Logic App

AI service with serverless data computing

Microsoft Student Partner
차주연

LOGIC APP

LOGIC APP이란

Logic Apps

Quickly build powerful integration solutions

Connect your business-critical apps and services with Azure Logic Apps, automating your workflows without writing a single line of code.

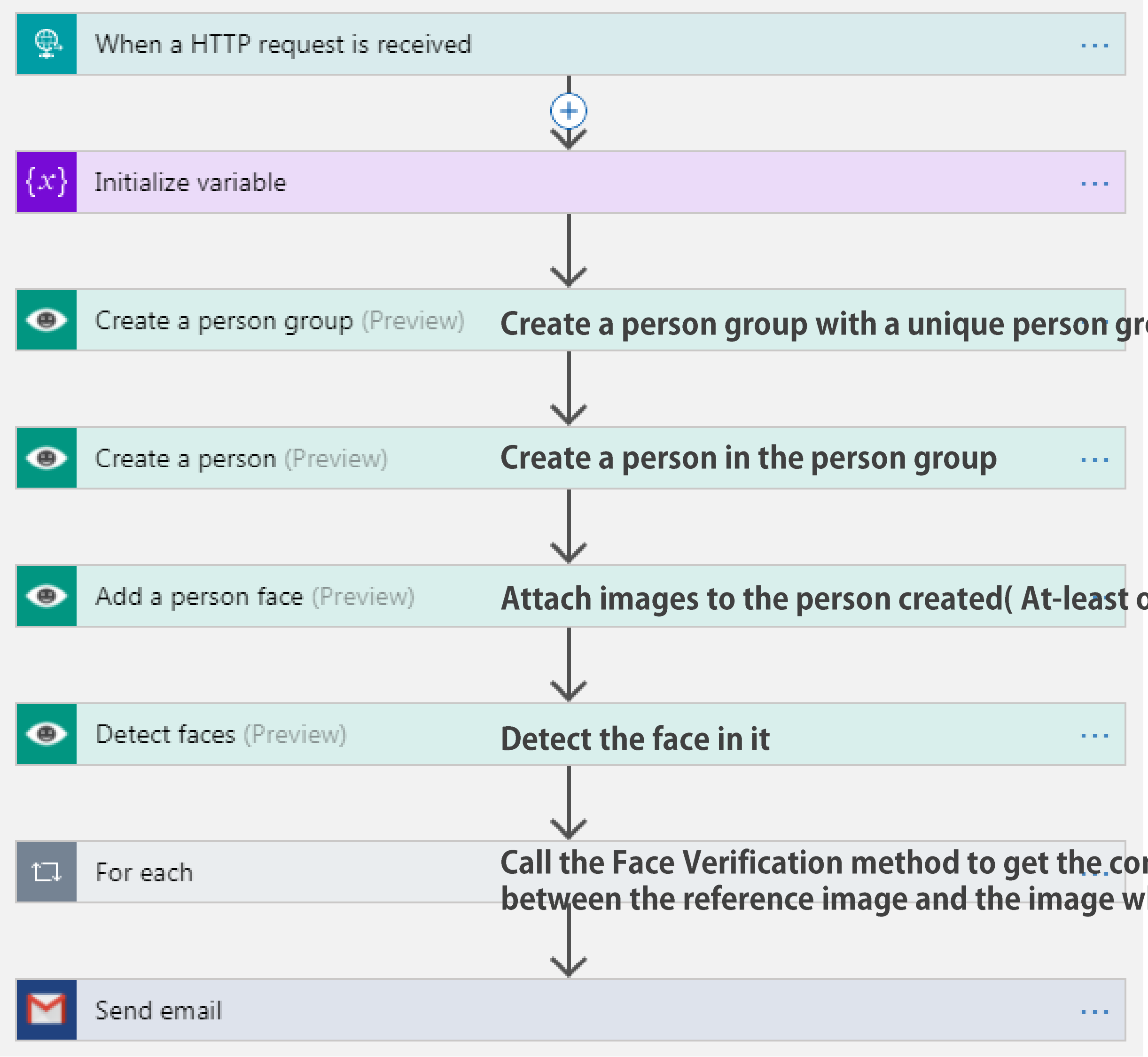
* The advantages of using Logic Apps

1. Easy to use design tools
2. Connect APIs easily
3. Get started quickly from templates
4. Extensibility baked-in
5. Real integration horsepower

참고 : https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-what-are-logic-apps

LOGIC APP

전체 흐름도



When a HTTP request is received

Initialize variable

Create a person group (Preview)

Create a person group with a unique person group id in which reference pictures of the face of person(s) will be stored.

Create a person (Preview)

Create a person in the person group

Add a person face (Preview)

Attach images to the person created(At-least one image per person is required!)

Detect faces (Preview)

Detect the face in it

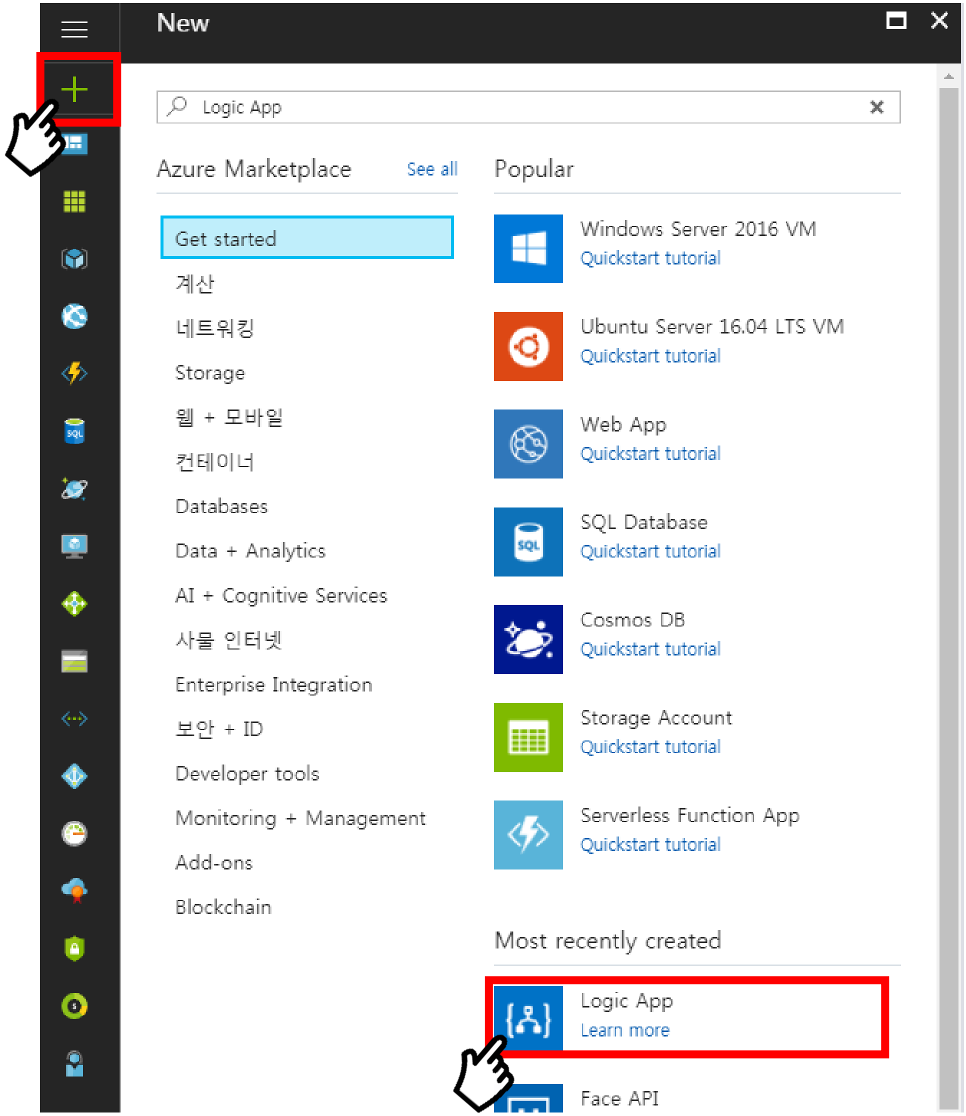
For each

Call the Face Verification method to get the confidence score after the comparison between the reference image and the image which requires verification.

Send email

LOGIC APP

Logic app 생성



New

Logic App

Azure Marketplace — See all

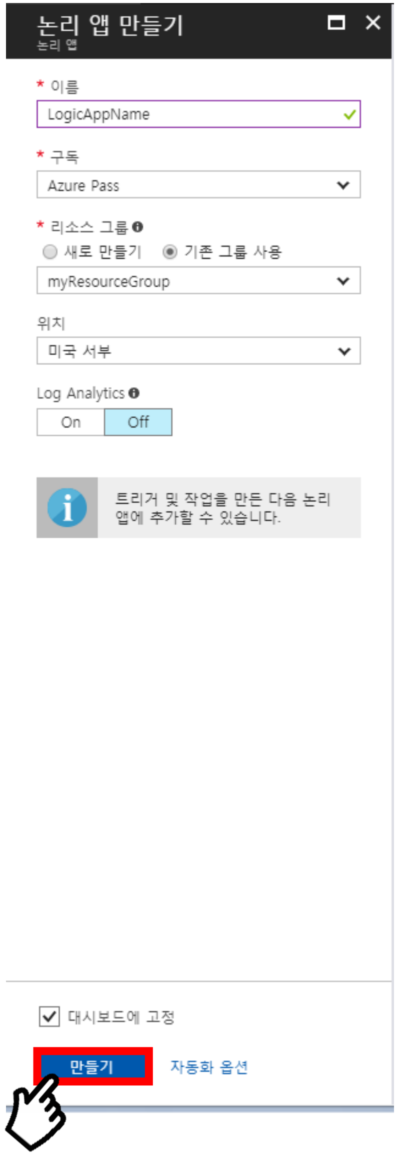
- Get started
- 계산
- 네트워킹
- Storage
- 웹 + 모바일
- 컨테이너
- Databases
- Data + Analytics
- AI + Cognitive Services
- 사물 인터넷
- Enterprise Integration
- 보안 + ID
- Developer tools
- Monitoring + Management
- Add-ons
- Blockchain

Popular

- Windows Server 2016 VM — Quickstart tutorial
- Ubuntu Server 16.04 LTS VM — Quickstart tutorial
- Web App — Quickstart tutorial
- SQL Database — Quickstart tutorial
- Cosmos DB — Quickstart tutorial
- Storage Account — Quickstart tutorial
- Serverless Function App — Quickstart tutorial

Most recently created

- Logic App — Learn more
- Face API



논리 앱 만들기
논리 앱

* 이름: LogicAppName ✓

* 구독: Azure Pass

* 리소스 그룹: ○ 새로 만들기 ● 기존 그룹 사용 — myResourceGroup

위치: 미국 서부

Log Analytics: On [✓ Off]

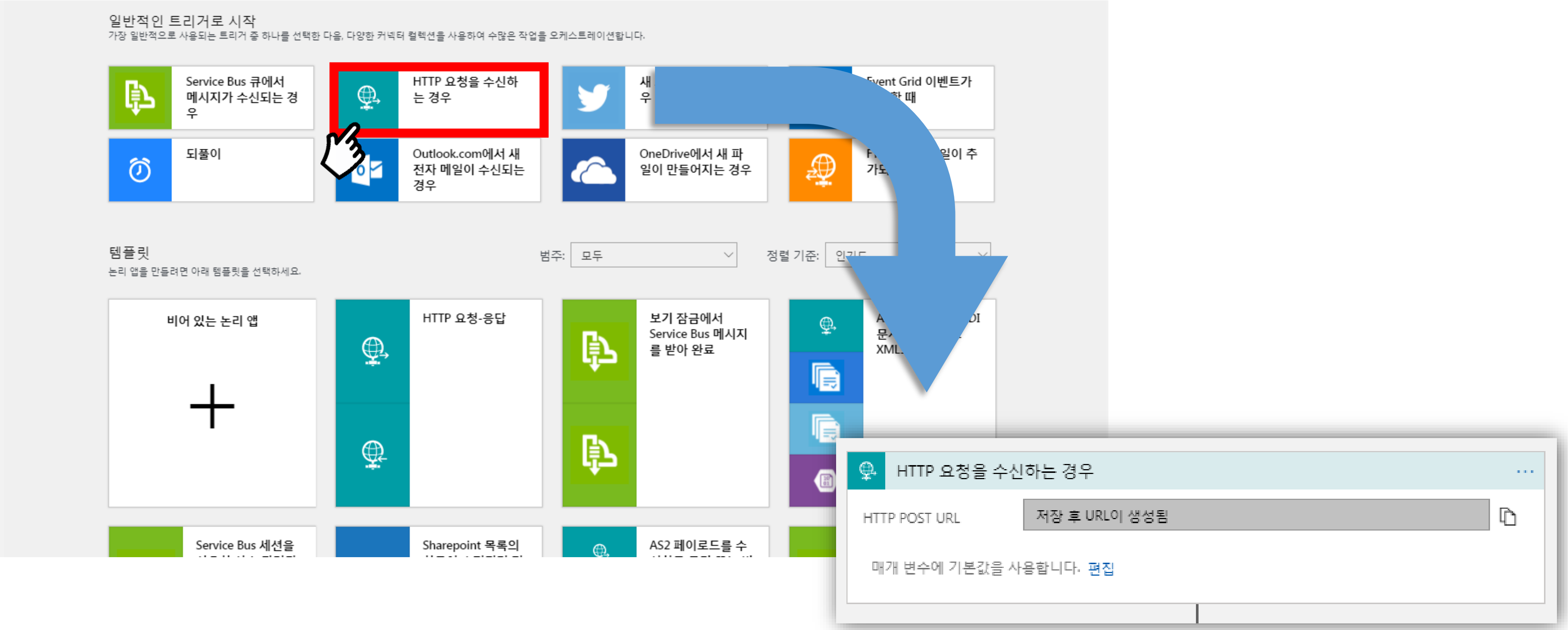
트리거 및 작업을 만든 다음 논리 앱에 추가할 수 있습니다.

☑ 대시보드에 고정

만들기 자동화 옵션

LOGIC APP

Logic app 생성



일반적인 트리거로 시작

가장 일반적으로 사용되는 트리거 중 하나를 선택한 다음, 다양한 커넥터 컬렉션을 사용하여 수많은 작업을 오케스트레이션합니다.

Service Bus 큐에서 메시지가 수신되는 경우
HTTP 요청을 수신하는 경우
새
우
Event Grid 이벤트가
할 때
되풀이
Outlook.com에서 새 전자 메일이 수신되는 경우
OneDrive에서 새 파일이 만들어지는 경우
F
일이 추
가되

템플릿

논리 앱을 만들려면 아래 템플릿을 선택하세요.

범주: 모두

정렬 기준: 인기도

비어 있는 논리 앱
+
HTTP 요청-응답
보기 잠금에서 Service Bus 메시지를 받아 완료
A
DI
문서
XML
Service Bus 세션을
Sharepoint 목록의
AS2 페이로드를 수

HTTP 요청을 수신하는 경우

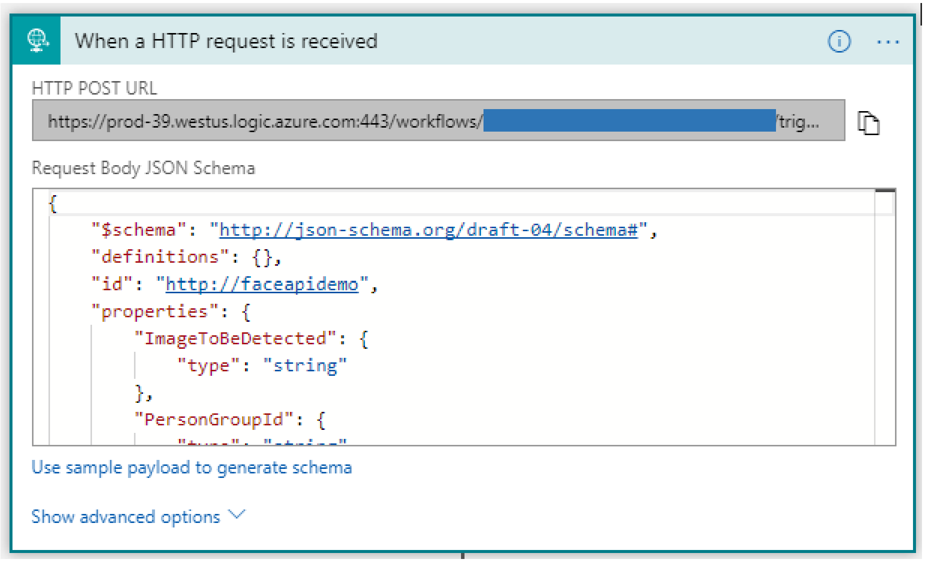
HTTP POST URL: 저장 후 URL이 생성됨

매개 변수에 기본값을 사용합니다. 편집

LOGIC APP

HTTP request

```
{
   "ImageToBeDetected" : image url,
   "referenceImageUrl" : image url,
   "sendEmail": 결과값 보낼 email,
   "PersonGroupId" :  faceapigrpid1" ,
   "PersonGroupName" : "faceapigrpname1",
   "PersonName" : "TestPerson1"
}
```



When a HTTP request is received

HTTP POST URL

https://prod-39.westus.logic.azure.com:443/workflows/[redacted]/trig...

Request Body JSON Schema

```
{
    "$schema": "http://json-schema.org/draft-04/schema#",
    "definitions": {},
    "id": "http://faceapidemo",
    "properties": {
        "ImageToBeDetected": {
            "type": "string"
        },
        "PersonGroupId": {
```

Use sample payload to generate schema

Show advanced options

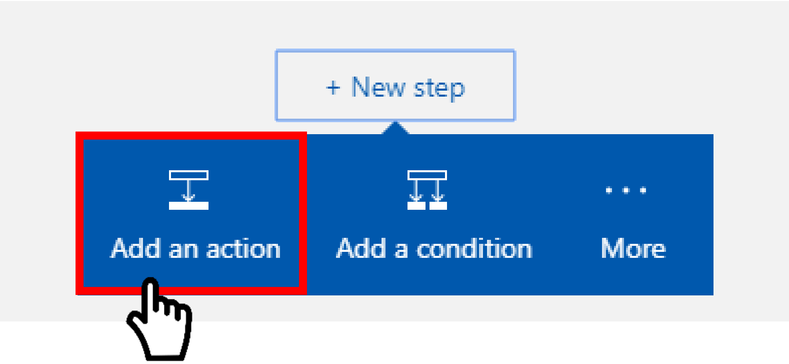
LOGIC APP

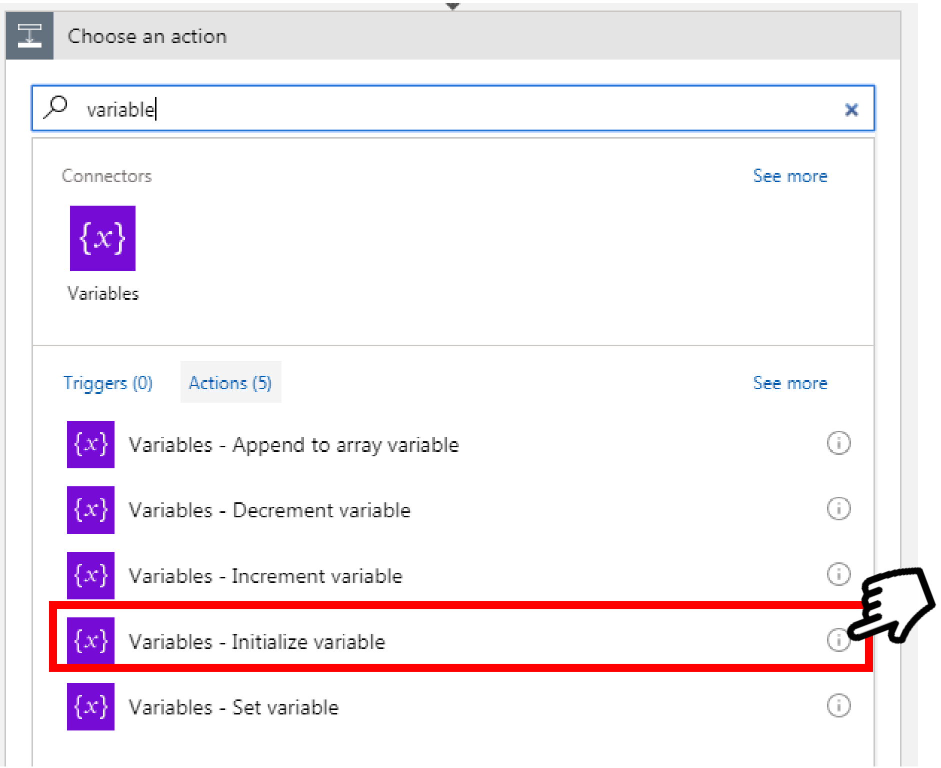
Request Body JSON Schema

```
{
        "$schema": "http://json-schema.org/draft-04/schema#",
        "definitions": {},
        "id": "http://faceapidemo",
        "properties": {
                "ImageToBeDetected": {
                                "type": "string"            },
                "PersonGroupId": {
                                "type": "string"            },
                "PersonGroupName": {
                                "type": "string"            },
                "PersonName": {
                                "type": "string"            },
                "referenceImageUrl": {
                                "type": "string"            },
                "sendEmail": {
                                "type": "string"            }
        },
        "required": [
                "referenceImageUrl",
                "ImageToBeDetected",
                "sendEmail",
                "PersonGroupId",
                "PersonGroupName",
                "PersonName"
        ],
        "type": "object"
}
```

LOGIC APP

유사도분석후결과값저장할변수생성

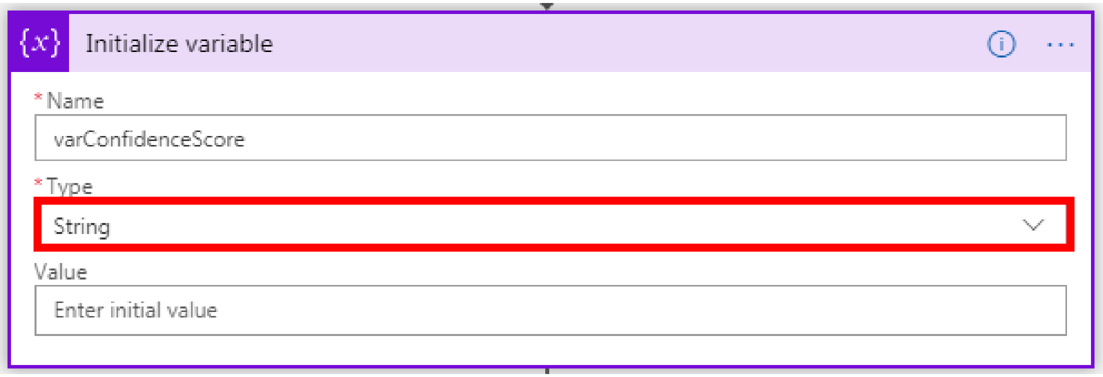



Choose an action

variable

Connectors
See more
Variables
Triggers (0)
Actions (5)
See more
Variables - Append to array variable
Variables - Decrement variable
Variables - Increment variable
Variables - Initialize variable
Variables - Set variable

Face Verification으로 나온 Confidence Score을 저장할 string 변수 생성


Initialize variable

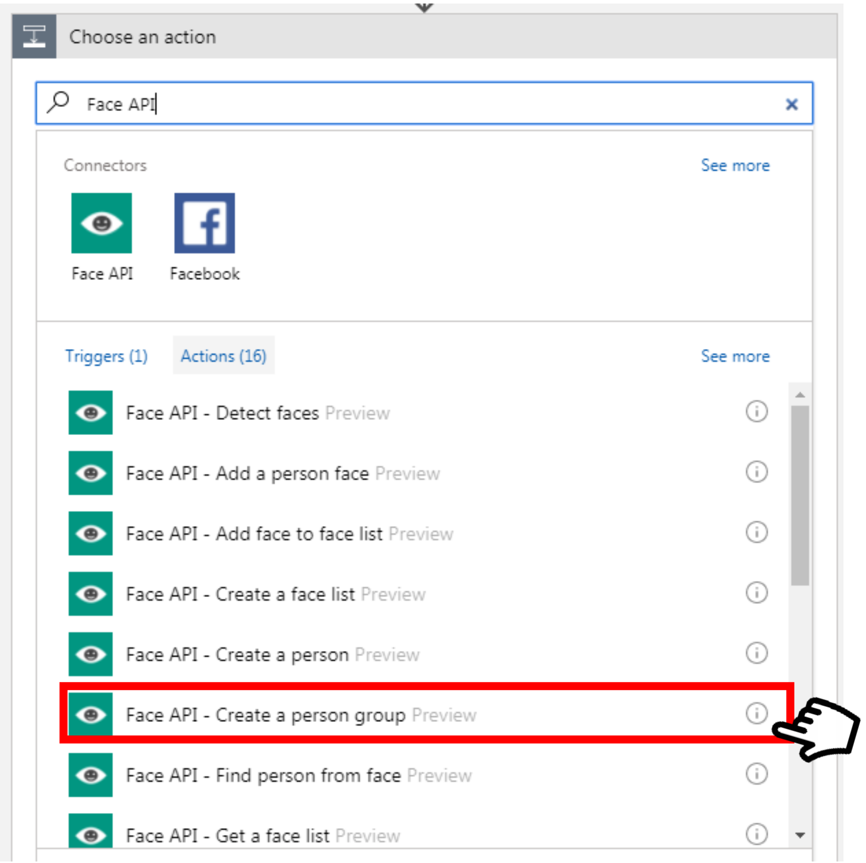
*Name: varConfidenceScore

*Type: String

Value: Enter initial value

LOGIC APP

Face API



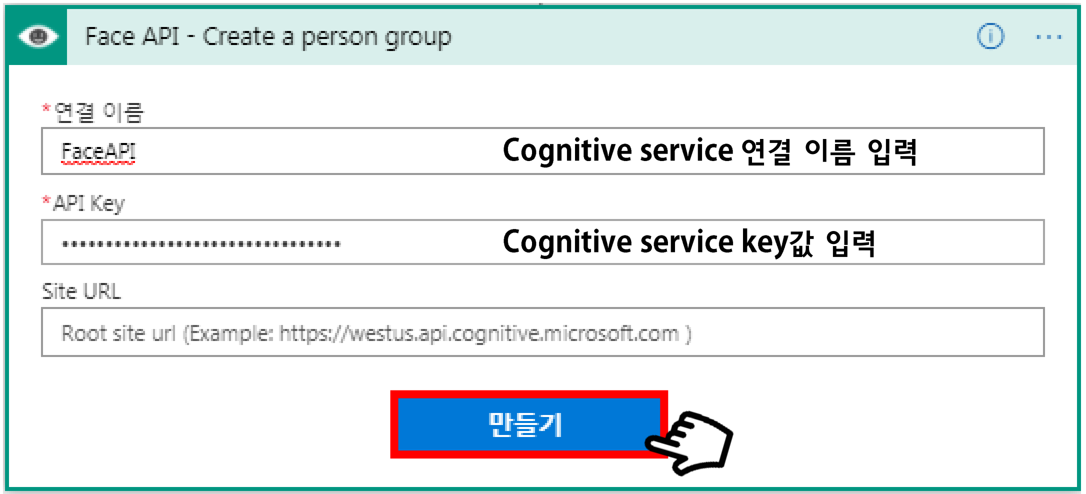
Choose an action

Face API

Connectors See more

Triggers (1) Actions (16) See more

- Face API - Detect faces Preview
- Face API - Add a person face Preview
- Face API - Add face to face list Preview
- Face API - Create a face list Preview
- Face API - Create a person Preview
- Face API - Create a person group Preview
- Face API - Find person from face Preview
- Face API - Get a face list Preview



Face API - Create a person group

*연결 이름	FaceAPI
*API Key	**********************************
Site URL	Root site url (Example: https://westus.api.cognitive.microsoft.com)

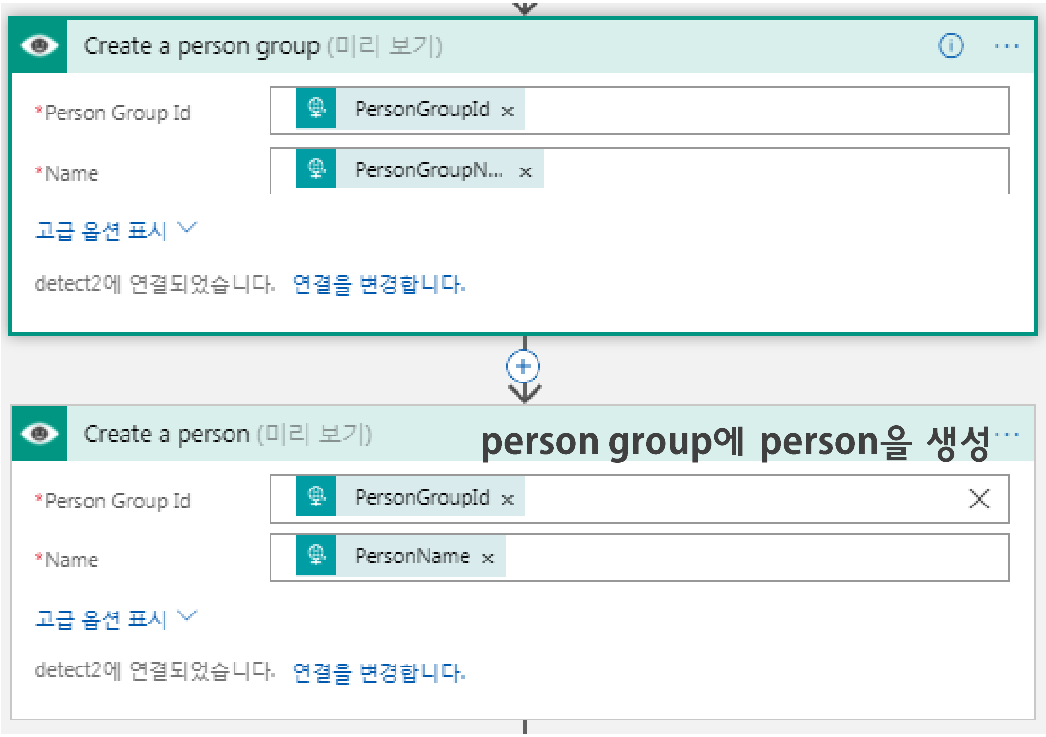
Cognitive service 연결 이름 입력

Cognitive service key값 입력

unique한 person group id를 통해 imageA의 얼굴을 저장할 person group을 생성

LOGIC APP

Face API



Create a person group (미리 보기)

*Person Group Id: PersonGroupId ×

*Name: PersonGroupN... ×

고급 옵션 표시

detect2에 연결되었습니다. 연결을 변경합니다.

Create a person (미리 보기)

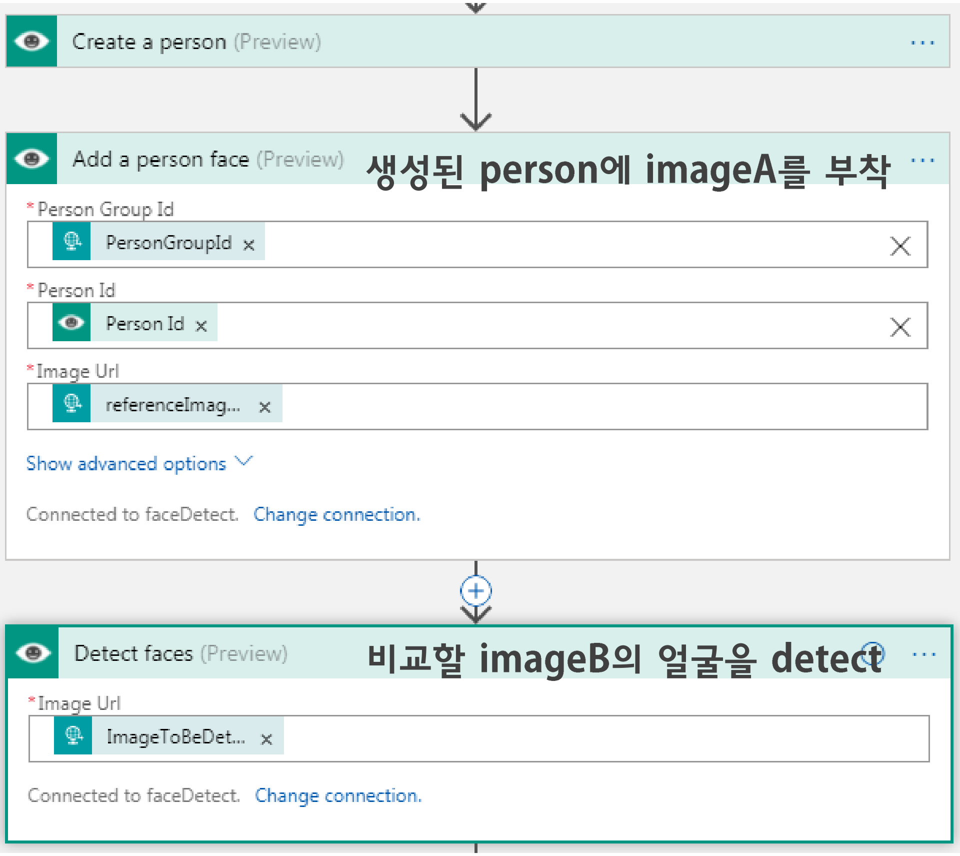
person group에 person을 생성

*Person Group Id: PersonGroupId ×

*Name: PersonName ×

고급 옵션 표시

detect2에 연결되었습니다. 연결을 변경합니다.



Create a person (Preview)

Add a person face (Preview)

생성된 person에 imageA를 부착

*Person Group Id: PersonGroupId ×

*Person Id: Person Id ×

*Image Url: referenceImag... ×

Show advanced options

Connected to faceDetect. Change connection.

Detect faces (Preview)

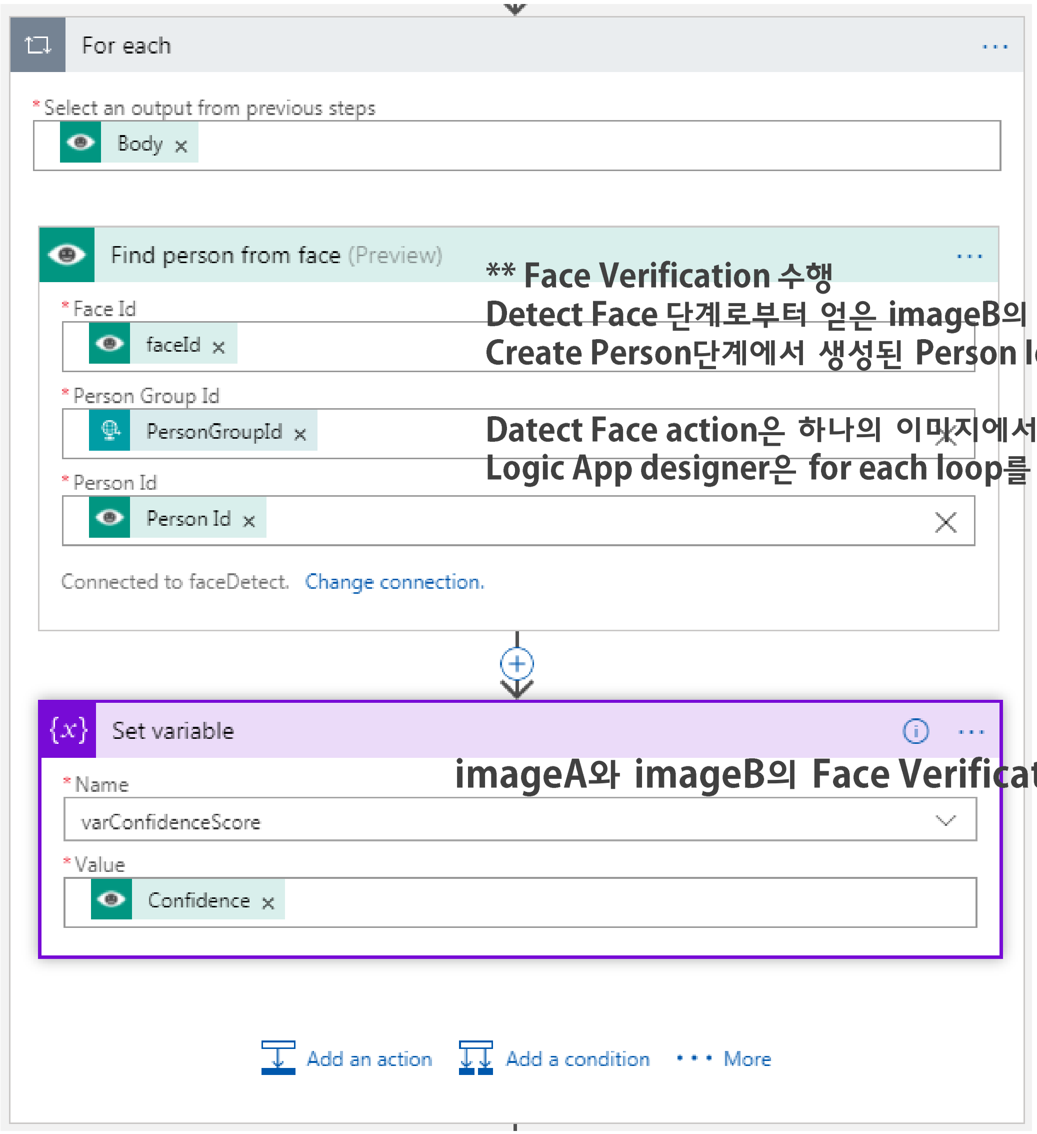
비교할 imageB의 얼굴을 detect

*Image Url: ImageToBeDet... ×

Connected to faceDetect. Change connection.

LOGIC APP

Face API



For each

* Select an output from previous steps
Body ×

Find person from face (Preview)

* Face Id
faceId ×

* Person Group Id
PersonGroupId ×

* Person Id
Person Id ×

Connected to faceDetect. Change connection.

Set variable

* Name
varConfidenceScore

* Value
Confidence ×

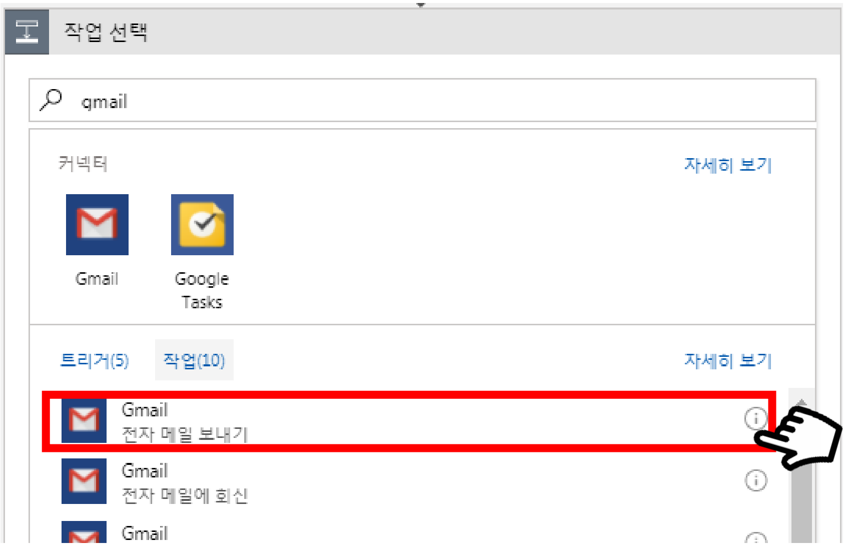
** Face Verification 수행
Detect Face 단계로부터 얻은 imageB의 Face Id와
Create Person단계에서 생성된 Person Id, 입력해둔 Person Group Id를 통해 Find Person From Face action을 수행

Datect Face action은 하나의 이미지에서 다양한 얼굴들을 detect 할 수 있기 때문에
Logic App designer은 for each loop를 추가 할 것

imageA와 imageB의 Face Verification을 통해 얻은 Confidence Score을 변수에 저장

LOGIC APP

Gmail 연결



작업 선택

gmail

커넥터 자세히 보기

Gmail

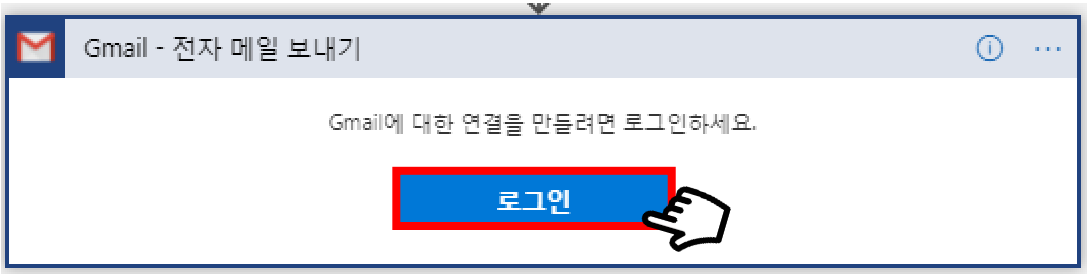
Google Tasks

트리거(5) 작업(10) 자세히 보기

Gmail
전자 메일 보내기

Gmail
전자 메일에 회신

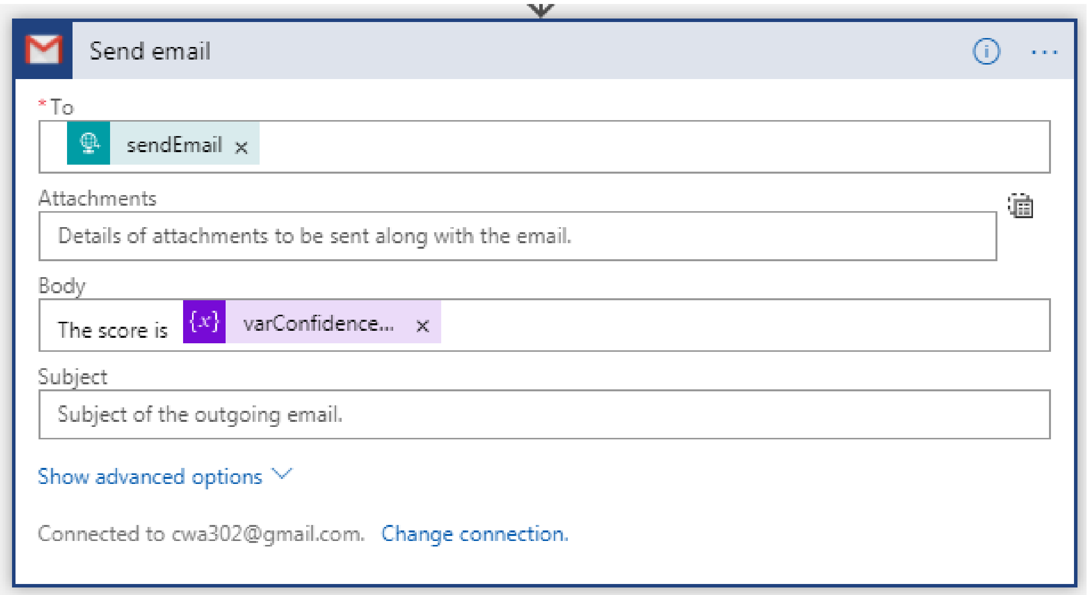
Gmail



Gmail - 전자 메일 보내기

Gmail에 대한 연결을 만들려면 로그인하세요.

로그인



Send email

*To

sendEmail ×

Attachments

Details of attachments to be sent along with the email.

Body

The score is {x} varConfidence... ×

Subject

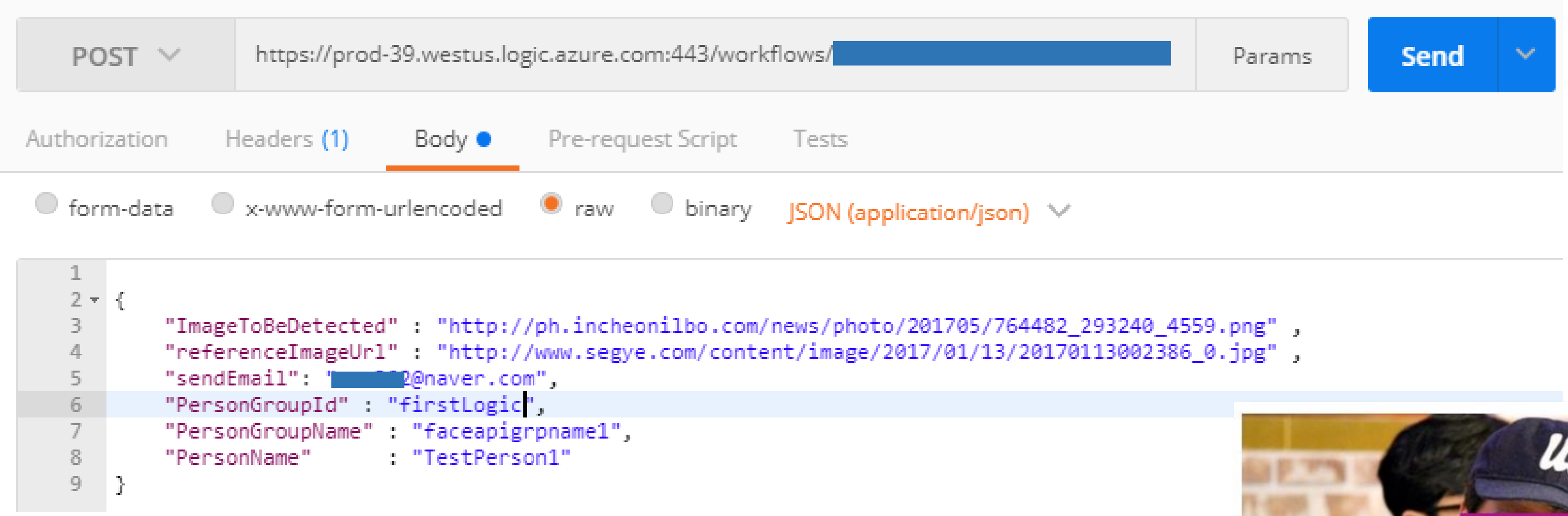
Subject of the outgoing email.

Show advanced options

Connected to cwa302@gmail.com. Change connection.

LOGIC APP

결과값 출력

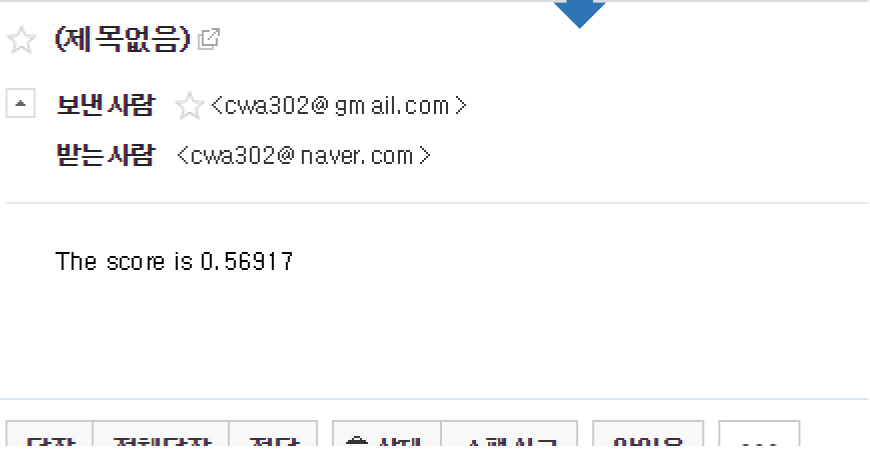


https://prod-39.westus.logic.azure.com:443/workflows/[redacted]

```

{
    "ImageToBeDetected" : "http://ph.incheonilbo.com/news/photo/201705/764482_293240_4559.png" ,
    "referenceImageUrl" : "http://www.segye.com/content/image/2017/01/13/20170113002386_0.jpg" ,
    "sendEmail": "[redacted]@naver.com",
    "PersonGroupId" : "firstLogic",
    "PersonGroupName" : "faceapigrpname1",
    "PersonName"        : "TestPerson1"
}
```

실습 결과값



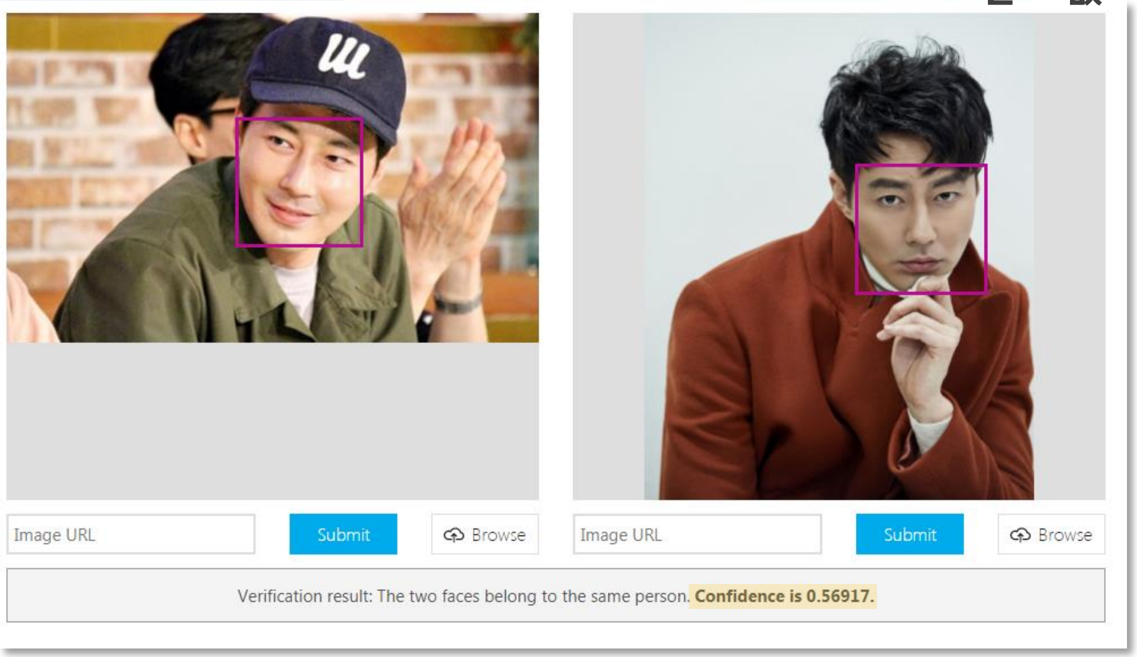
☆ (제목없음)

보낸사람 ☆ <cwa302@gmail.com>

받는사람 <cwa302@naver.com>

The score is 0.56917

website 결과값



Verification result: The two faces belong to the same person. **Confidence is 0.56917.**

감사합니다☺