, days=1):
    codef_url = 'https://development.codef.io'
    token_url = 'https://oauth.codef.io/oauth/token'
    transaction_path = '/v1/kr/bank/b/account/transaction-list'
    token = ''
    enddate = datetime.datetime.now()
    startdate = enddate - datetime.timedelta(days=days)

    body = {
        'connectedId': '<ID>',
        'organization': '0004',
        'account': account,
        'startDate': startdate.strftime('%Y%m%d'),
        'endDate': enddate.strftime('%Y%m%d'),
        'orderBy': '0',
    }
    ...
```

# 코드로 보는 사례

9

은행서비스 (입출금내역 조회) 자동화

## CODEF 예제코드 그대로 활용

- CODEF 예제코드: <https://github.com/codef-io/codef-python>

...


```
response_oauth = request_token(token_url, 'token1', 'token2')
if response_oauth.status_code == 200:
    dict = json.loads(response_oauth.text)
    token = dict['access_token']
    response = http_sender(codef_url + transaction_path, token, body)
    if response.status_code == 200:          # success
        dict = json.loads(urllib.parse.unquote_plus(response.text))
        if 'data' in dict and str(dict['data']) != '{}':
            response.connection.close()
            response = None
            return True, dict['data']
        response.connection.close()
    response = None
response_oauth.connection.close()
response_oauth = None
return False, dict
```

# 코드로 보는 사례


10














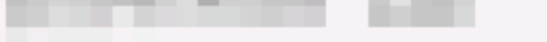

은행서비스 (입출금내역 조회) 자동화

Only visible to you








 **다리봇** APP 1:57 PM  
준비가되면 보내드릴게요

Only visible to you

현재잔액: 

출금	20200130	13시43분07초	
출금	20200130	12시19분43초	
출금	20200130	12시17분44초	
출금	20200130	11시47분55초	
출금	20200130	09시53분54초	
출금	20200129	21시21분22초	
출금	20200129	20시28분18초	
출금	20200129	16시37분50초	
출금	20200129	16시14분50초	
출금	20200129	15시05분37초	
출금	20200129	14시30분23초	
출금	20200129	12시47분21초	
출금	20200129	11시44분43초	
출금	20200129	09시39분01초	
출금	20200129	09시01분53초	

/bank

 **B** *I*      

Aa @ 😊

# 코드로 보는 사례

11

인터파크 공연 티켓팅 자동화 (1자리!)

## ■ 인터파크 공연 티켓팅

- Selenium을 활용한 공연 티켓팅 (~~아계 업무자동화인가~~)
- Selenium 맛보기 예시 먼저 보기

# 코드로 보는 사례

12

인터파크 공연 티켓팅 자동화 (1자리!)

## ■ 인터파크 공연 티켓팅

- 로그인은 어떻게?
  - ▶ 자동로그인 기능 사용. user-data-dir 활용.

# 코드로 보는 사례

13

인터파크 공연 티켓팅 자동화 (1자리!)

## 인터파크 공연 티켓팅 - 티켓팅 클릭

```
import os, time
import traceback
from selenium import webdriver
from selenium.webdriver.common.action_chains import ActionChains

basedir = os.path.dirname(os.path.abspath(__file__))
basedir = os.path.join(basedir, 'Chrome')
opts = webdriver.ChromeOptions()
opts.add_argument('user-data-dir=' + basedir)
d = webdriver.Chrome(options=opts)
date = '2020-03-13'
count = 1 #회차

try:
    #d.get('https://ticket.interpark.com')
    #input()
    d.get('티켓팅주소')
    elem = d.find_element_by_class_name('btn_rev')
    elem.click()

    windows = d.window_handles
    d.switch_to.window(windows[1])
```

# 코드로 보는 사례

14

인터파크 공연 티켓팅 자동화 (1자리!)

## ■ 인터파크 공연 티켓팅 - 날짜 선택

...

```
year = date.split('-')[0]
month = date.split('-')[1]
day = date.split('-')[2]
for _ in range(1000):
    try:
        d.switch_to.frame('ifrmBookStep')
        break
    except:
        pass

m = d.find_element_by_class_name('month')
mnd = m.find_elements_by_xpath('./span/em')

if int(mnd[0].text) != int(year) or int(mnd[1].text) != int(month):
    d.execute_script("fnChangeMonth('"+ year + month + "')")

_day = d.find_element_by_link_text(str(int(day)))
_day.click()
```



# 코드로 보는 사례

15

인터파크 공연 티켓팅 자동화 (1자리!)

## ■ 인터파크 공연 티켓팅 - 회차 선택

...

```
counts = d.find_elements_by_xpath(' //span[@id="TagPlaySeq"]/ul/li/a' )
for _ in range(1000):
    counts = d.find_elements_by_xpath(' //span[@id="TagPlaySeq"]/ul/li/a' )
    if counts: break

counts[count-1].click()
d.switch_to.default_content()
d.switch_to.window(windows[1])

ac = ActionChains(d)
for _ in range(100):
    try:
        nextbtn = d.find_element_by_id('LargeNextBtn')
        ac.move_to_element(nextbtn)
        ac.click()
        ac.perform()
        break
    except Exception as e:
        print(e)
```

# 코드로 보는 사례

16

인터파크 공연 티켓팅 자동화 (1자리!)

## ■ 인터파크 공연 티켓팅 - 회차 선택

...

```
d.switch_to.frame('ifrmSeat')
for _ in range(100):
    try:
        fold = d.find_element_by_xpath('//div[@id="divCaptchaFolding"]/a')
        fold.click()
        break
    except:
        pass
for _ in range(1000):
    try:
        d.switch_to.frame('ifrmSeatDetail')
        break
    except:
        pass
seats = d.find_elements_by_xpath('//img[@src="http://ticketimage.interpark.com/TMGSNAS/TMGS/G/1_90.gif"]')
if not seats:
    seats = d.find_elements_by_xpath('//img[@src="http://ticketimage.interpark.com/TMGSNAS/TMGS/G/2_90.gif"]')
if seats:
    seats[0].click()
```

# 코드로 보는 사례

유튜브 공개 데이터 크롤링

17

## ■ 유튜브 공개 데이터

- 공개된 데이터
  - ▶ 좋아요 수, 댓글 수, 카테고리, 영상 주소, 관련 동영상 등
- API도 있는데 왜 크롤링을 할 까?
  - ▶ Costs...

# 코드로 보는 사례

유튜브 공개 데이터 크롤링

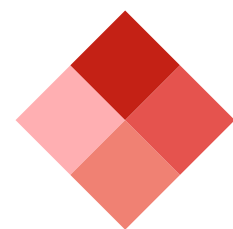
18

## ■ 유튜브 공개 데이터

- 공개된 데이터
  - ▶ 좋아요 수, 댓글 수, 카테고리, 영상 주소, 관련 동영상 등
- API도 있는데 왜 크롤링을 할 까?
  - ▶ Costs...
- 코드는?
  - ▶ 너무 길다. VSCode로!

## ■ 파이썬으로 할 수 있는 것들

- 수강하고 있는 웹 개발, 빅데이터 관련이 아니더라도 무궁무진
- 게다가, 이것들이 융합되면 할 수 있는 건 더 무궁무진
- 은행 자동화
  - ▶ 슬랙연동을 위해 Flask 웹(API)서버 개발
  - ▶ 만약, 간단한 웹사이트에서 버튼으로 자동화하고 싶다면? 웹개발
  - ▶ 수집된 내용을 분석해서 시각화/정리하고 싶다면? (빅)데이터 분석
- 티케팅, 유튜브 크롤링도 마찬가지 :)



# Good Bye

See you next time