Contents

1. Vision Api 이용을 위한 사전 작업
2. 원격 이미지의 얼굴 감지
3. 로컬 이미지의 얼굴 감지
4. 에러 해결

**(1) Vision api 이용을 위한 사전 작업**

먼저

<https://console.developers.google.com/apis/api/vision.googleapis.com/>

이 링크로 들어가 api 사용 설정을 클릭합니다.

* 사용이 시작됩니다.

다음,

구글 클라우드 sdk(=gcloud)을 설치해야하는데

<https://cloud.google.com/sdk/docs/quickstart-linux>

이거 보고 하시면 됩니다.

이걸 깔면 자기가 사용하는 운영체제에서 gcloud명령어를 사용할 수 있습니다.

그다음 터미널에서

gcloud auth login

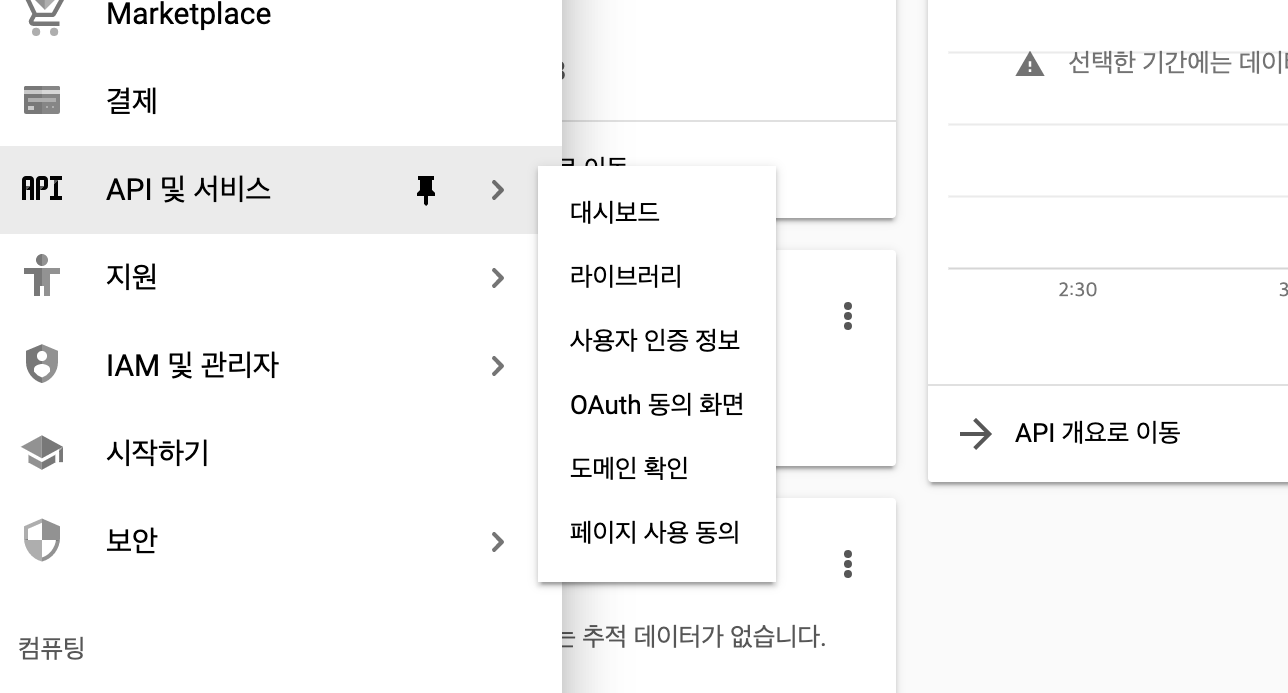
이라고 적으면 브라우저가 열리면서 구글 계정 선택하라는 화면이 나오는데

자기가 프로젝트 만든 계정을 클릭해서 자기 컴퓨터랑 해당 계정의 구글 클라우드 연결해주면 됩니다.

다 하셨으면,

<https://console.cloud.google.com/home/dashboard?project=big-involution-254610&_ga=2.245753028.-1124793354.1570100987&_gac=1.117297524.1570131295.Cj0KCQjwuNbsBRC-ARIsAAzITuetXLsV6r88MbO-VF4sD-TzWhDpKRJyPrRWDt-tBfHJarQHJz-eVU8aAjc1EALw_wcB>

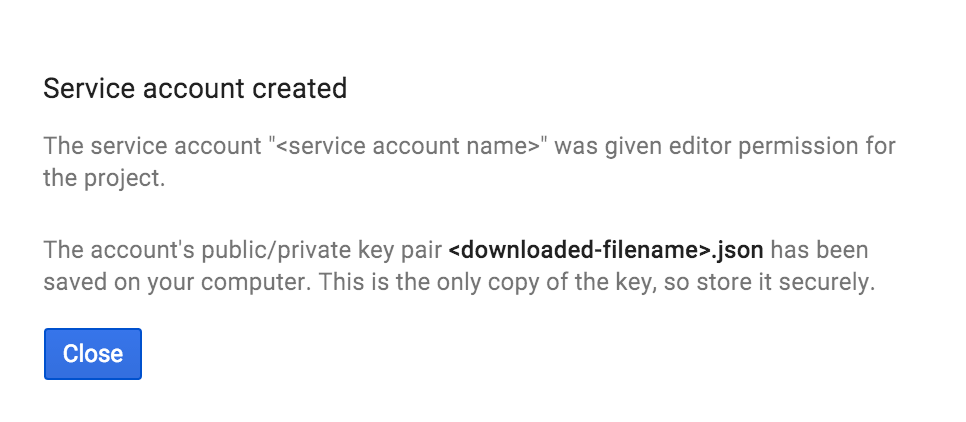
이 링크로 들어가서 api 및 서비스를 클릭 -> 대시보드로 들어가서



화면 왼쪽 메뉴를 보면 열쇠모양(사용자 인증 정보) 메뉴가 있습니다. 클릭. 여기서 [사용자 인증 정보 만들기 – 서비스 계정 키]를 선택합니다.

새로운 서비스 계정의 이름과 ID를 입력하고

Key type 은 JSON 으로 선택합니다. 서비스 계정이 생성되면 downloaded-filename.json 링크 눌러서 JSON 파일을 다운로드 받아둡니다. 이 파일은 구글이 제공하는 API에서 사용할 인증 파일이므로 자신의 서버에 올려두고 사용해야 합니다.



저는 제 mac에서 실험했기때문에 /Users/mac/Desktop/google-cloud-python/renovate.json 위치에 저장했는데

라즈베리파이에서 사용할 때는 라즈베리 파이에 원하는 경로에 올려놓고 사용하면 됩니다.

Ex ) /home/pi/temp/vision/gac.json

그리고 bashrc에 들어가서 환경변수를 지정해줍니다.

export GOOGLE\_APPLICATION\_CREDENTIALS= jason파일의 경로

export GOOGLE\_CLOUD\_SDK\_PATH=구글 클라우드가 깔린 경로

저 같은 경우는

export GOOGLE\_APPLICATION\_CREDENTIALS= /Users/mac/Desktop/google-cloud-python/renovate.json

export GOOGLE\_CLOUD\_SDK\_PATH=/Users/mac/Desktop/google-cloud-sdk

일케 올려놨습니다…

마지막으로 터미널에서 gcloud auth application-default login 을 해주고

디폴트 계정 연결을 해줍니다.

그다음에

파이썬을 설치해줍니다.

찾아보니 구글 클라우드 비전은 파이선 2,3 둘다 가능하지만 파이썬 2는 2020년 1월에 종료되므로 3을 씁시다.

근데 전 처음에 2 깔아놔서 그냥 2 계속 썼습니다.

그다음

구글 클라우드비전에서

파이썬 버전을 설치해야하는데

터미널에서

sudo pip install google-cloud

sudo pip install google-cloud-vision

이렇게 적으면 설치가 됩니다…근데 이건 파이썬 2버전이고

python3은 터미널에서

sudo pip3 install google-cloud

sudo pip3 install google-cloud-vision

이라고 써주면 됩니다.

이렇게 해서 환경설정은 끝났습니다.

**(2) 원격 이미지의 얼굴 감지**

클라우드 비전에서 이미지를 불러오는데에는 두가지 방법이 있습니다.

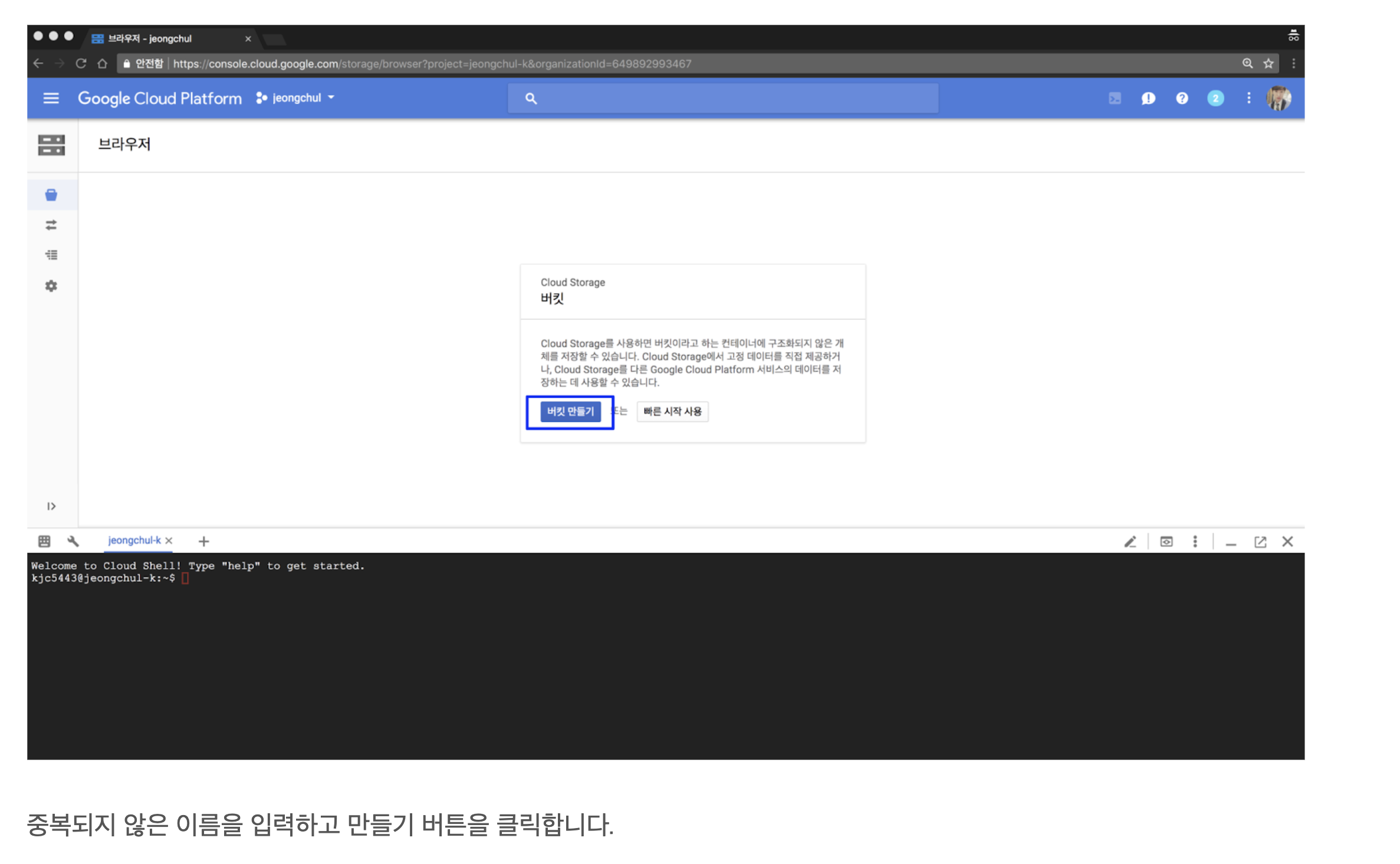
-구글 클라우드 버켓에 이미지를 업로드 한 뒤, 그 이미지를 인식

-로컬에 있는 이미지를 불러와서 인식

먼저 클라우드 버켓에 이미지를 업로드 해서 인식하는 방법입니다.

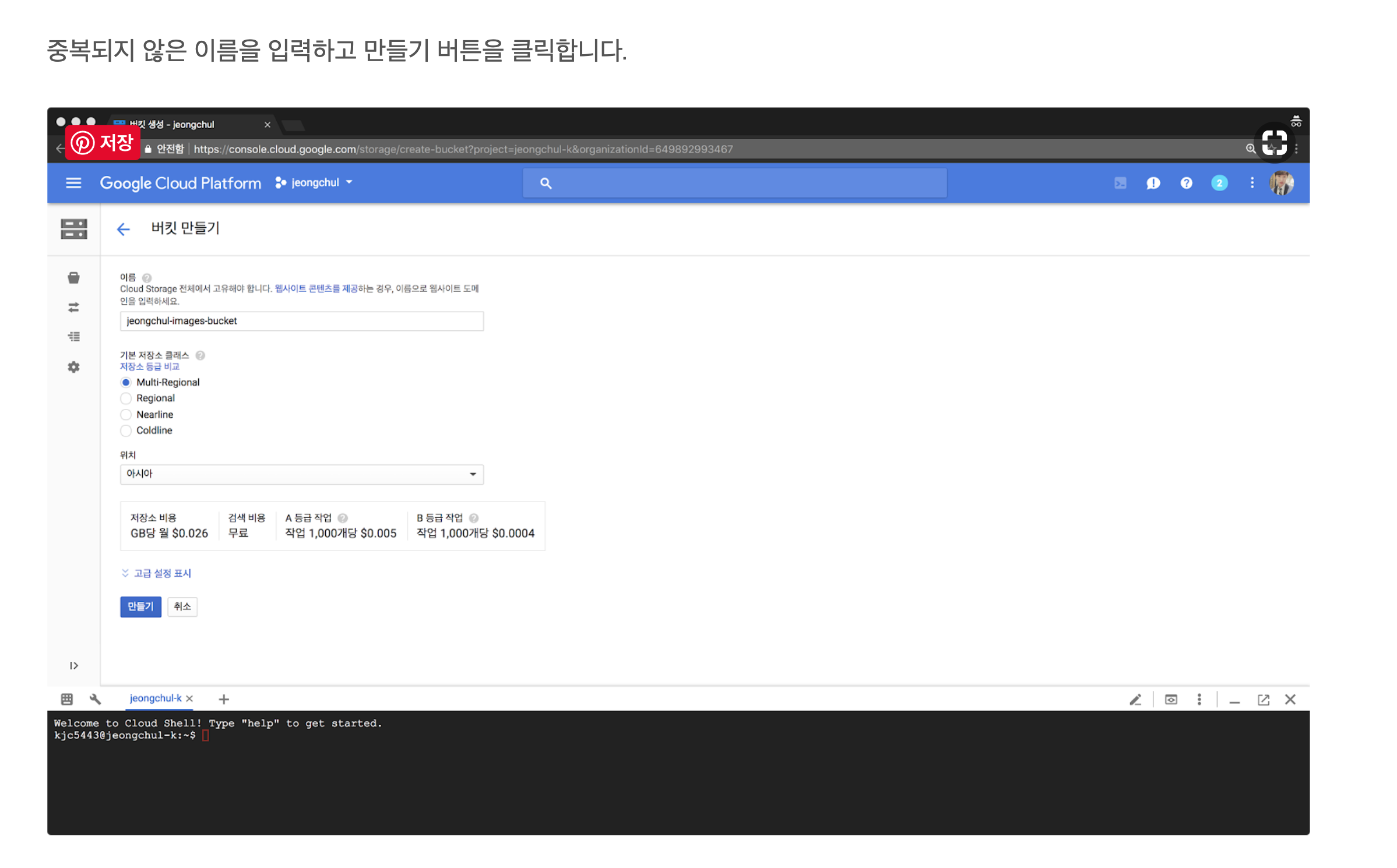
클라우드 버킷에 이미지를 업로드 하는 법 :

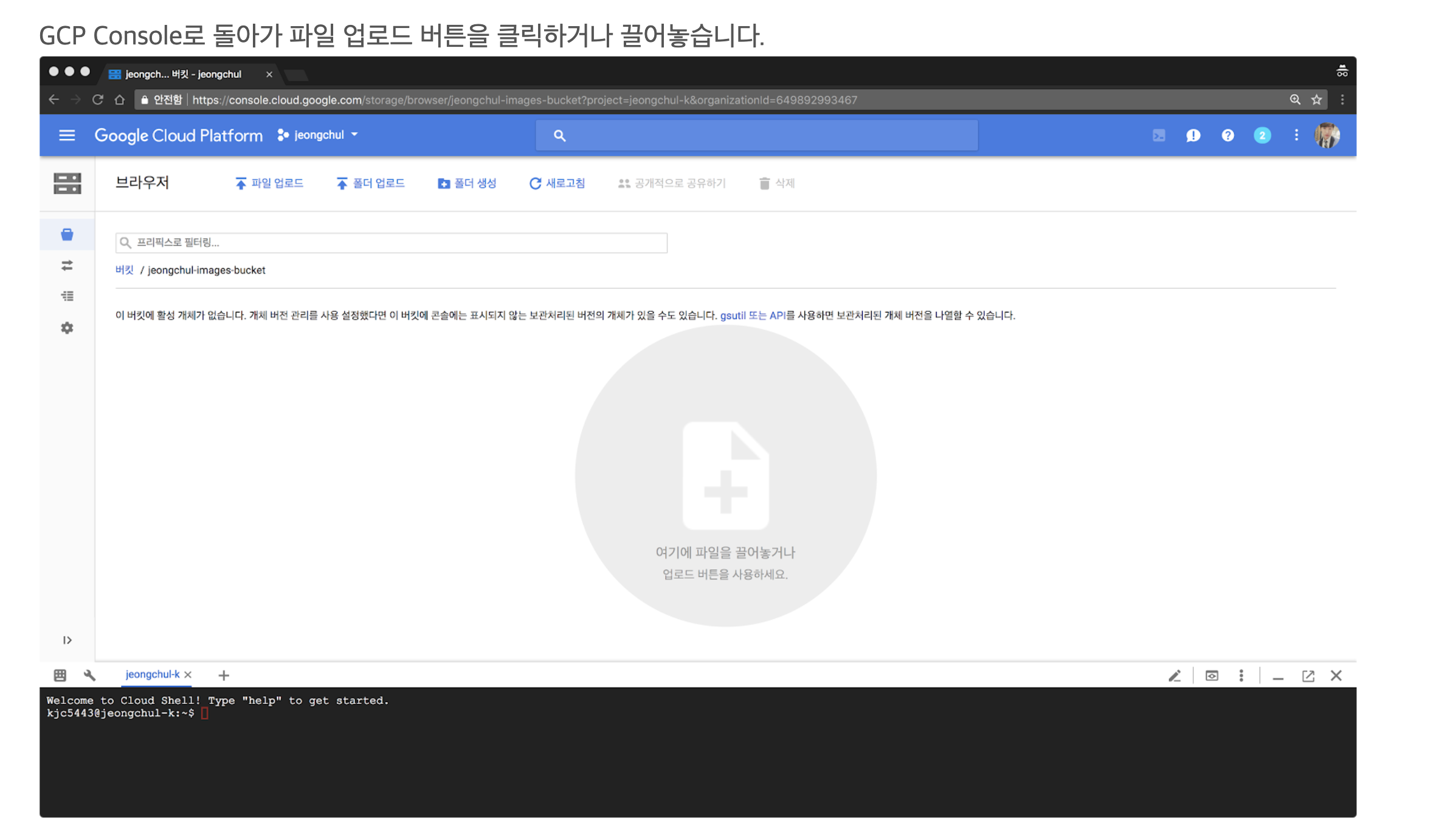


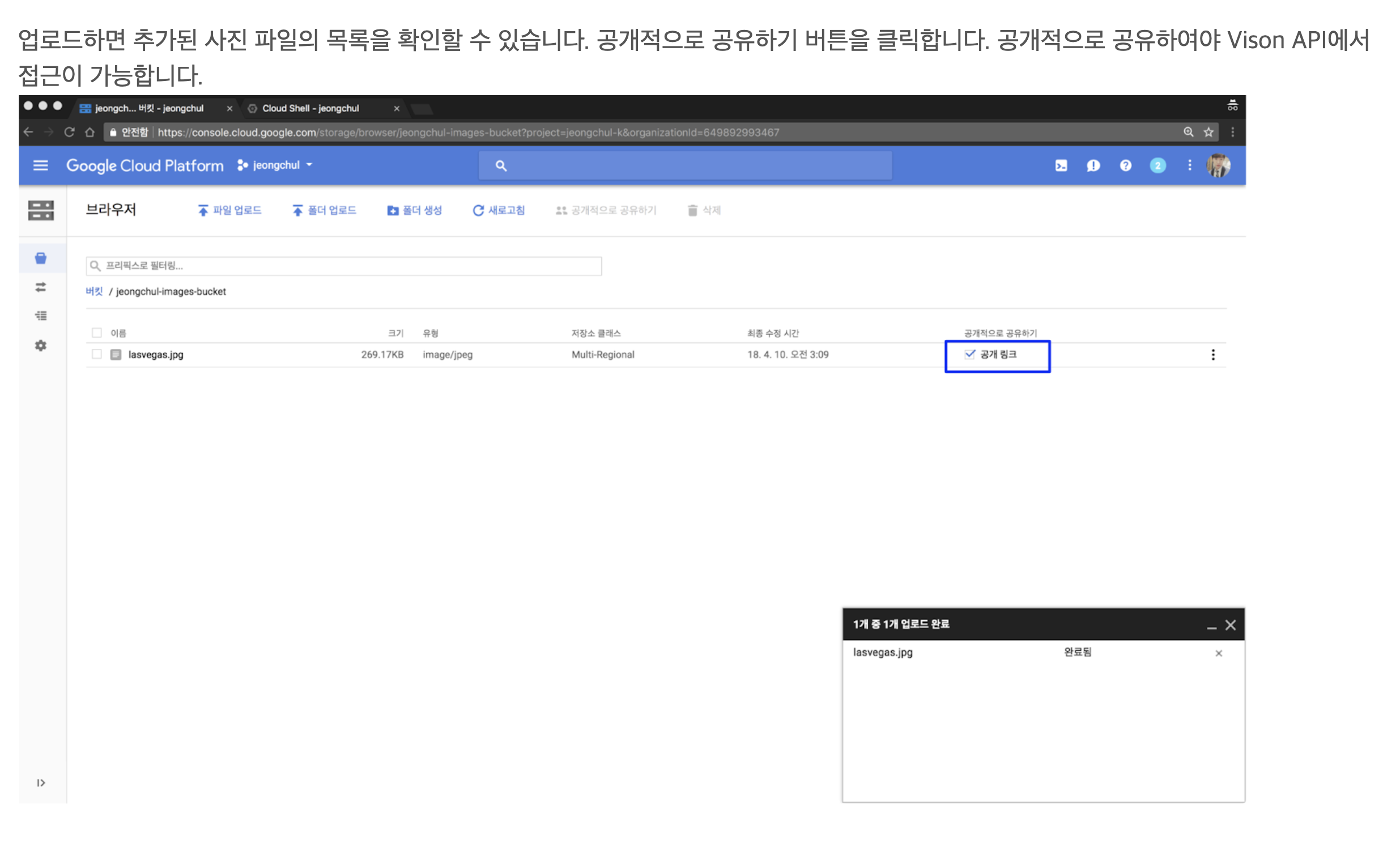


이름은 내 계정내에서만 중복되지 않으면 되는게 아니라

전 세계에서 중복되지 않는 이름이어야합니다.





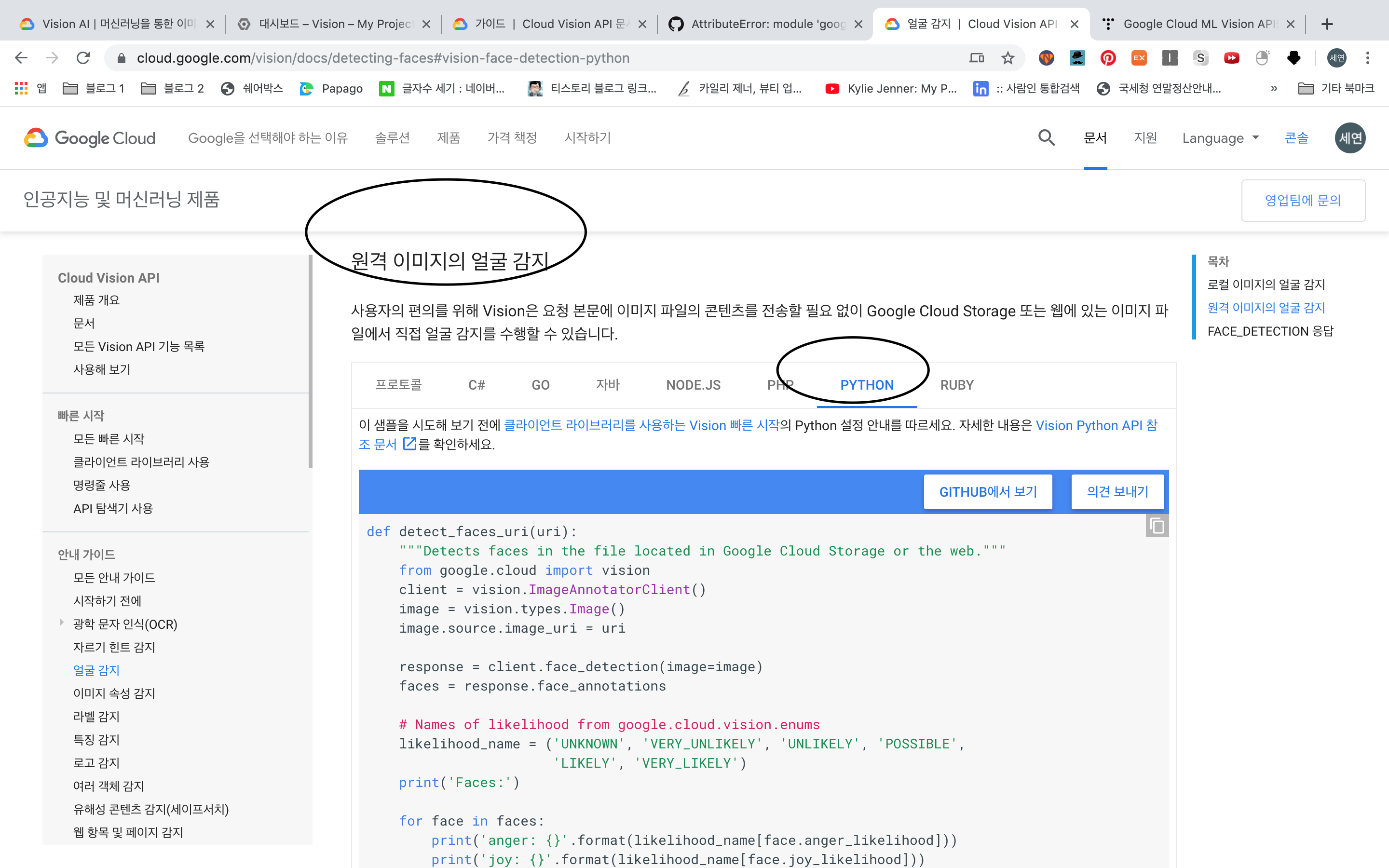


이렇게 올리면 됩니다..

이상태에서 에디터로 파이썬 코드를 복사해옵니다.

<https://cloud.google.com/vision/docs/detecting-faces#vision-face-detection-python>

이 링크로 들어가서



이 코드를 복사해옵니다.

그리고 에디터로 옮긴다음

마지막줄에

detect\_face\_run(gs://버킷 이름/ 버킷에 이름 이미지 명)

이렇게 작성하면 해당 코드가 표정을 읽어옵니다.

예시)

def detect\_faces\_uri(uri):

    """Detects faces in the file located in Google Cloud Storage or the web."""

    from google.cloud import vision

    client = vision.ImageAnnotatorClient()

    image = vision.types.Image()

    image.source.image\_uri = uri

    response = client.face\_detection(image=image)

    faces = response.face\_annotations

    # Names of likelihood from google.cloud.vision.enums

    likelihood\_name = ('UNKNOWN', 'VERY\_UNLIKELY', 'UNLIKELY', 'POSSIBLE',

                       'LIKELY', 'VERY\_LIKELY')

    print('Faces:')

    for face in faces:

        print('anger: {}'.format(likelihood\_name[face.anger\_likelihood]))

        print('joy: {}'.format(likelihood\_name[face.joy\_likelihood]))

        print('surprise: {}'.format(likelihood\_name[face.surprise\_likelihood]))

        vertices = (['({},{})'.format(vertex.x, vertex.y)

                    for vertex in face.bounding\_poly.vertices])

        print('face bounds: {}'.format(','.join(vertices)))

detect\_faces\_uri('gs://face\_detection\_seyeon/test.jpg')

결과값 :

Faces:

anger: VERY\_UNLIKELY

joy: VERY\_UNLIKELY

surprise: VERY\_UNLIKELY

face bounds: (250,108),(454,108),(454,345),(250,345)

사용한 사진 :



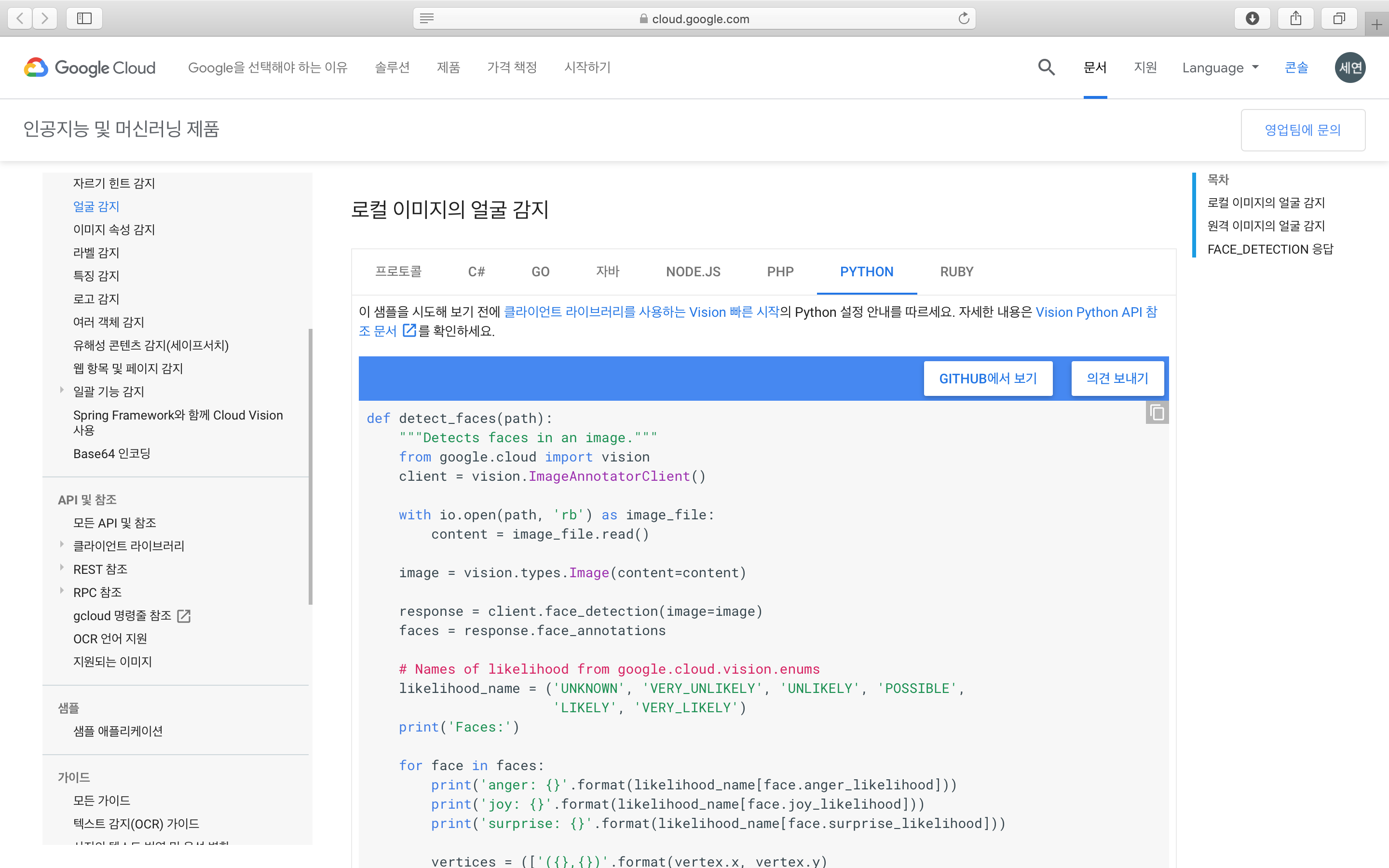
**(3) 로컬 이미지의 얼굴 감지**

다음으로 pc에 저장된 이미지에서 얼굴을 감지하는 방법입니다.

<https://cloud.google.com/vision/docs/detecting-faces#vision-face-detection-python>

이 링크로 들어가서

원격 이미지의 얼굴 감지 위에 있는 로컬 이미지의 얼굴감지의 파이썬 코드를 긁어옵니다.



그다음 에디터에 넣고

마지막에 detect\_faces(이미지 파일의 절대경로!!!)

를 적어주고 실행하면 됩니다.

예시)

import io

def detect\_faces(path):

    """Detects faces in an image."""

    from google.cloud import vision

    client = vision.ImageAnnotatorClient()

    with io.open(path, 'rb') as image\_file:

        content = image\_file.read()

    image = vision.types.Image(content=content)

    response = client.face\_detection(image=image)

    faces = response.face\_annotations

    # Names of likelihood from google.cloud.vision.enums

    likelihood\_name = ('UNKNOWN', 'VERY\_UNLIKELY', 'UNLIKELY', 'POSSIBLE',

                       'LIKELY', 'VERY\_LIKELY')

    print('Faces:')

    for face in faces:

        print('anger: {}'.format(likelihood\_name[face.anger\_likelihood]))

        print('joy: {}'.format(likelihood\_name[face.joy\_likelihood]))

        print('surprise: {}'.format(likelihood\_name[face.surprise\_likelihood]))

        vertices = (['({},{})'.format(vertex.x, vertex.y)

                    for vertex in face.bounding\_poly.vertices])

        print('face bounds: {}'.format(','.join(vertices)))

detect\_faces('/Users/mac/Desktop/test2.jpg')

결과값

Faces:

anger: VERY\_LIKELY

joy: VERY\_UNLIKELY

surprise: UNLIKELY

face bounds: (166,10),(322,10),(322,191),(166,191)

사용한 사진은

입니다.

**(4) 에러 해결**

파이썬 파일을 실행할 때,

ImportError: cannot import name 'vision' from 'google.cloud' (unknown location)

이런 에러가 계속 떴습니다.

웹에 검색해보니

pip install google-cloud라는 명령어를 통해 클라우드 비전 모듈을 깔아야하는데 깔지 않아서 생긴 오류라고 해서 여러번 명령어를 쳐서 여러번 다시 깔아보았지만 여전히 먹지 않았습니다.

그래서 코드를 요리조리 바꿔보고,

파이썬 3로 도전해보기도 하고

혹시 모듈의 위치가 파이썬이 깔린 위치랑 달라서 그런가 싶어서

(제 mac에선 파이썬은 system에 깔리고 모듈은 library에 깔렸습니다.)

bashrc에 모듈의 path값도 넣어줘보고

아예 google-cloud 모듈을 직접 파이썬이 깔린 위치에 옮겨보기도 했지만 끝까지… 먹지 않았습니다.

포기하는 마음으로

sudo pip uninstall google-cloud라는 명령을 통해 클라우드 모듈을 지웠다가

sudo pip install google-cloud라는 명령어를 통해 다시 깔아보니 되었습니다.

아마 처음 모듈 설치할때 에러가 있었는데

이 오류를 해결하려면 모듈을 한번 지웠다가 다시 깔아야 했던것 같습니다…

(아마 뭐 하나가 안 깔렸던것 같음)

[참고자료]

<https://cloud.google.com/vision/docs/quickstart>