

Raspberry Pi를 활용한 IoT 프로젝트

나만의 가상비서 만들기(구글 어시스턴트)

5일차

2021.07.16 10:00~13:00

담당교수 : 조도은



1일차:라즈베리파이 소개와 환경 구축(3H)

2일차:라즈베리파이를 위한 리눅스 기초 배우기(3H)

3일차:파이썬 기초 명령어 익히기(3H)

4일차:라즈베리파이 GPIO와 센서 동작하기(3H)

5일차:나만의 가상비서 만들기(구글 어시스턴트)(3H)



- 인공지능 비서, AI 어시스턴트

- 하드웨어 및 네트워크 액세스 설정
- 오디오 마이크 설정 및 테스트하기
- 개발자 프로젝트 및 계정 설정 구성
- 장치 모델 등록
- SDK 및 샘플 코드 설치
- 샘플 코드 실행
- 나만의 프로젝트 구상하기



- 필수 하드웨어
 - 라즈베리파이4모델B 및 전원공급장치(권장사양: 3B 이상)
 - USB 마이크
 - USB 스피커
 - 라즈비안 OS 설치된 SD 카드
 - USB 케이블, USB 마우스 및 HDMI 케이블이 있는 모니터



- 하드웨어 연결 및 네트워크 액세스 구성
 - 마이크와 스피커를 Raspberry Pi에 연결
 - SD 카드를 Raspberry Pi에 삽입
 - 이더넷 케이블을 연결하거나 Wi-Fi 네트워크에 연결
- 날짜 및 시간 구성
 - `$ sudo apt-get install rdate`

※참고: 날짜 또는 시간이 올바르지 않으면 SSL 오류가 발생할 수 있습니다.
따라서 시간 서버와 동기화 해주는 작업이 필요합니다.



■ 스피커 연결

- USB 단자와 스피커 잭을 연결



■ 마이크 연결

- USB 단자에 연결

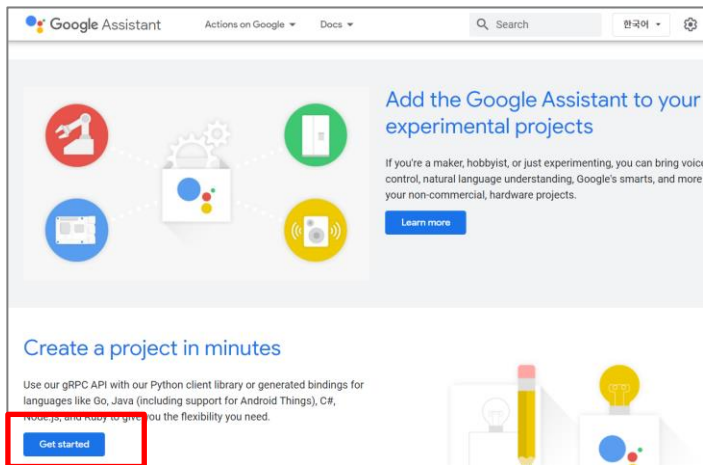




오디오 마이크 설정 및 테스트하기

- 구글 어시스턴트 사이트 이동하기

- <https://developers.google.com/assistant/sdk>
- 구글 계정으로 로그인 >> Get started





■ 다음 단계에 따라 차례대로 진행 (총 12단계)

GuidesReferenceSupport

Learn

Overview

Device Actions

Release Notes

Terms of Service

Google Assistant Service

Introduction

▶ Embed the Google Assistant (Python)

▶ Extend the Google Assistant (Python)

Integrate the Assistant into Your Project (Other Languages)

Reference

Troubleshooting

▶ Best Practices

Google Assistant Library

Embed the Google Assistant

This section gets the Google Assistant Service sample working on your device:

1. [Set Up Hardware and Network Access](#)
2. [Configure and Test the Audio](#)
3. [Configure a Developer Project and Account Settings](#)
4. [Register the Device Model](#)
5. [Install the SDK and Sample Code](#)
6. [Run the Sample Code](#)
7. [Next Steps](#)

Extend the Google Assistant

This section extends the Google Assistant Service sample to include [Device Actions](#):

1. [Install Hardware](#)
2. [Register Traits](#)
3. [Handle Commands](#)
4. [Add More Traits and Handlers](#)
5. [Register Custom Device Actions](#)



마이크와 스피커 Card & Device number 확인

- 녹음및재생장치찾기
 - 캡처하드웨어장치목록에서 **USB마이크** 찾기

\$ **arecord -l** #카드번호,장치번호기록

```
pi@raspberrypi:~$ arecord -l
**** List of CAPTURE Hardware Devices ****
card 1: Device [USB PnP Sound Device], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

- 재생하드웨어장치목록에서 **스피커** 찾기

\$ **aplay -l** #카드번호,장치번호기록

```
pi@raspberrypi:~$ aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: Headphones [bcm2835 Headphones], device 0: bcm2835 Headphones [bcm2835 Headphones]
  Subdevices: 8/8
  Subdevice #0: subdevice #0
  Subdevice #1: subdevice #1
  Subdevice #2: subdevice #2
  Subdevice #3: subdevice #3
  Subdevice #4: subdevice #4
  Subdevice #5: subdevice #5
  Subdevice #6: subdevice #6
  Subdevice #7: subdevice #7
```



~\$ nano ~/.asoundrc

```
pcm.!default {  
    type asym  
    capture.pcm "mic"  
    playback.pcm "speaker"  
}  
pcm.mic {  
    type plug  
    slave {  
        pcm "hw:1,0"  
    }  
}  
pcm.speaker {  
    type plug  
    slave {  
        pcm "hw:0,0"  
    }  
}
```

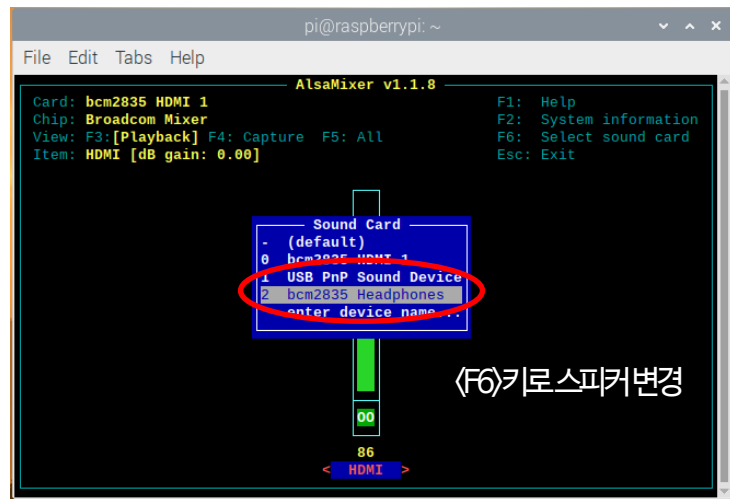
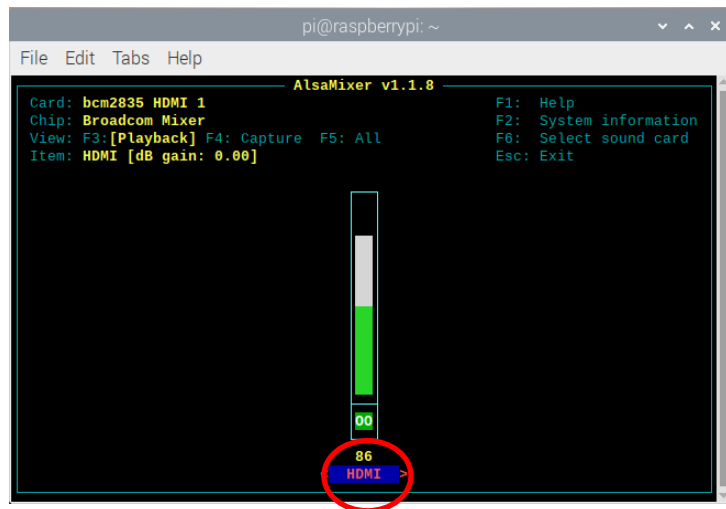


오디오구성 및 테스트

- 녹음 및 재생 확인

- 재생음량 조절

\$alsamixer # 위쪽 화살표 키를 눌러 재생 볼륨 레벨을 약 70으로 설정





오디오 구성 및 테스트

- 스피커 소리 테스트하기(완료 **Ctrl+C**)
\$ **speaker-test -t wav**

```
pi@raspberrypi: ~  
File Edit Tabs Help  
pi@raspberrypi:~$ gedit .asoundrc  
pi@raspberrypi:~$ alsamixer  
pi@raspberrypi:~$ speaker-test -t wav  
  
speaker-test 1.1.8  
  
Playback device is default  
Stream parameters are 48000Hz, S16_LE, 1 channels  
WAV file(s)  
Rate set to 48000Hz (requested 48000Hz)  
Buffer size range from 512 to 65536  
Period size range from 512 to 65536  
Using max buffer size 65536  
Periods = 4  
was set period_size = 16384  
was set buffer_size = 65536  
  0 - Front Left  
Time per period = 0.360085  
  0 - Front Left  
Time per period = 1.371370  
  0 - Front Left  
^CTransfer failed: Bad address  
pi@raspberrypi:~$
```



오디오 구성 및 테스트

- 마이크 녹음 및 재생 확인

- 짧은 오디오 클립을 녹음

```
$ arecord --format=S16_LE --duration=5 --rate=16000 --file-type=raw out.raw
```


- 녹음 재생

```
$ aplay --format=S16_LE --rate=16000 out.raw
```

```
pi@raspberrypi: ~  
File Edit Tabs Help  
pi@raspberrypi:~ $ arecord --format=S16_LE --duration=5 --rate=16000 --file-type=raw out.raw  
Recording raw data 'out.raw' : Signed 16 bit Little Endian, Rate 16000 Hz, Mono  
pi@raspberrypi:~ $ aplay --format=S16_LE --rate=16000 out.raw  
Playing raw data 'out.raw' : Signed 16 bit Little Endian, Rate 16000 Hz, Mono  
pi@raspberrypi:~ $
```



개발자 프로젝트 및 계정 구성 설정

- Actions Console 프로젝트 구성
 - 작업 콘솔을 열기
 -  (<https://console.actions.google.com/?pli=1>)
 - 프로젝트 추가/가져오기를 클릭
 - 새 프로젝트를 만들려면 프로젝트 이름 입력



개발자 프로젝트 및 계정 구성 설정

Welcome to Actions on Google


Actions on Google is the platform for developers to extend the Google Assistant. Join this emerging ecosystem by developing actions to engage users on Google Home, Pixel, and many other surfaces where the Google Assistant will be available.

[Documentation](#) [Sample code](#) [API reference](#) [Support](#)

NEW We've improved the build experience with the new Actions Builder and Actions SDK. [Watch Video](#) [Learn More](#) [×](#)

Your projects

[New project](#)

Recent projects	Deployed channels	Last modified date
 My Project	—	Jul 18, 2020

[Terms](#) · [Privacy Policy](#)

CANCEL CREATE PROJECT



개발자 프로젝트 및 계정 구성 설정

New Project

Project Name

Enter name or choose a project

Choose the default language for your Actions English ▾

Choose your country or region United States ▾

CANCEL CREATE PROJECT



New Project

Project Name

airaspberry

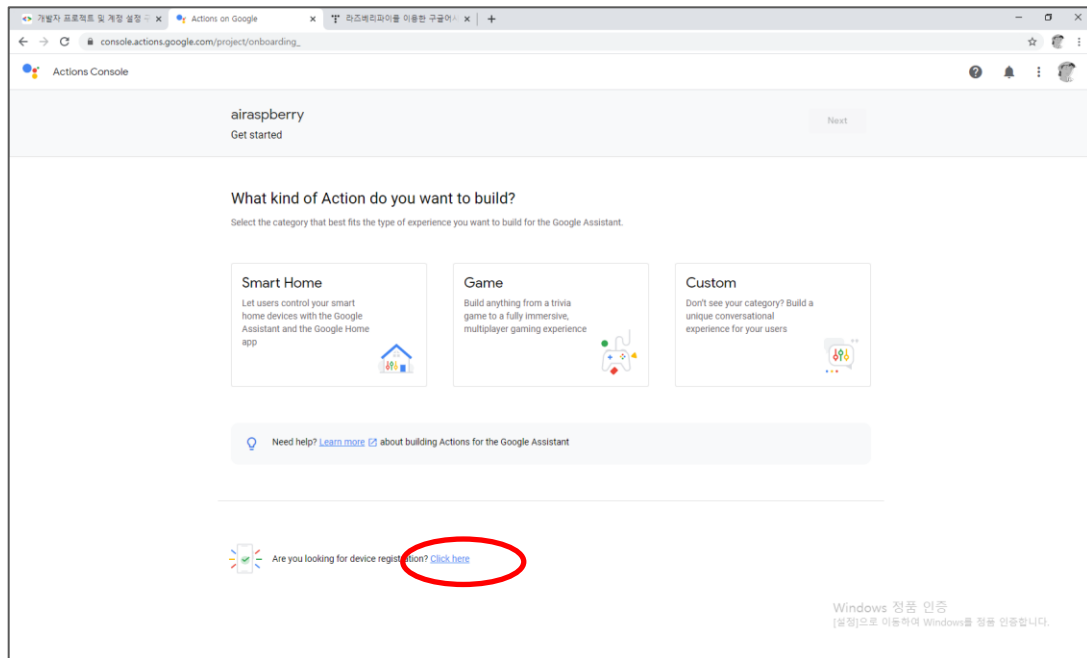
Choose a language for your action (you can change later) Korean ▾

Choose your country or region South Korea ▾

Cancel Create project

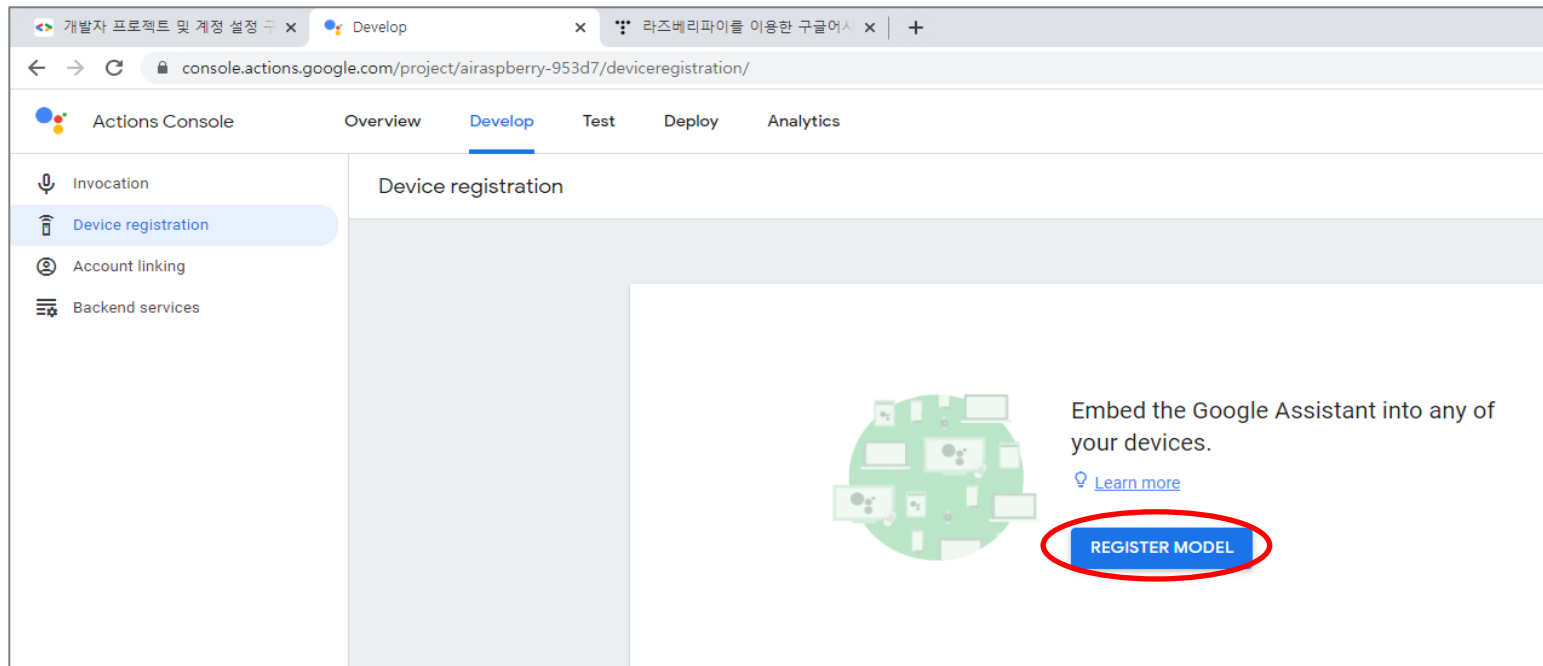


개발자 프로젝트 및 계정 구성 설정



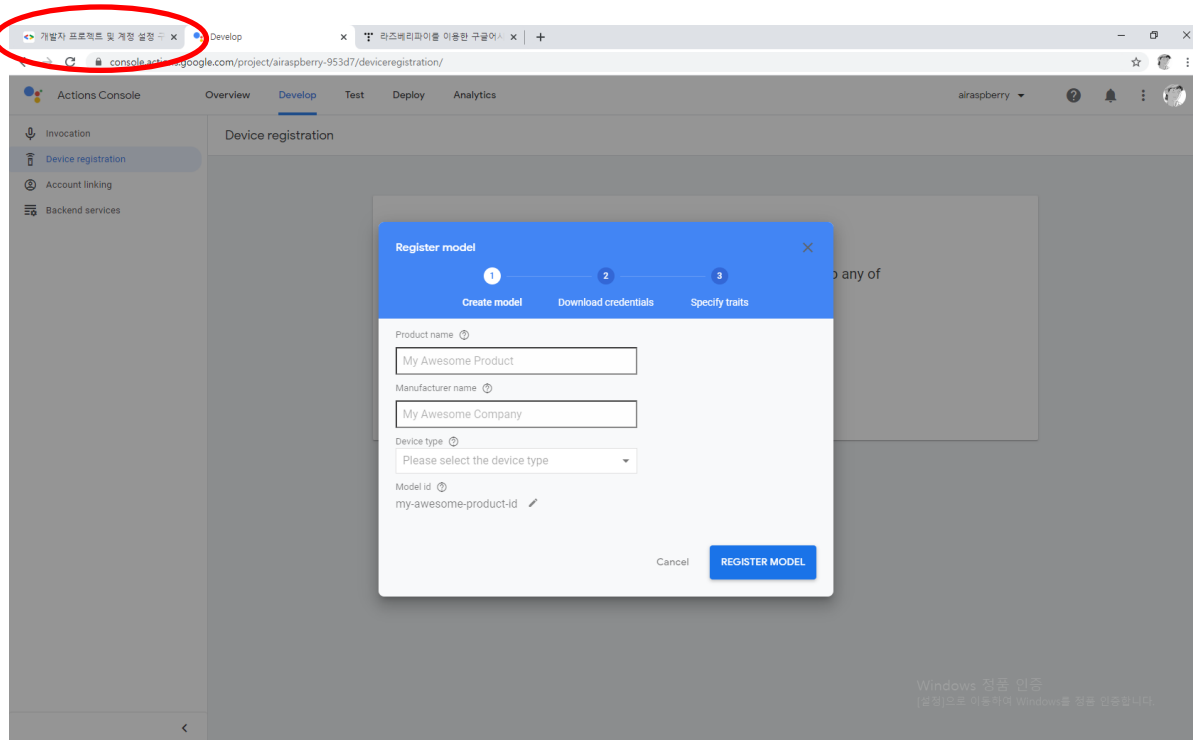


장치 등록(REGISTER MODEL 클릭)



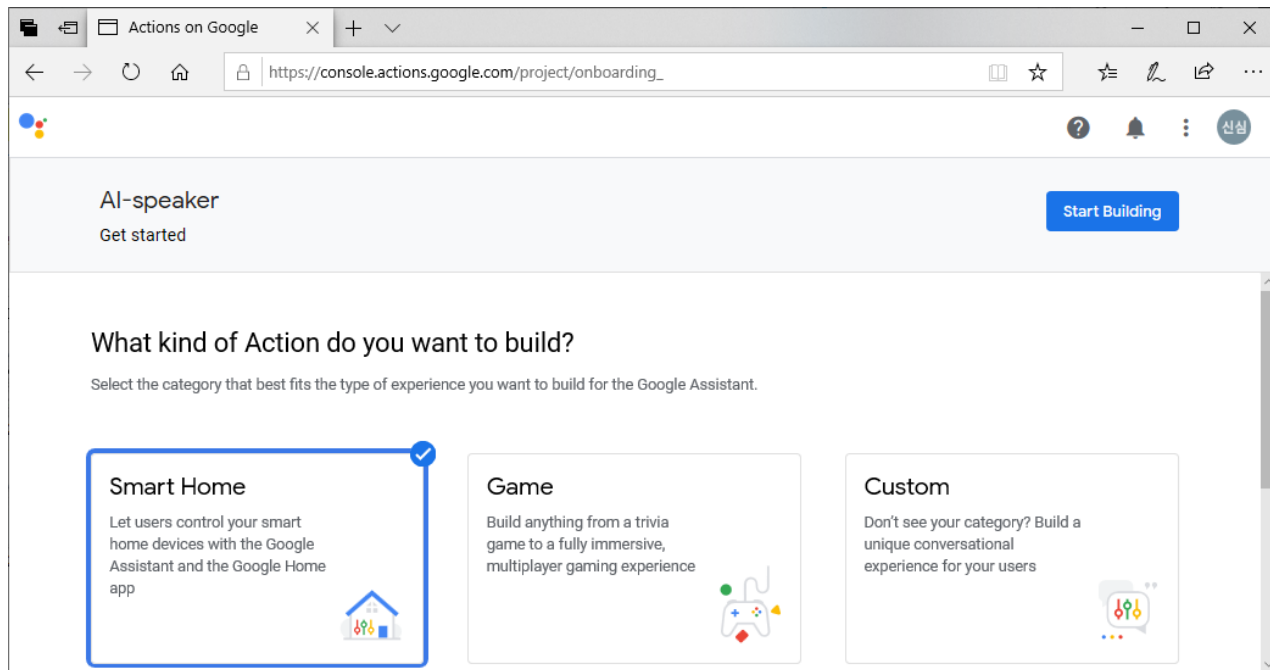


클릭 후 뜨는 창 그대로 두고 이전 작업으로 복귀





개발자 프로젝트 및 계정 구성 설정

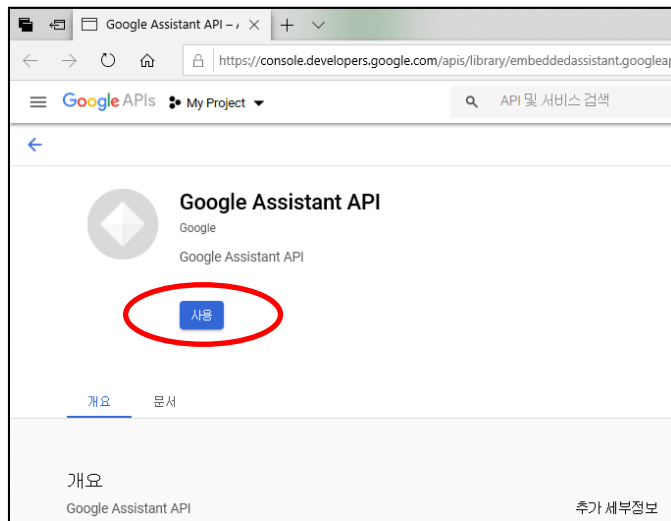




개발자 프로젝트 및 계정 구성 설정

- Actions Console 프로젝트 구성
 - 선택한 프로젝트에서 Google Assistant API 사용 설정
 - Cloud Platform 콘솔에서 작업을 수행

Enable the API





개발자 프로젝트 및 계정 구성 설정

The screenshot shows the Google Assistant API console interface. The left sidebar contains navigation links: 개요 (Overview), 측정항목 (Metrics), 할당량 (Quota), and 사용자 인증 정보 (User Consent). The main content area displays the 'API 사용 중지' (API Disabled) status with a message: '이 API를 사용하려면 사용자 인증 정보가 필요할 수 있습니다. 사용자 인증 정보 만들기를 클릭하여 시작하세요.' (You may need user consent information to use this API. Click 'Create user consent' to get started). A red circle highlights the '사용자 인증 정보 만들기' (Create user consent) button. Below the message, there is a '세부 정보' (Details) section showing the API name 'Google Assistant API', the creator 'Google', and the service name 'embeddedassistant.googleapis.com'. The '응답 코드별 트래픽' (Traffic by response code) section shows a line graph with the message 'No data is available for the selected time frame.' and a '측정항목 보기' (View metrics) link.

Google APIs My Project API 및 서비스 검색

API 및 서비스 Google Assistant API 개요 API 사용 중지

이 API를 사용하려면 사용자 인증 정보가 필요할 수 있습니다. 사용자 인증 정보 만들기를 클릭하여 시작하세요.

사용자 인증 정보 만들기

세부 정보

이름
Google Assistant API

생성자:
Google

서비스 이름
embeddedassistant.googleapis.com

개요
Google Assistant API

활성화 상태
사용 설정됨

가이드 및 문서
Learn more

응답 코드별 트래픽

요청/초(2시간 평균)

No data is available for the selected time frame.

측정항목 보기

https://console.developers.google.com/apis/api/embeddedassistant.googleapis.com/metrics?project=my-proj



개발자 프로젝트 및 계정 구성 설정

The screenshot shows the Google Developers console interface. The browser address bar displays the URL: `https://console.developers.google.com/apis/credentials/wizard?project=my-project-1573191566432`. The page title is "API 및 서비스 검색". The left sidebar contains a menu with the following items: "API 및 서비스", "대시보드", "라이브러리", "사용자 인증 정보" (highlighted), "OAuth 동의 화면", "도메인 확인", and "페이지 사용 동의". The main content area is titled "사용자 인증 정보" and "프로젝트에 사용자 인증 정보 추가". It contains two steps: 1. "필요한 사용자 인증 정보의 종류 확인" (Confirm the type of user authentication information needed), which includes instructions on selecting the type of authentication (API key, OAuth 2.0 client ID, or Service account) and a dropdown menu currently showing "Google Assistant API". 2. "사용자 인증 정보 가져오기" (Get user authentication information), which includes a "취소" (Cancel) button.



개발자 프로젝트 및 계정 구성 설정

The screenshot shows the Google API Console interface for configuring an OAuth consent screen. The left sidebar contains a menu with options: 대시보드, 라이브러리, 사용자 인증 정보, OAuth 동의 화면 (selected), 도메인 확인, and 페이지 사용 동의. The main content area is titled 'OAuth 동의 화면' and contains the following text: '대상 사용자를 비롯해 앱을 구성하고 등록하려는 방식을 선택하세요. 프로젝트에는 하나의 앱만 연결할 수 있습니다.' Below this, under the heading 'User Type', there are two radio button options: '내부' (Internal) and '외부' (External). The '외부' option is selected and circled in red. Below the radio buttons, the text reads: '조직 내 사용지만 사용할 수 있습니다. 확인을 위해 앱을 제출할 필요는 없습니다.' and 'Google 계정이 있는 모든 사용자가 사용할 수 있습니다.' At the bottom of the configuration area is a blue button labeled '만들기' (Create). A link at the very bottom says 'Let us know what you think about our OAuth experience'.



개발자 프로젝트 및 계정 구성 설정

Configure a Developer Project x 동 의 화면 - airasberry - Google x 개요 - API 및 서비스 - airasberry x Develop

← → ↻ console.developers.google.com/apis/credentials/consent/edit?newAppInternalUser=false&project=airasberry-953d7&duration=

📦 무료 평가판을 사용할 수 있습니다. 지금 활성화하여 Google Cloud 제품에 쓸 수 있는 \$300의 크레딧을 받아 보세요. [자세히 알아보기](#)

☰ Google APIs airasberry 🔍 API 및 서비스 검색

API	API 및 서비스	OAuth 동의 화면
🔄	대시보드	
📖	라이브러리	
👤	사용자 인증 정보	
🔗	OAuth 동의 화면	<p>사용자는 인증 전에 이 동의 화면을 통해 비공개 데이터에 대한 액세스 권한을 부여할지 선택할 수 있으며 서비스 약관 및 개인정보처리방침 링크도 제공됩니다. 이 페이지에서 프로젝트에 속한 모든 애플리케이션의 동의 화면을 구성합니다.</p> <p>확인 상태 게시되지 않음</p> <p>애플리케이션 이름 동의를 요청하는 앱의 이름 <u>project-554172461886</u></p> <p>애플리케이션 로고 사용자가 앱을 알아보는 데 도움이 되는 동의 화면의 이미지입니다. 업로드용 로컬 파일 <input type="text"/> <input type="button" value="탐색"/></p>
🔒	도메인 확인	
🔗	페이지 사용 동의	

동의 화면 정보

동의 화면에서 사용자에게 제공하는 당사자 및 액세스 권한을 안내합니다.

OAuth 확인

나와 내 사용자를 보호하기 위
면과 애플리케이션을 확인해
공개로 표시되어 있고 다음 중
우 확인이 필요합니다.

- 앱에서 민감하거나 제한
- 앱의 OAuth 동의 화면에
- 앱에 승인된 도메인이 많
- 이전에 확인된 OAuth 동

다.



개발자 프로젝트 및 계정 구성 설정

4@gmail.com

Google에서 앱의 범위
범위를 통해 애플리케이션에서 사용자의 비공개 데이터에 액세스할 수 있습니다. 자세히 알
아보기

캘린더 또는 드라이브에 대한 전체 액세스 권한을 부여하는 민감한 범위를 추가하는 경우에
는 Google에서 동의 화면을 확인한 후에 게시됩니다.

email

profile

openid

범위 추가

승인된 도메인 ⓘ
나와 내 사용자를 보호하기 위해 Google에서는 OAuth로 인증하여 승인된 도메인을 사용하
는 애플리케이션만 허용합니다. 애플리케이션의 링크가 승인된 도메인에서 호스팅되어야
합니다. 자세히 알아보기

airaspberry-953d7.firebaseio.com

example.com

도메인을 입력하고 Enter 키를 눌러 추가하세요.

애플리케이션 종파이지 링크
동의 화면에 표시됩니다. 승인된 도메인에서 호스팅해야 합니다.

https:// 또는 http://

애플리케이션 개인정보처리방침 링크
동의 화면에 표시됩니다. 승인된 도메인에서 호스팅해야 합니다.

https:// 또는 http://

애플리케이션 서비스 약관 링크 (선택사항)
동의 화면에 표시됩니다. 승인된 도메인에서 호스팅해야 합니다.

https:// 또는 http://

저장 재확인 확인받기 취소



개발자 프로젝트 및 계정 구성 설정

API 키 제한 및 이름 변경

이름 *
API 키 1개

API Key
AIzaSyD-5P80EmBCfepd0QpodQA81n1KPq8fTVk

key=API_KEY 매개변수로 키를 전달하여 애플리케이션에서 이 키를 사용하세요.

생성일 2020년 7월 18일 PM 5시 42분 17초 GMT+9

생성자 sskang7@gmail.com (LI)

총 사용량(지난 30일) 0

키 제한사항

이 키는 제한되지 않습니다. 제한사항을 통해 승인되지 않은 사용 및 할당량 도용을 방지할 수 있습니다. [자세히 알아보기](#)

애플리케이션 제한사항

애플리케이션 제한사항은 API 키를 사용할 수 있는 웹 사이트, IP 주소 또는 애플리케이션을 제어합니다. 키별로 애플리케이션 제한사항 1개를 설정할 수 있습니다.

☒ 없음

☐ HTTP 리퍼러(웹 사이트)

☐ IP 주소(웹 서버, 크론 작업 등)

☐ Android 앱

☐ iOS 앱

API 제한사항

API 제한사항은 이 키를 호출할 수 있는 사용 설정된 API를 지정합니다.

☒ 키 제한 안함

이 키는 모든 API를 호출할 수 있습니다.

☐ 키 제한

참고: 설정이 적용되는 데 최대 5분이 걸릴 수 있습니다.

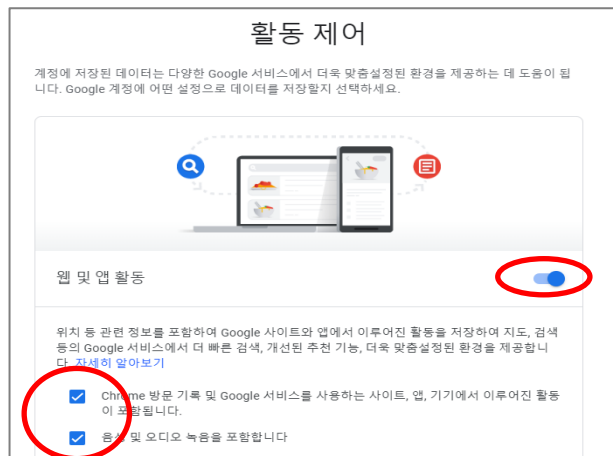
저장 취소



개발자 프로젝트 및 계정 구성 설정

- 계정 활동 관리 설정

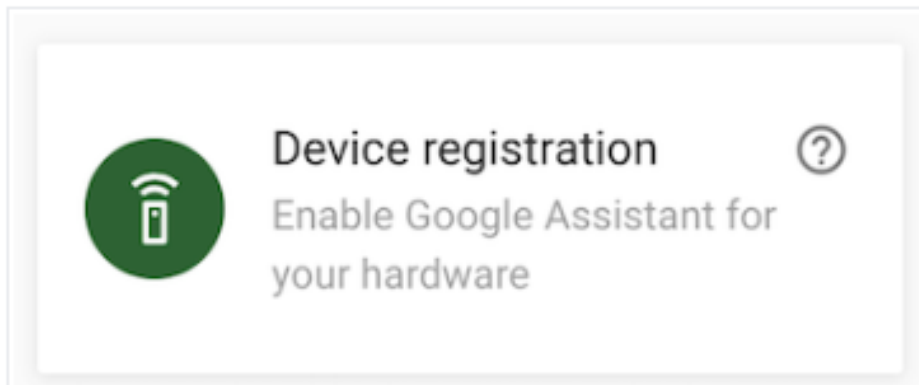
- Google 어시스턴트를 사용하려면 특정 활동 데이터를 Google 과 공유
- Google 어시스턴트가 제대로 작동하려면 데이터가 필요
- Google 계정이 필요





개발자 프로젝트 및 계정 구성 설정

- Actions Console 프로젝트 구성
 - 새 프로젝트를 생성한 경우 장치 등록 상자 클릭(이전에 만든 프로젝트를 가져오는 경우 나타나지 않음)
 - 이 브라우저 탭을 열어두기, 이후 단계에서 이를 사용

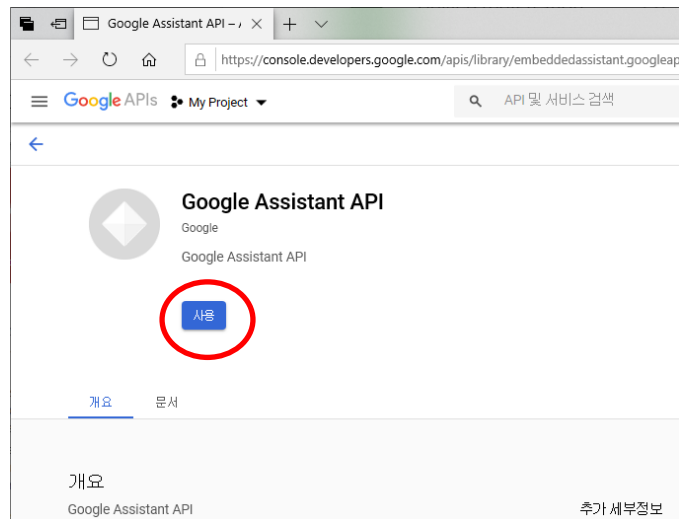




개발자 프로젝트 및 계정 구성 설정

- Actions Console 프로젝트 구성
 - 선택한 프로젝트에서 Google Assistant API 사용 설정
 - Cloud Platform 콘솔에서 작업을 수행
 - API 사용

Enable the API





개발자 프로젝트 및 계정 구성 설정

- Actions Console 프로젝트 구성
 - Cloud Platform 콘솔에서 프로젝트에 대한 OAuth 동의화면을 구성
 - 대부분이 선택사항

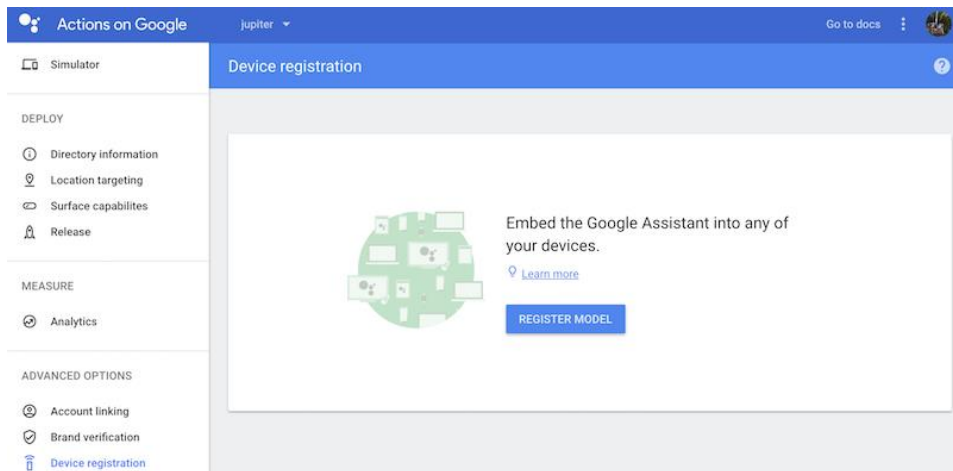
API	APIs & Services	Credentials
	<ul style="list-style-type: none">DashboardLibraryCredentials	<div>Credentials <u>OAuth consent screen</u> Domain verification</div> <div>Email address ? nickcooktest7@gmail.com</div> <div>Product name shown to users ? project-447230429267</div> <div>Homepage URL (Optional) https:// or http://</div>



장치 모델 등록

- 등록 UI 사용

- [Actions Console](#)에서 등록 UI를 사용하여 장치 모델을 등록



※ 참고 : 프로젝트를 생성 할 때 장치 등록 상자를 클릭하지 않으면 등록 UI가 나타나지 않습니다 (이전 단계 참조).



장치 모델 등록

- 모델만들기
 - 장치의 모든 필드 채우기
 - 등록 모델 클릭

Register model

1 2 3

Create model Download credentials Specify traits

Product name ⓘ

My Awesome Product

Manufacturer name ⓘ

My Awesome Company

Device type ⓘ

Please select the device type

Device Model id ⓘ

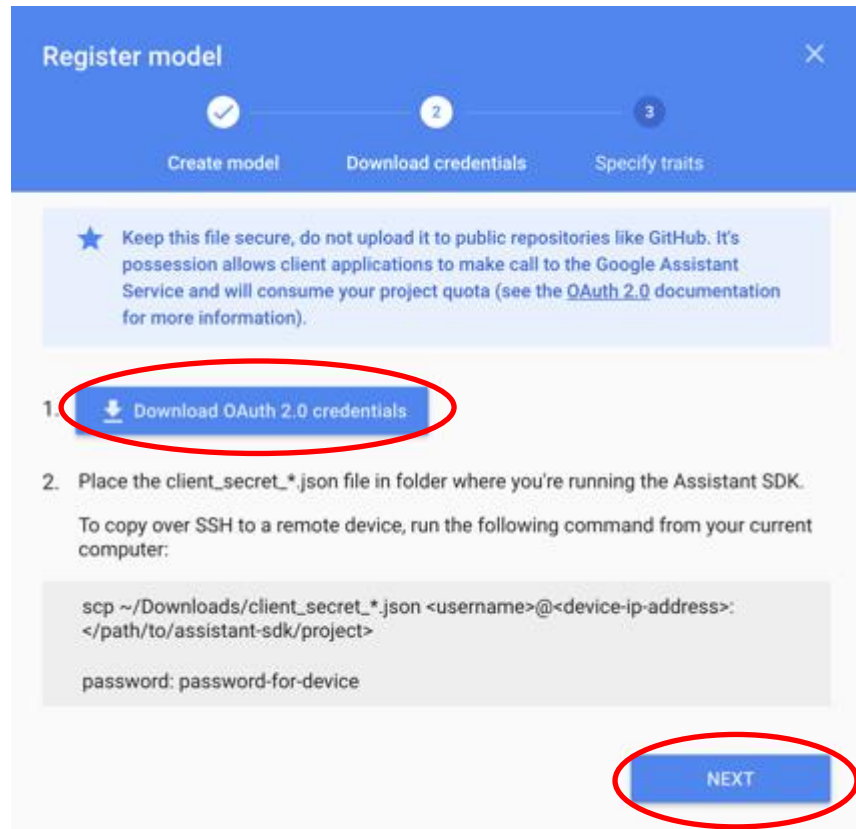
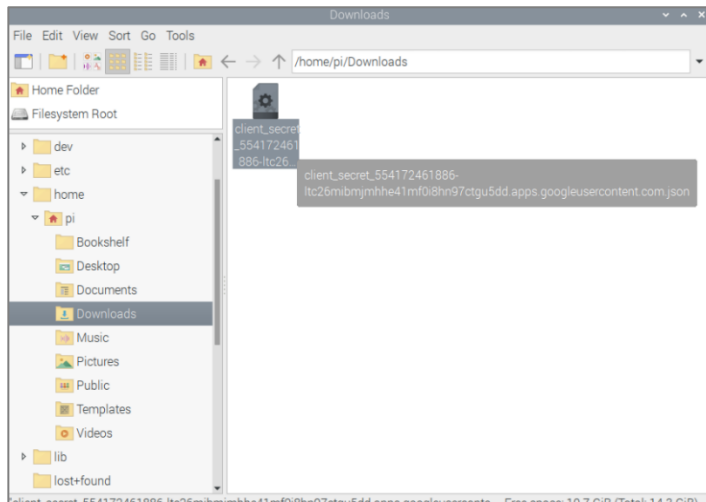
my-awesome-product-id

CANCEL REGISTER MODEL



장치 모델 등록

- 인증서 다운로드
 - client_secret_<client-id>.json
파일은 라즈베리파이 장치에 있어야 함





장치 모델 등록

- 특성 지정
 - 차후 장치가 지원하는 다양한 기능을 지정
 - 지금은 SKIP

Register model

Create model Download credentials Specify traits

Search All traits

☐ All 7 traits

☐ Brightness
This trait covers how to control the brightness of a device. Absolute brightness setting is in a normalized range from 0 to 100 (individual lights may not support every point in the range based on their LED configuration).
[View details](#)

☐ ColorSpectrum
This trait belongs to any device that is able to set a color spectrum. This applies to 'full' color bulbs that take RGB color space. Lights may have any combination of ColorSpectrum and

SKIP SAVE TRAITS



장치 모델 등록

- 모델 편집

					REGISTER MODEL
Product name	Manufacturer name	Model ID	Device Type	Last updated time	
Assistant SDK light	Assistant SDK developer	jupiter-86b58-assistant-sdk-light-xw2x1y	action.devices.types.LIGHT	Jul 13, 2018, 02:20 PM	

Linked device models

← Assistant SDK light SAVE ⋮

Product name

Manufacturer name

Model id

Device type

Download OAuth 2.0 credentials

Remove model



장치 모델 등록

- 라즈베리파이에서 실행 (json 파일을 라즈베리파이에서 저장하는 작업)

- 새 터미널 창 열기

```
$ scp ~/Downloads/client_secret_<i>client-id</i>.json pi@<i>raspberry-pi-ip-address</i>:/home/pi/  
password: <i>password-for-device</i>
```

- WinSCP 프로그램을 사용하면 간단하게 파일을 이동할 수 있음

WinSCP 설치하기 : <https://winscp.net/eng/download.php>

SCP 프로그램 설치 후 윈도우즈에서 json 파일을 라즈베리파이로 복사 실행



SDK 및 샘플 코드 설치

- 파이썬 가상환경

※ Raspberry Pi의 경우 **/home/pi** 디렉토리에서 다음 명령을 실행하십시오 .

```
$ sudo apt-get update
```

```
$ sudo apt-get install python3-dev python3-venv
```

```
$ python3 -m venv env
```

```
$ env/bin/python -m pip install --upgrade pip setuptools wheel
```

```
$ source env/bin/activate
```

```
(env) $
```



SDK 및 샘플 코드 설치

- 패키지받기

```
(env) $ sudo apt-get install portaudio19-dev libffi-dev libssl-dev
```

```
(env) $ python -m pip install --upgrade google-assistant-sdk[samples]
```

- 인증받기

```
(env) $ python -m pip install --upgrade google-auth-oauthlib[tool]
```

```
(env) $ google-oauthlib-tool --scope https://www.googleapis.com/auth/assistant-sdk-prototype \ --save --  
headless --client-secrets /home/pi/client_secret_client-id.json
```

Please visit this URL to authorize this application: https://...

Enter the authorization code:



SDK 및 샘플 코드 설치

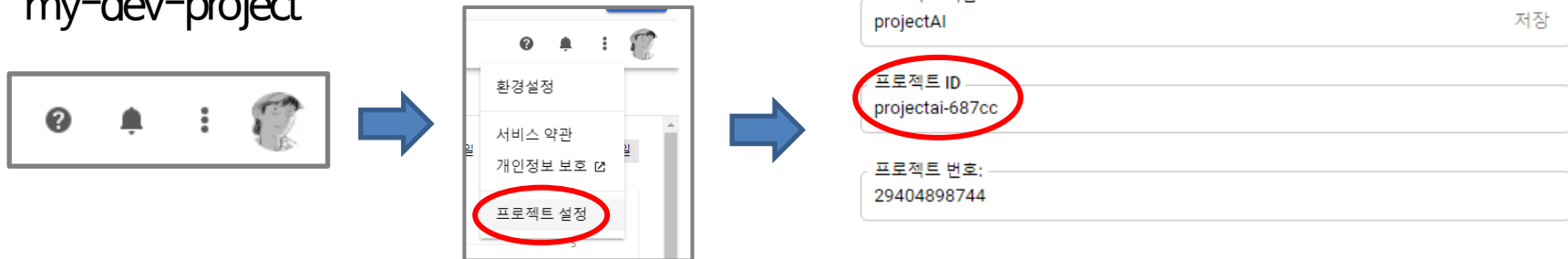
```
pi@raspberrypi: ~  
File Edit Tabs Help  
(env) pi@raspberrypi:~ $ google-oauthlib-tool --scope https://www.googleapis.com  
/auth/assistant-sdk-prototype \  
> --save --headless --client-secrets /home/pi/client_secret_554172461886-l  
tc26mibmjmhhe41mf0i8hn97ctgu5dd.apps.googleusercontent.com.json  
Please visit this URL to authorize this application: https://accounts.google.com  
/o/oauth2/auth?response_type=code&client_id=554172461886-ltc26mibmjmhhe41mf0i8hn  
97ctgu5dd.apps.googleusercontent.com&redirect_uri=urn%3Aietf%3Aawg%3Aoauth%3A2.0%  
3Aoob&scope=https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fassistant-sdk-prototype&st  
ate=LR8mBPQK0zU97yZVtfGHYE1iRACuag&prompt=consent&access_type=offline  
Enter the authorization code: 
```

복사해서 웹 브라우저에 붙여넣기

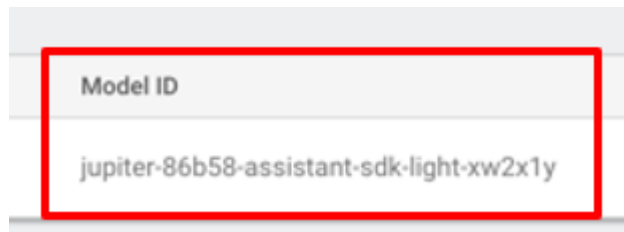


project-id & device-model-id

- my-dev-project



- device-model-id





샘플 코드 실행

- 아래와같이 실행

```
(env) $ ./googlesamples-assistant-pushtotalk --project-id my-dev-project --device-model-id my-model
```

- 장치 인스턴스 ID 찾기

```
INFO:root:Connecting to embeddedassistant.googleapis.com
WARNING:root:Device config not found: [Errno 2] No such file or directory: '/home/pi/.config/googlesamples-assistant/device_config.json'
INFO:root:Registering device
INFO:root:Device registered: 0eea18ae-d17e-11e7-ac7a-b827ebb8010f # Device instance ID
Press Enter to send a new request...
```

Raspberry Pi를 활용한 IoT 프로젝트

Thank You