



연예인 사진 모으기 프로젝트


2. 연예인 사진 모으기 프로젝트

2.1 카카오 오픈 API 사용 준비

- 카카오 개발자 사이트 가입
- OPEN API 사용을 위한 Auth Key 발급

Kakao Developers

Start with kakao platform + service api products Start with kakao platform + service api products 인간은 바로 사람과 사람이 서로 기대어 구성됩니다. 상생은

 <http://developers.kakao.com>

kakao developers

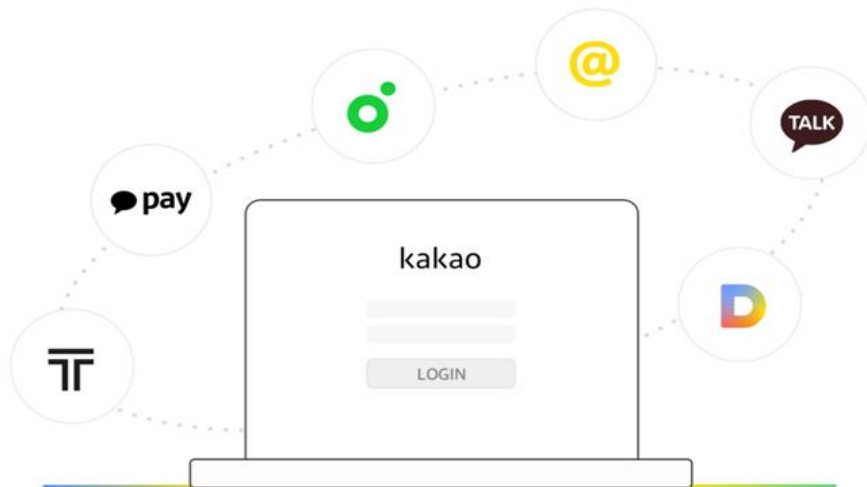
Create `{/}` with Kakao Developers

시작하기

추천 제품

카카오계정 하나로 충분합니다.

카카오 서비스 뿐 아니라 Daum, Melon 및 다양한 외부 서비스까지
카카오계정 하나로 편리하게 이용해보세요!



kakao

카카오메일 아이디, 이메일, 전화번호

비밀번호

☒ 로그인 상태 유지

로그인

또는

QR코드 로그인

회원가입

카카오계정 | 비밀번호 찾기

회원가입

카카오디벨로퍼스의 모든 기능을 사용하기 위해 서비스 이용 약관에 동의해주세요.



서비스 이용 동의

☒

 [필수] 서비스 이용 정책에 대한 동의

플랫폼 서비스 약관

제1조 목적

이 약관은 (주)카카오(이하 “카카오”라 합니다)가 웹 서비스, 모바일 애플리케이션, 기타 응용 프로그램(이하 “프로그램 등”이라고 합니다)을 개발하는데 필요한 플랫폼과 카카오의 다양한 서비스와 연계할 수 있는 부가적인 기능을 제공하는 웹사이트(<https://developers.kakao.com>, 이하 “개발자 웹사이트”라 합니다)를 회원이 이용하기 위한 조건 및 절차에 관한 사항과 기타 필요한 사항을 규정함을 목적으로 합니다.

제2조 정의

① “서비스”라 함은 플랫폼API, 서비스API 등 회원의 프로그램 등 개발 지원을 위해 카카오가 개발자 웹사이트를 통해

☒

 [필수] 개인정보 수집 및 이용에 대한 동의

서비스 내 이용자 식별 및 회원 관리	카카오계정(이메일 또는 카카오톡 전화번호), 아이디, 프로필 사진, 이름	회원탈퇴 후 지체없이 삭제
서비스 내 이용자 검색	아이디	회원탈퇴 후 지체없이 삭제

카카오계정 정보

카카오계정

class@class.com

[다른 카카오계정으로 가입하기](#)

개발자 이름

홍길동

3/45

취소

회원가입



Auth Key 생성을 위한 Application 생성

내 애플리케이션

전체 애플리케이션 (1)

애플리케이션 이름



애플리케이션 추가하기



python-starter

ID 752113

OWNER

내 애플리케이션

전체 애플리케이션

+

APP

py

ID

애플리케이션 추가하기

앱 아이콘

이미지
업로드

파일 선택

JPG, GIF, PNG

권장 사이즈 128px, 최대 250KB

앱 이름

python-starter

사업자명

없음

- 입력된 정보는 사용자가 카카오 로그인을 할 때 표시됩니다.
- 정보가 정확하지 않은 경우 서비스 이용이 제한될 수 있습니다.

취소

저장

내 애플리케이션

전체 애플리케이션 (1)



애플리케이션 추가하기



python-starter

ID 752113

OWNER

REST API 키 값을 메모

APP

python-starter



ID 752113

OWNER

앱 키

네이티브 앱 키

386e81

REST API 키

137b1f

JavaScript 키

17b97f

Admin 키

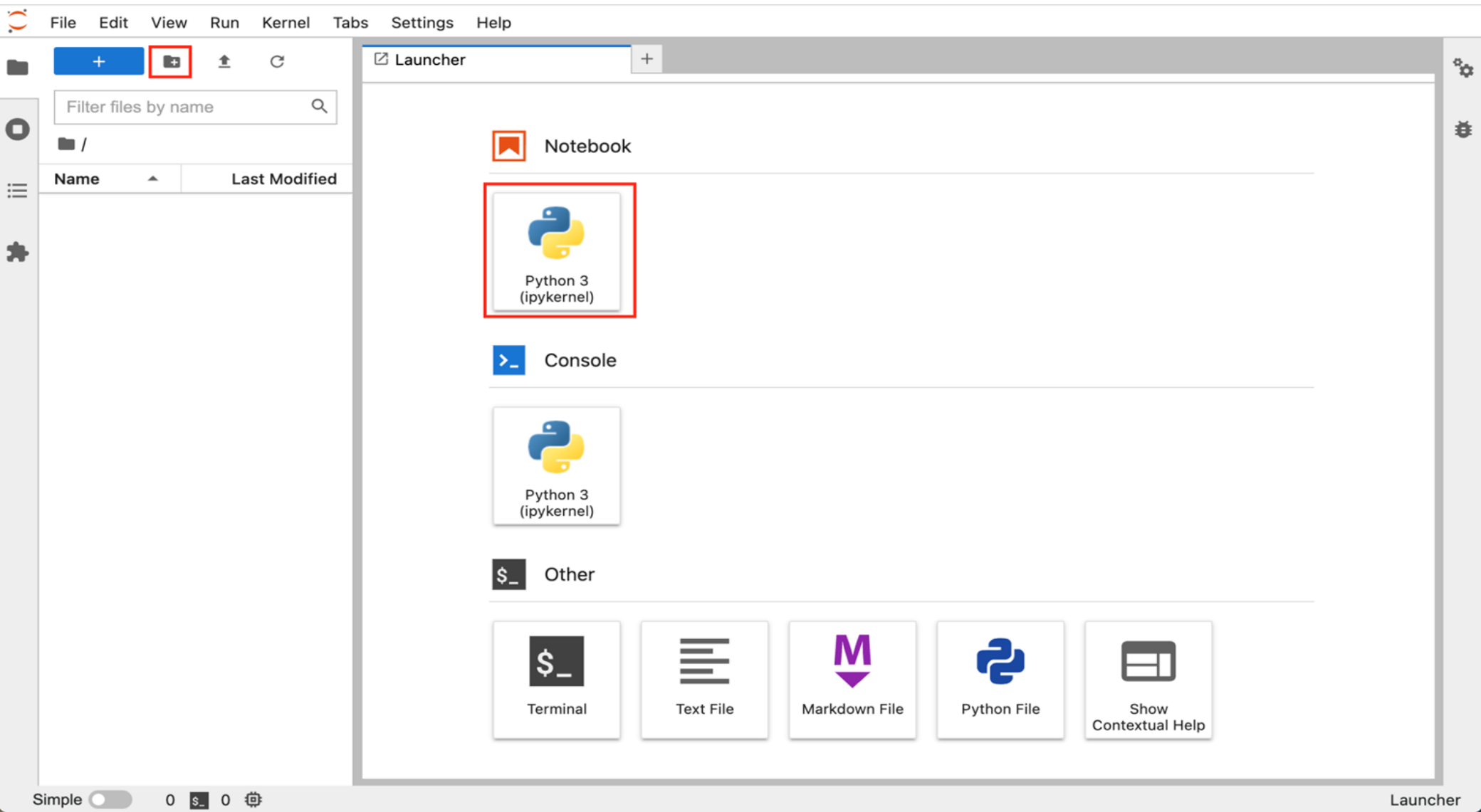
5c288c

플랫폼

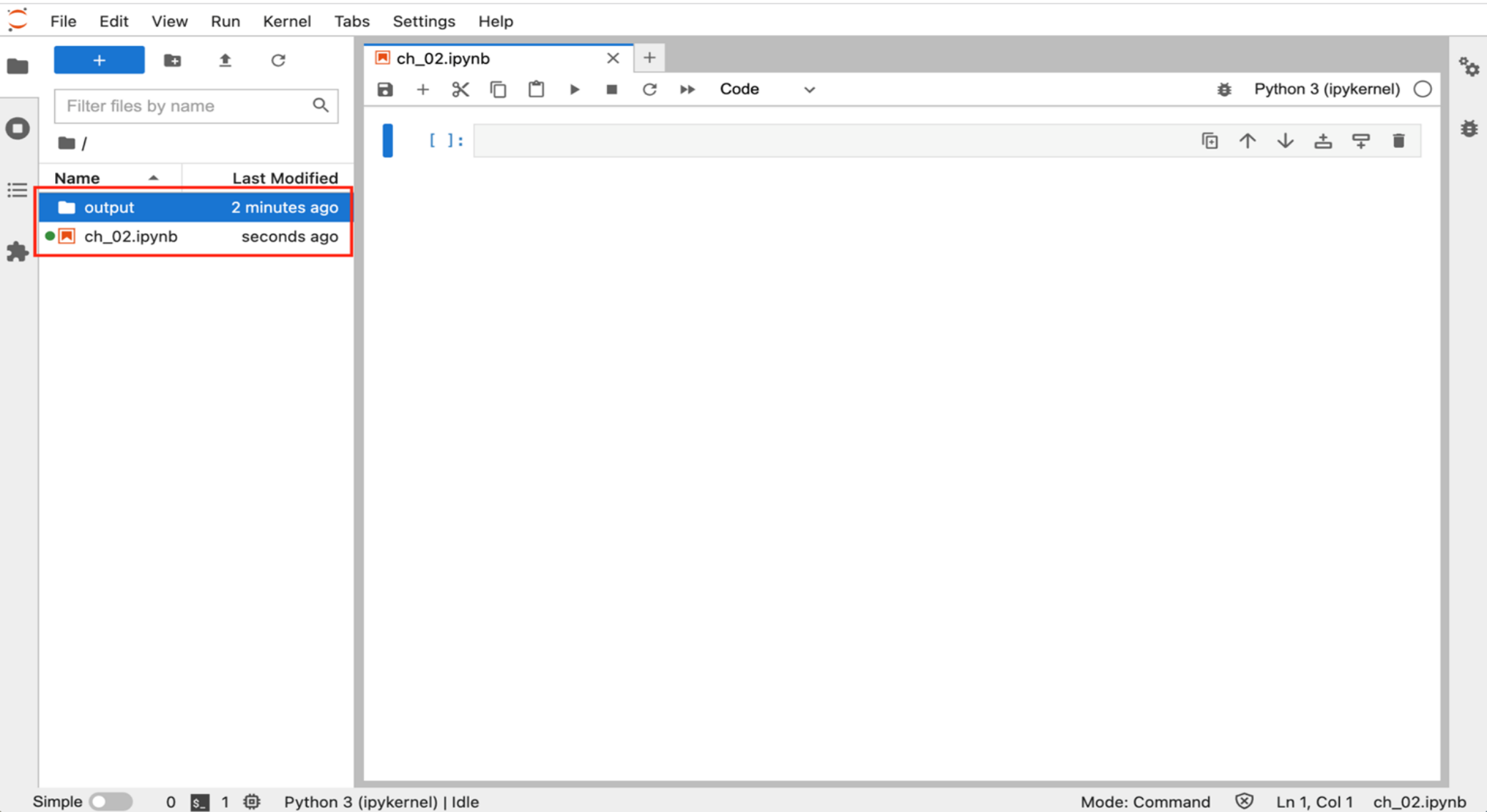
보통의 플랫폼 정보는 앱 키를 생성할 때 지정된 플랫폼에 따라 달라집니다.

2.2 파일 쓰고 읽기

read와 write 학습하기



- 실습을 위한 output 폴더와 ch_02.ipynb 파일 생성
- output 폴더 안에 hello.txt 파일 저장하기



•파일쓰기(write)

```
data = 'hello'
with open('./output/hello.txt','w') as fp :
    fp.write(data)
```

ch_02.ipynb hello.txt

1 hello

+

+

↑

↺

Filter files by name



/

Name



Last Modified

ch_02.ipynb

hello.txt



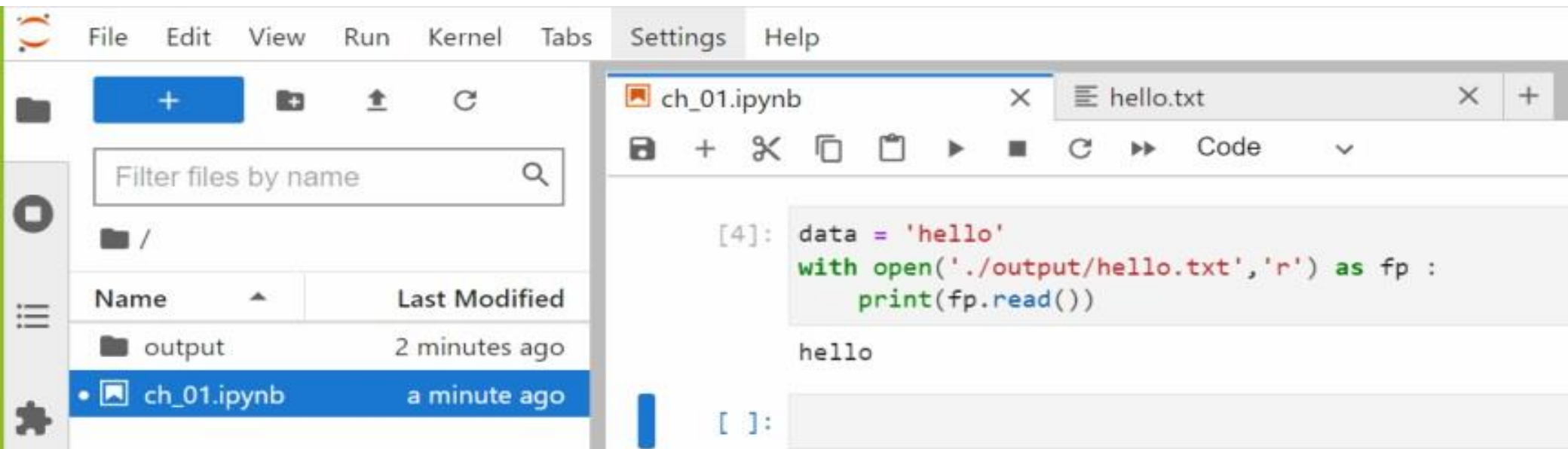
Code



```
[16]: data = 'hello'
      with open('./output/hello.txt','w') as fp :
          fp.write(data)
```

•파일읽기(read)

```
data = 'hello'  
with open('./output/hello.txt','r') as fp :  
    print(fp.read())
```



웹에 있는 이미지 파일을 저장

- get_image(url, FileName) 함수 만들기
- 내용> url(http) 을 주고 test1와 test2를 만들기
- URL
https://search1.kakaocdn.net/argon/600x0_65_wr/ImZk3b2X1w8 을 브라우저에 복사해서 내용을 확인한다.
- <https://img.hankyung.com/photo/201912/01.21302999.1.jpg>
- 실습하기

웹에 있는 이미지 파일을 저장

```
import requests

def get_image(url, FileName) :
    img_response = requests.get(url)

    if img_response.status_code == 200 :
        #print(img_response.content)
        print("=" * 30 + "[이미지 저장]" + "=" * 30)
        file = "./output/" + FileName
        with open(file, "wb") as fp :
            fp.write(img_response.content)

url = "https://search1.kakaocdn.net/argon/600x0_65_wr/lmZk3b2X1w8"
get_image(url, "펍수1.jpg")
url = "https://img.hankyung.com/photo/201912/01.21302999.1.jpg"
get_image(url, "펍수2.jpg")
```


+

Filter files by name

/ output /

Name	Last Modified
펭수1.jpg	3 minutes ago
펭수2.jpg	3 minutes ago
hello.txt	14 minutes ago
text_1.jpg	34 minutes ago
text_10.jpg	34 minutes ago
text_11.jpg	34 minutes ago
text_12.jpg	34 minutes ago
text_13.jpg	34 minutes ago
text_14.jpg	34 minutes ago
text_15.jpg	34 minutes ago
text_16.jpg	34 minutes ago
text_17.jpg	34 minutes ago
text_18.jpg	34 minutes ago
text_19.jpg	34 minutes ago
text_2.jpg	34 minutes ago
text_20.jpg	34 minutes ago
text_21.jpg	34 minutes ago
text_22.jpg	34 minutes ago
text_23.jpg	34 minutes ago
text_24.jpg	34 minutes ago
text_25.jpg	34 minutes ago

ch_02.ipynb

hello.txt

Untitled.ipynb

Untitled1.ipynb

+

✂

📄

▶

■

↺

↻

▶▶

Code

▼

```

[1]: import requests

def get_image(url, FileName) :
    img_response = requests.get(url)

    if img_response.status_code == 200 :
        #print(img_response.content)
        print("=" * 30 + "[ 이미지 저장 ]" + "=" * 30)
        file = "./output/" + FileName
        with open(file, "wb") as fp :|
            fp.write(img_response.content)

url = "https://search1.kakaocdn.net/argon/600x0_65_wr/ImZk3b2X1w8"
get_image(url, "펭수1.jpg")
url = "https://img.hankyung.com/photo/201912/01.21302999.1.jpg"
get_image(url, "펭수2.jpg")

===== [ 이미지 저장 ] =====
===== [ 이미지 저장 ] =====

```

펭수1.jpg

✕

+

카카오 이미지 검색 Open API 호출하기

```
import requests
import json

url = 'https://dapi.kakao.com/v2/search/image'
headers = {
    "Authorization": "KakaoAK <REST API Key>"
}
data = {
    "query": "펭수"
}

response = requests.post(url, headers=headers, data=data)
if response.status_code != 200 :
    print("error !!! because ", response.json())
else :
    count = 0

    for image_info in response.json()['documents'] :
        print(f"[{count}th] image_url =", image_info['image_url'])
        count = count + 1
```

카카오 이미지 검색 Open API 호출하기

- `res.request` # 내가 보낸 request 객체에 접근 가능
- `res.status_code` # 응답 코드
- `res.raise_for_status()` # 200 OK 코드가 아닌 경우 에러 발동
- `res.json()` # json response일 경우 딕셔너리 타입으로 바로 변환

전체 코드

```
import requests
import json

def save_image(url, FileName) :
    img_response = requests.get(url)

    if img_response.status_code == 200 :
        #print(img_response.content)
        print("=" * 30 + "[이미지 저장]" + "=" * 30)
        file = "./output/" + FileName
        with open(file, "wb") as fp :
            fp.write(img_response.content)

url = 'https://dapi.kakao.com//v2/search/image'
headers = {
    "Authorization" : "KakaoAK <REST API Key>"
}
```

```
data = {
    "query" : "펭수"
}

response = requests.post(url, headers=headers, data=data)
if response.status_code != 200 :
    print("error !!! because ", response.json())
else :
    count = 0
    for image_info in response.json()['documents'] :
        print(f"[{count}th] image_url =", image_info['image_url'])
        count = count + 1
        file_name = "test_%d.jpg" %(count)
        save_image(image_info['image_url'], file_name)
```

실행화면

The screenshot displays a Jupyter Notebook environment with two main components: a file browser on the left and a code editor on the right.

File Browser (Left Panel):

- Search bar: "Filter files by name"
- Current directory: "/ output /"
- Table of files:

Name	Last Modified
hello.txt	2 hours ago
text_1.jpg	a minute ago
text_10.jpg	a minute ago
text_11.jpg	a minute ago
text_12.jpg	a minute ago
text_13.jpg	a minute ago
text_14.jpg	a minute ago
text_15.jpg	a minute ago
text_16.jpg	a minute ago
text_17.jpg	a minute ago
text_18.jpg	a minute ago
text_19.jpg	a minute ago
text_2.jpg	a minute ago

Code Editor (Right Panel):

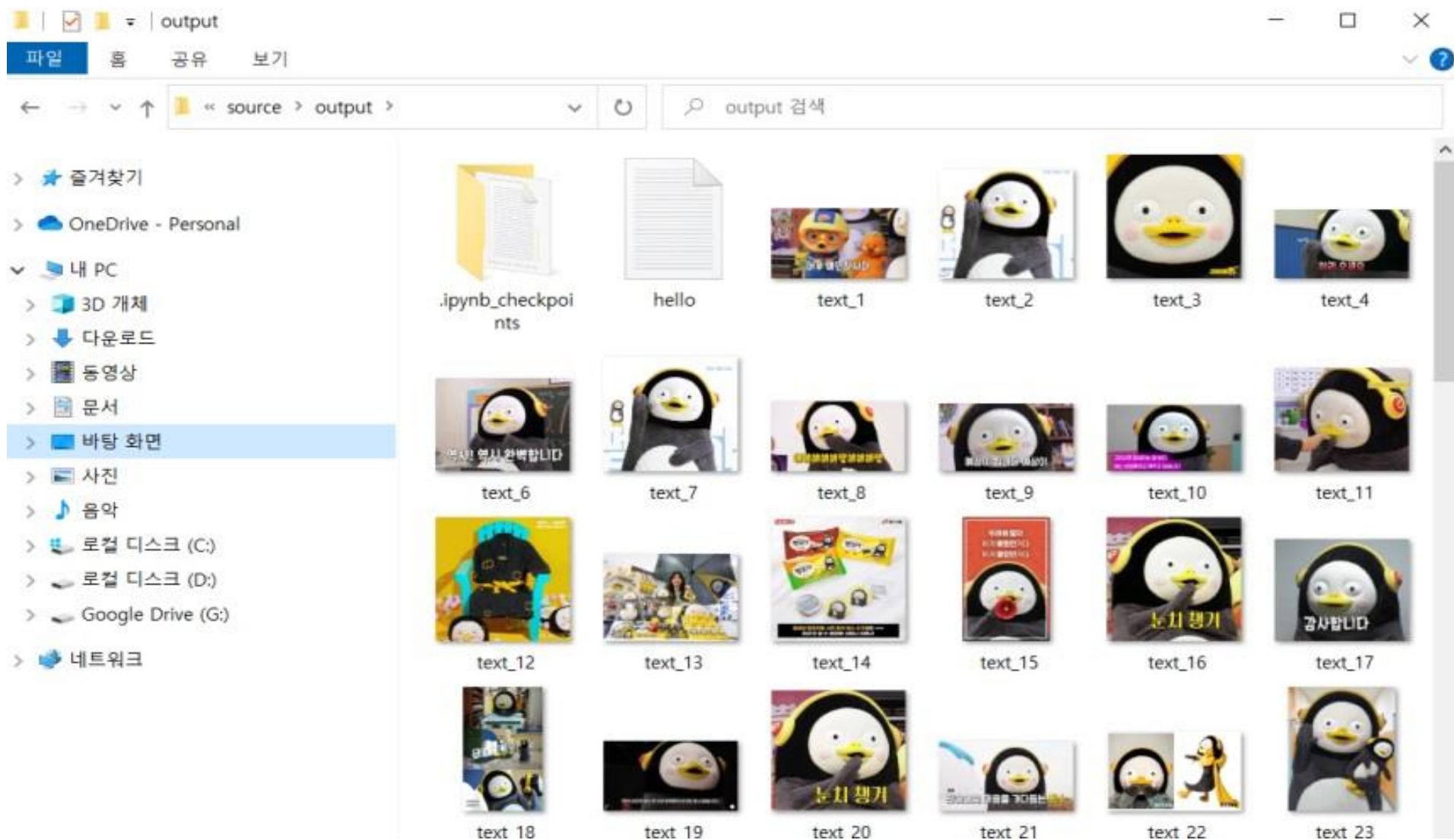
- Open files: ch_01.ipynb, hello.txt, Untitled.ipynb
- Language: Python 3 (ipykernel)
- Code cell content:

```
file_name='text_%d.jpg'%(count)
save_image(image_info['image_url'],file_name)
```

The code cell shows a loop that collects image URLs and saves them as individual image files. The output of the code cell is as follows:

```
[0th] image_url = https://postfiles.pstatic.net/MjAyMDAzMDVfMTc5/MDAxNTgzMzkxNDc5MTA1.6ovQ3UEi00GVp082npMADv
CJAYad07-0K0CV9J9macwg.sbxr2lp2j2SU2pEu14Ch1sbsqbiwEtXRfzMefmFv1Ukg.JPEG.hyungee9115/IMG_6597.JPG?type=w966
===== [이미지 저장] =====
[1th] image_url = https://pds.joins.com/news/component/joongang_sunday/201912/21/904b917f-c6e5-44fa-854f-d67
5bc869b9f.jpg
===== [이미지 저장] =====
[2th] image_url = https://image.genie.co.kr/Y/IMAGE/IMG_MUZICAT/IV2/Genie_Magazine/8311/Mgz_Sub_IMG_20200421
170614.png
===== [이미지 저장] =====
[3th] image_url = https://k.kakaocdn.net/dn/zW6mm/btqAFoSsOVu/bzpztA0Emc8Xr5bc421rXK/img.jpg
===== [이미지 저장] =====
[4th] image_url = http://www.newscj.com/news/photo/202006/newscj_%EC%B2%9C%EC%A7%80%EC%9D%BC%EB%B3%84_749091
_779664_459.jpg
[5th] image_url = https://postfiles.pstatic.net/MjAyMTEwMDZfMjI5/MDAxNjMzNDg5ODY0NTQ3.ZEpnGPgcZBKPTJPUYpi-GM
piHqGwcMZVBpzd4K0cNqsg.HfKdq_p3Jach6X2-9p5EeD5xj278-aeIuXGUyji3dZgg.JPEG.hwit21/IMG_0193.JPG?type=w773
===== [이미지 저장] =====
[6th] image_url = https://t1.daumcdn.net/news/201912/23/mk/20191223171504726muhi.jpg
===== [이미지 저장] =====
[7th] image_url = http://t1.daumcdn.net/cafeattach/1YWEt/3472eb116fa3fc4fbb3637faaf77bc1f56dc9c95
===== [이미지 저장] =====
```

결과화면



과제

- 연예인 이름으로 검색하기
- 파일은 연예인_1.jpg 형식으로 저장하기
- 캡처하여 제출하기

Thank you
