

앱인벤터 AI

김경민 교수



부산대학교 소프트웨어교육센터
PUSAN NATIONAL UNIVERSITY SOFTWARE EDUCATION CENTER

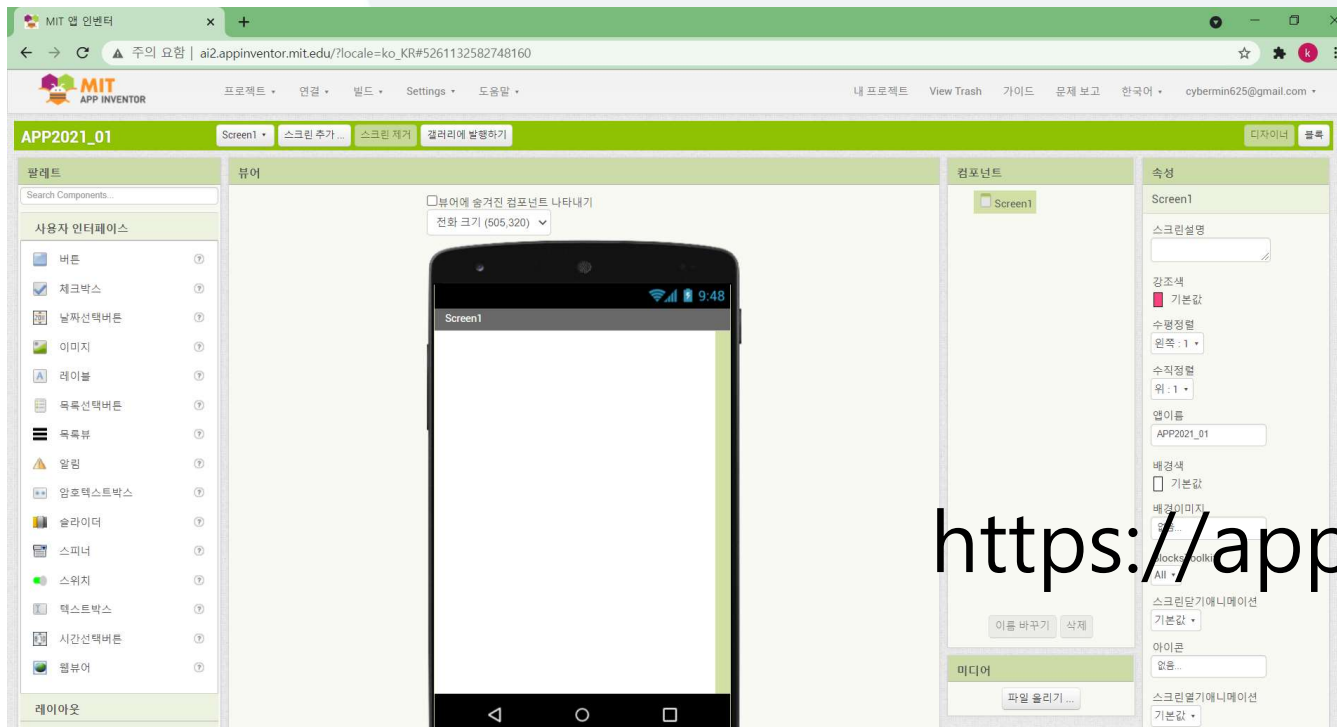
SW중심대학





인공지능 스피커

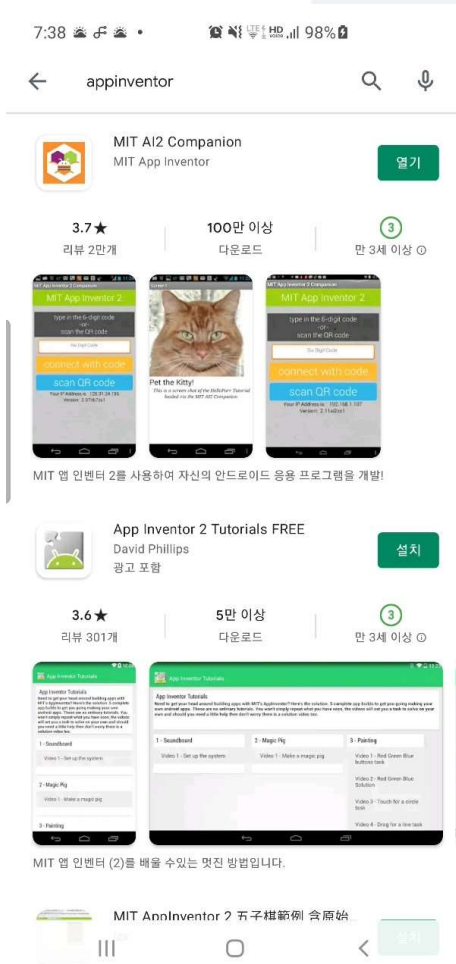
앱인벤터 앱인벤터(App Inventor)



<https://appinventor.mit.edu/>

- 안드로이드의 어플리케이션을 개발하는 툴(도구)
- 앱인벤터 파일 : aia
- 앱설치 파일 : apk

앱인벤터 앱인벤터(App Inventor) 실행



<http://appinventor.mit.edu/explore/ai2/setup.html>

옵션 1 - 권장 WiFi 연결을 사용하여 iOS 또는 Android 장치로 앱 빌드(선택): **지침**

컴퓨터, 스마트폰 또는 태블릿이 있고 WiFi 연결이 있는 경우 기기에서 앱을 빌드하고 테스트하는 가장 쉬운 방법입니다.



옵션 2 Chromebook으로 앱 빌드: **지침**

많은 Chromebook에서 Android 앱을 실행할 수 있습니다. 이를 통해 동일한 장치에서 완성된 앱을 빌드하고 실행할 수 있습니다.



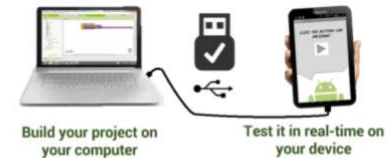
옵션 3 장치가 없습니까? 에뮬레이터 사용: **지침**

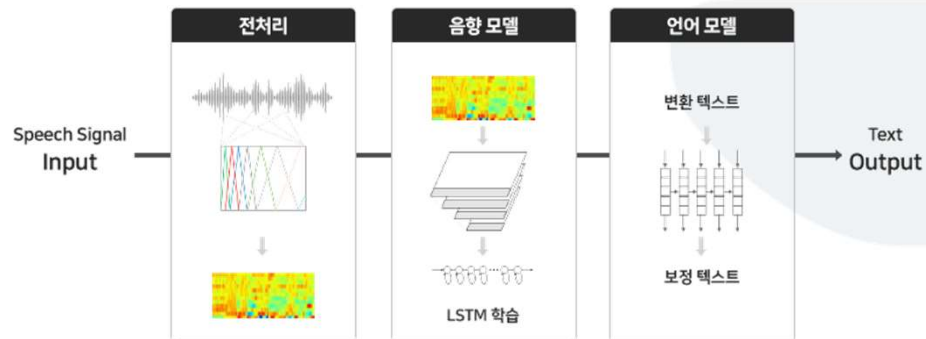
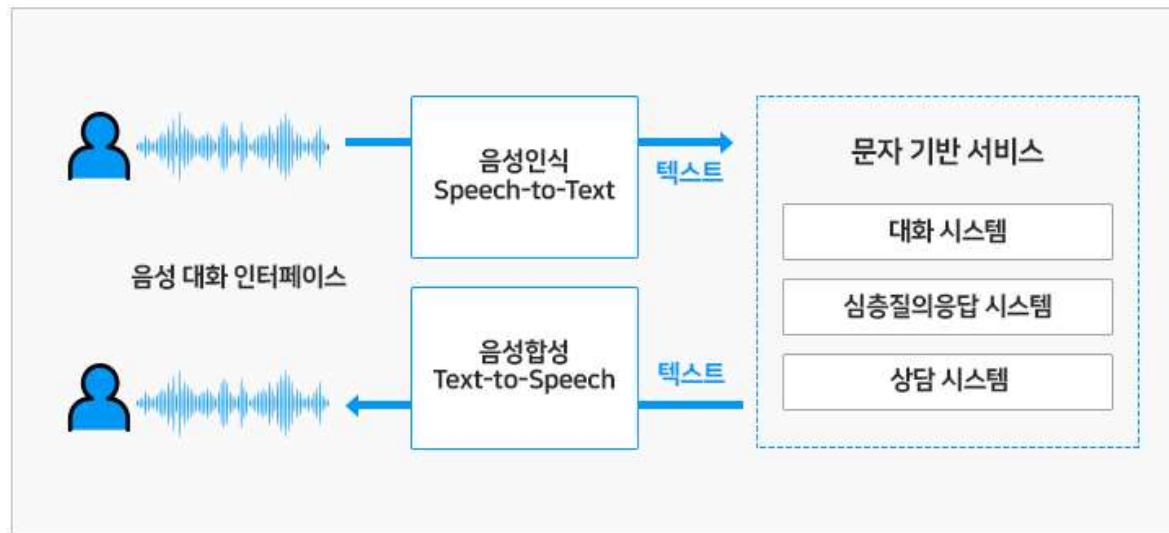
휴대전화나 태블릿이 없는 경우에도 App Inventor를 사용할 수 있습니다. 30명의 학생이 수업을 듣습니까? 에뮬레이터에서 주로 작업하고 몇 가지 장치를 공유하도록 합니다.



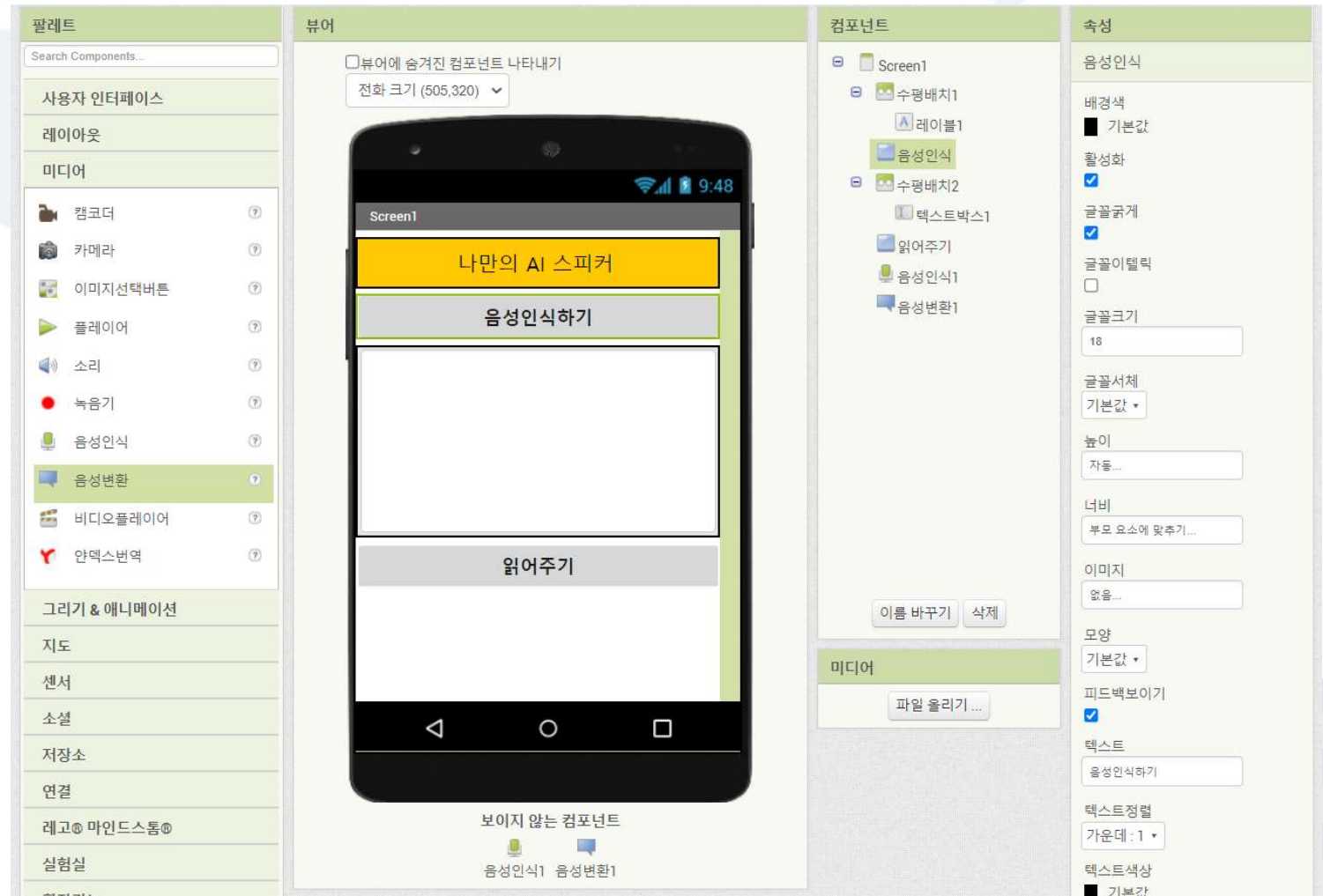
옵션 4 WiFi가 없습니까? Android 기기 및 USB 케이블로 앱 빌드: **지침**

학교 및 조직 내의 일부 방화벽은 필요한 유형의 WiFi 연결을 허용하지 않습니다. WiFi가 작동하지 않으면 USB를 사용해 보십시오.





실습 인공지능 스피커 만들기



실습 인공지능 스피커 만들기



언제 음성인식 .클릭했을때
실행 호출 음성인식1 .텍스트가져오기

언제 음성인식1 .텍스트가져온후에
결과 partial
실행 지정하기 텍스트박스1 . 텍스트 값 가져오기 결과

전역변수 만들기 말하기 초기값 " "

언제 읽어주기 .클릭했을때
실행 만약
 텍스트박스1 . 텍스트 = " "
 이러면 실행 지정하기 전역변수 말하기 값 " 읽어줄 내용이 없습니다. "
 아니라면 지정하기 전역변수 말하기 값 텍스트박스1 . 텍스트
 호출 음성변환1 .말하기
 메시지 가져오기 전역변수 말하기

실습

인공지능 스피커 만들기

인공지능 스피커 앱으로 자기소개

실습

인공지능 스피커 만들기

인공지능 스피커를 활용한 앱 만들기

실습 인공지능 Call 만들기

9:42 88%

Screen1

AI Call

음성인식하기 ①

텍스트박스1의 힌트

텍스트박스2의 힌트

전화걸기

문자보내기

9:53 89%

Screen1

AI Call

음성인식하기

텍스트박스1의 힌트

Google

아무 말이나 해 보세요.

한국어 (대한민국)

'전화'가 인식되었을 때

'문자'가 인식되었을 때

음성안내 후 음성입력

전화/문자를 인식하지 못했을 때

9:57 89%

Screen1

AI Call

음성인식하기

안녕하세요

텍스트박스2의 힌트

전화걸기

문자보내기

9:55 89%

경민김
010-2833-4613

010-2833-4613

1 2 3
4 5 6
7 8 9
* 0 #

9:57 89%

Screen1

AI Call

음성인식하기

010-2833-4613 문자

Google

아무 말이나 해 보세요.

한국어 (대한민국)

Google Speech 서비스가 오디오를 텍스트로 변환하고 텍스트를 이 앱에 공유합니다.

전송할 문자 음성인식

9:56 89%

< 경민김
010-2833-4613

1 [국경발신] Your Google verification code is 312393 오늘 1:12

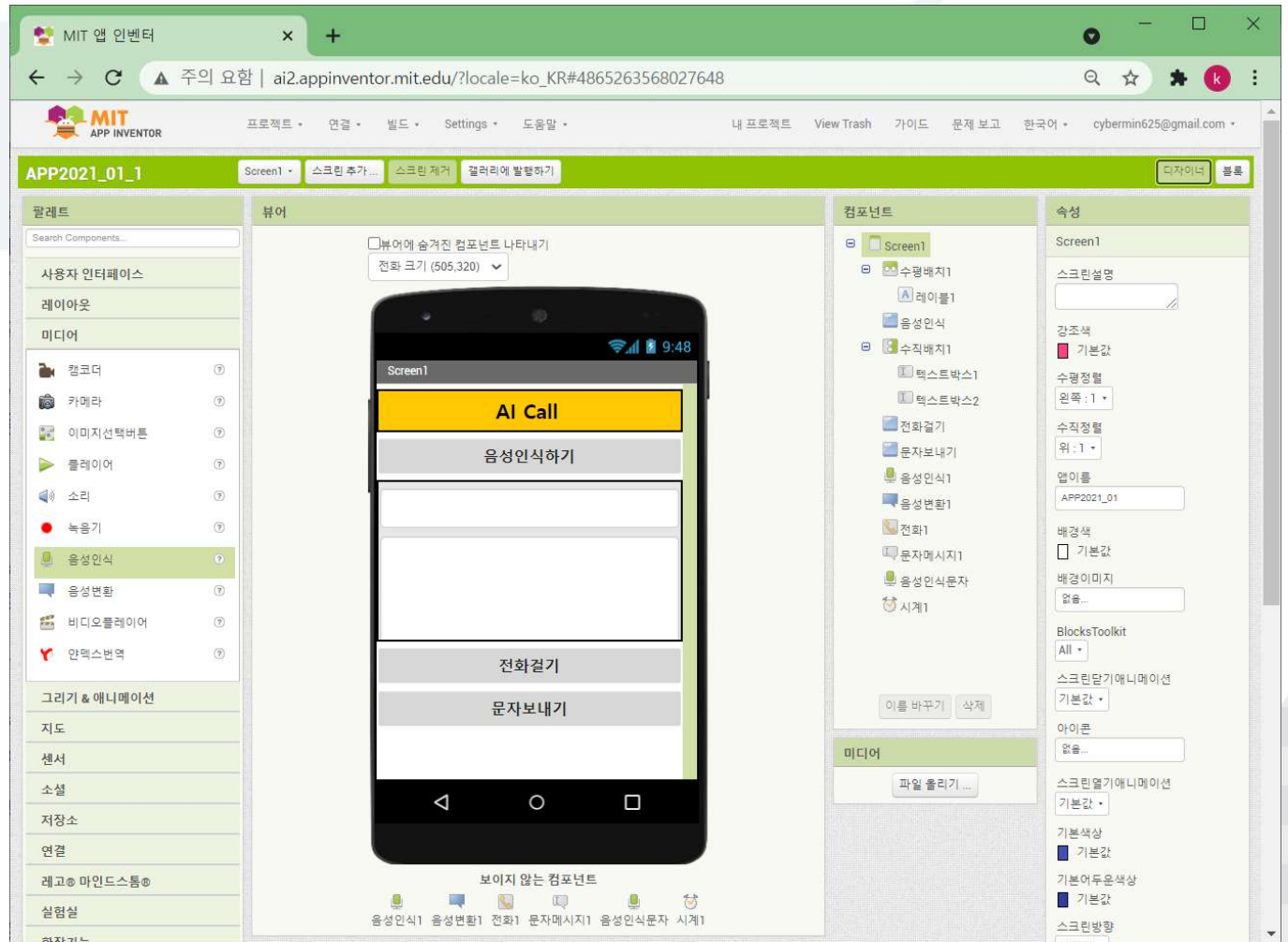
2 [KT] KT 보안 인증번호 [628778]를 입력해 주세요. 오늘 10:28

3 [Web발신] KT>서서당메시지가 도착했습니다. 연결은 음성통화입니다 (신규이전제01). 010-2833-4613 오늘 3:21

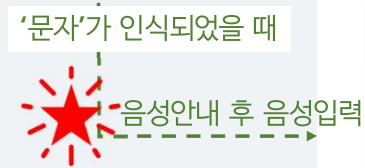
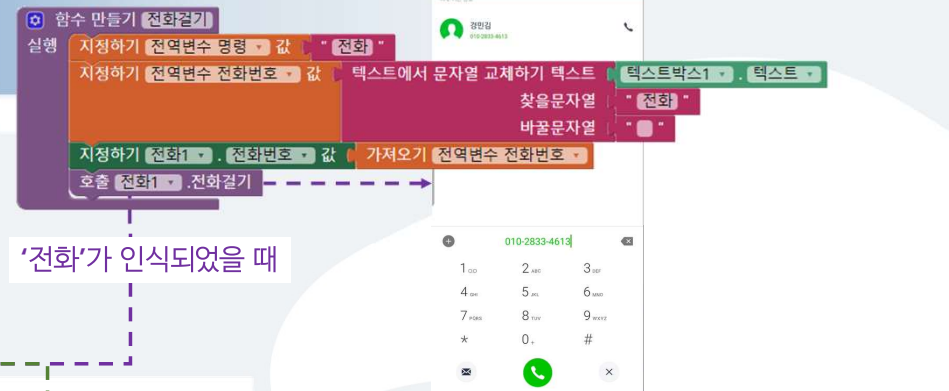
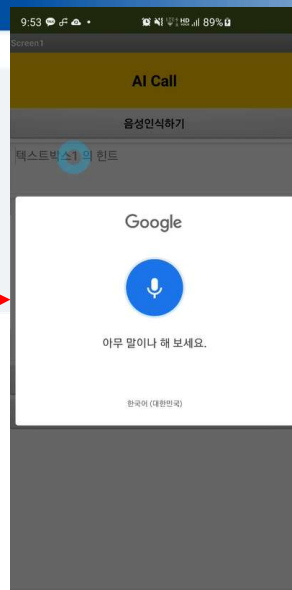
130143

이미 전송할 문자가 있을 경우

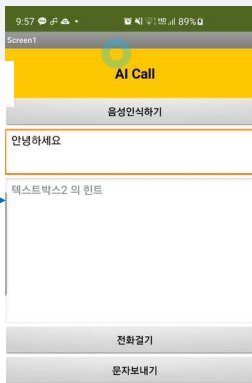
실습 인공지능 Call 만들기



실습 인공지능 Call 만들기



전화/문자를 인식하지 못했을 때





인공지능 번역기

인공지능 번역기

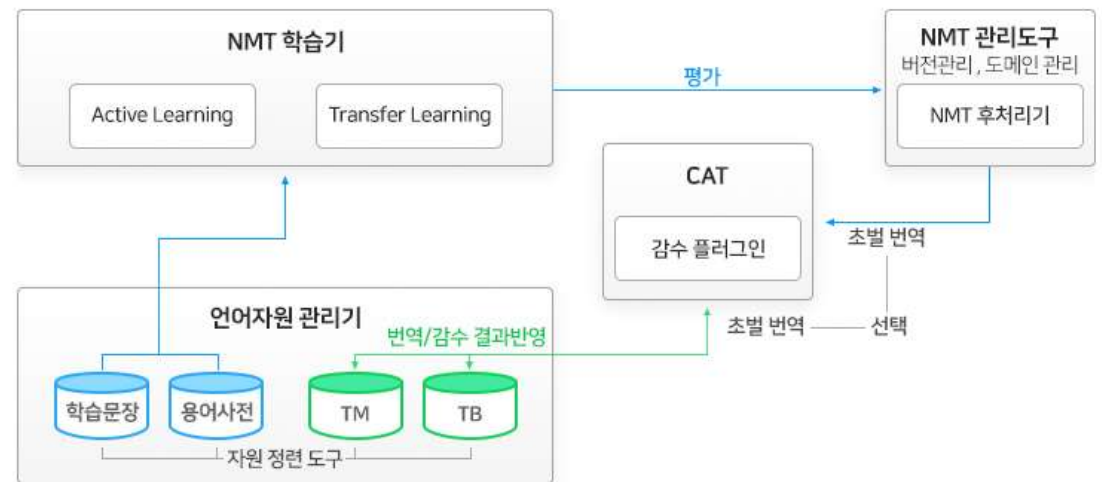
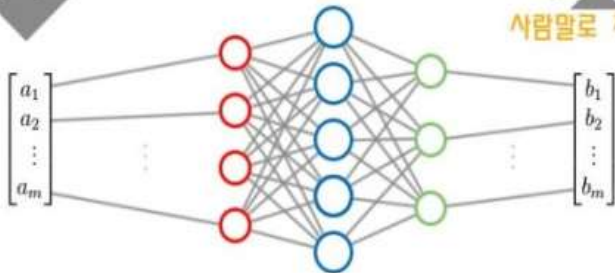
인공지능 번역기

Neural machine translation (NMT) is the approach to machine translation in which a large neural network is trained to maximize translation performance.

신경 기계 번역 (NMT)은 번역 성능을 극대화하기 위해 대규모 신경 네트워크를 교육하는 기계 번역 방법입니다.

기계어로 재조합

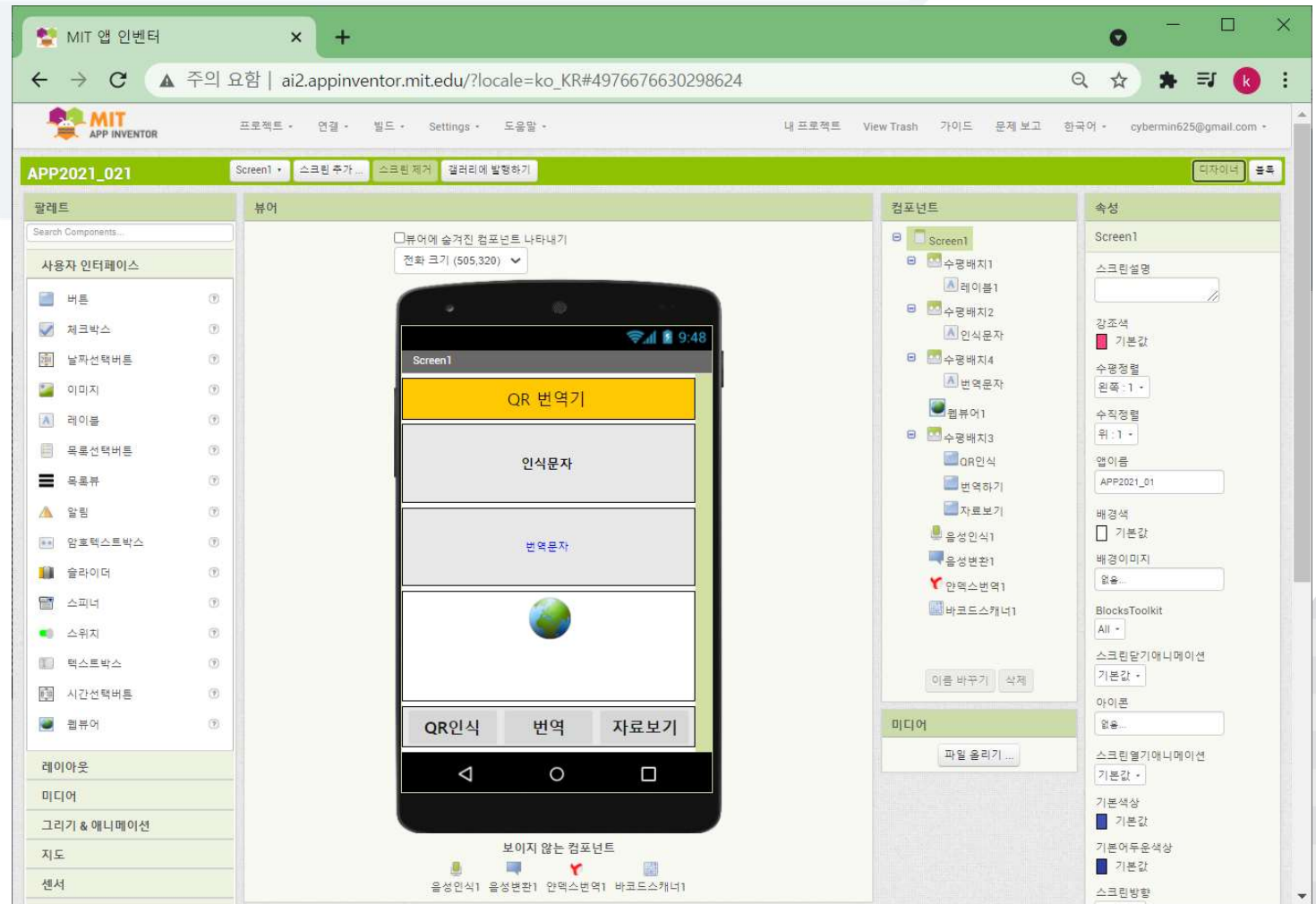
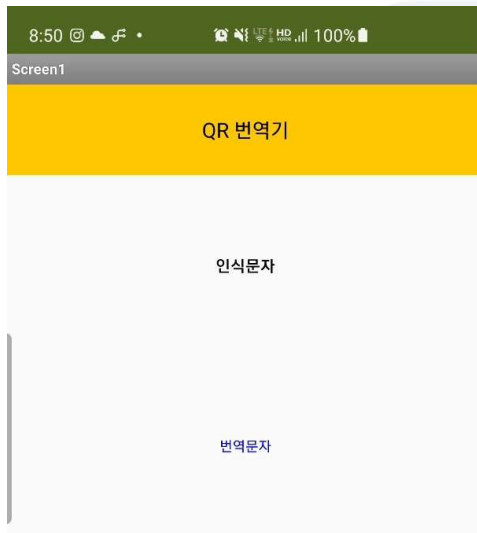
사람말로 재조합



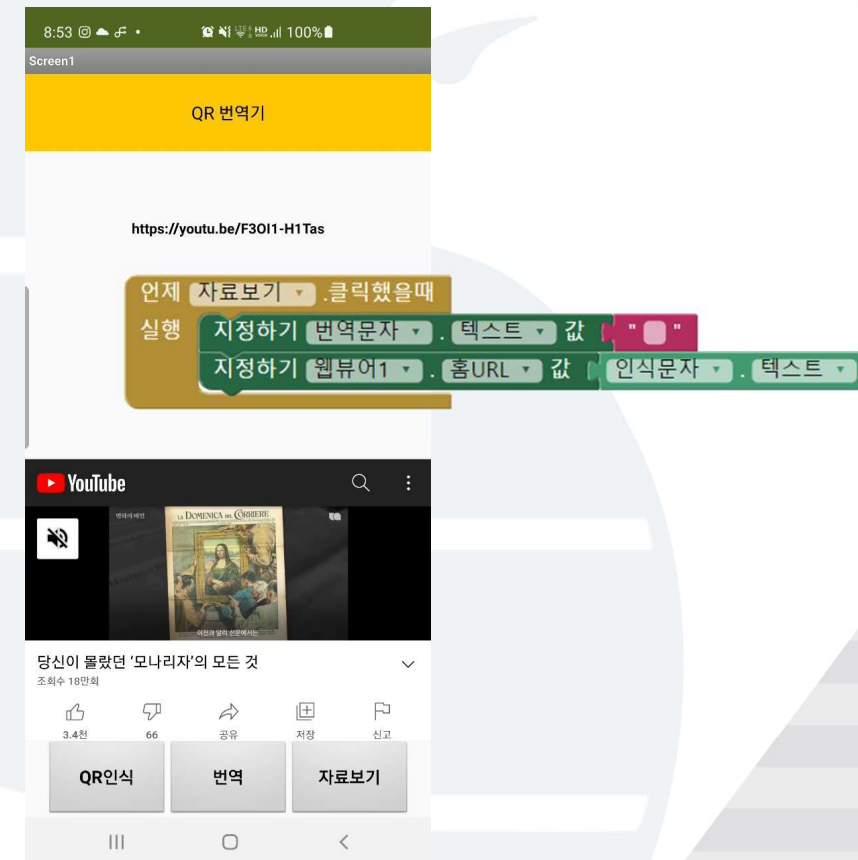
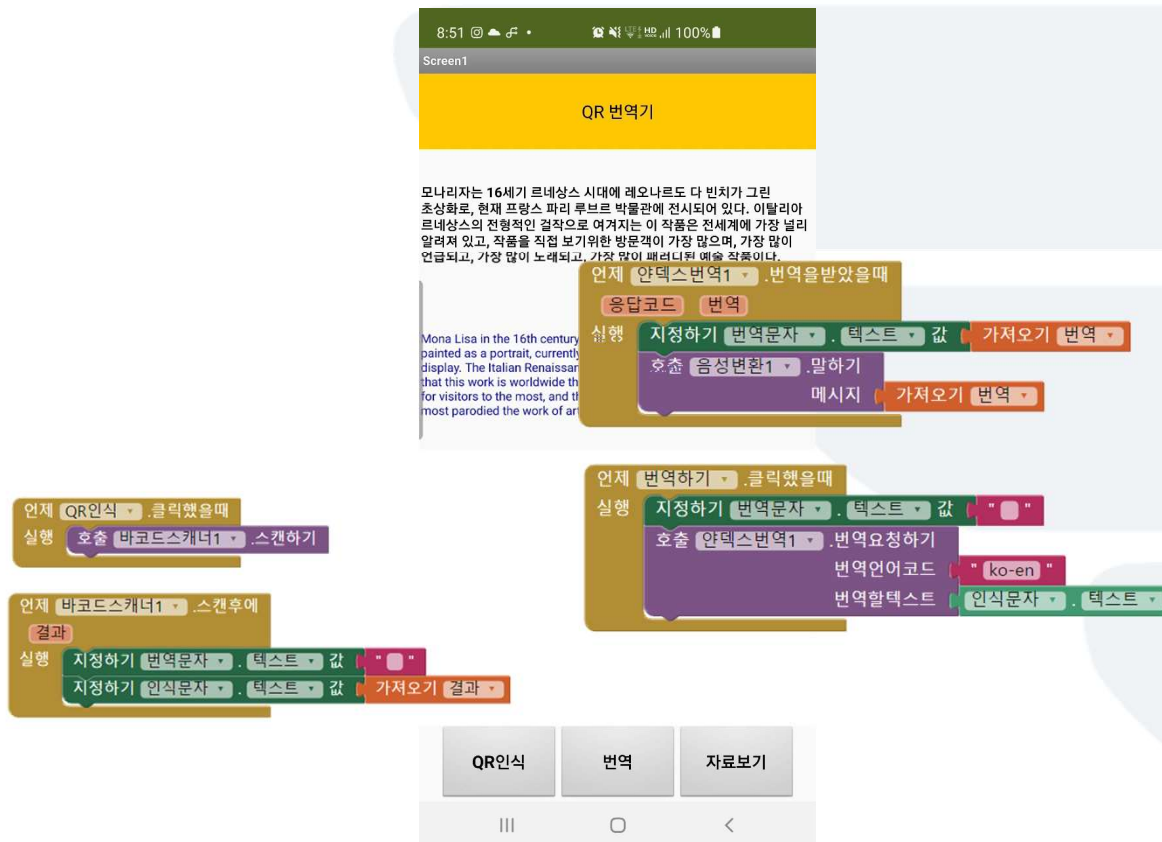
<https://hub.zum.com/heraldcorp/6727>

<http://www.saltlux.com/ai/neuralNetwork.do?menuNumber=1>

실습 인공지능 번역기 만들기



실습 인공지능 번역기 만들기



참고링크 : <https://sites.google.com/view/appmona/%ED%99%88>

실습

인공지능 번역기

인공지능 번역기를 활용한 앱 만들기



인공지능 마스크인식

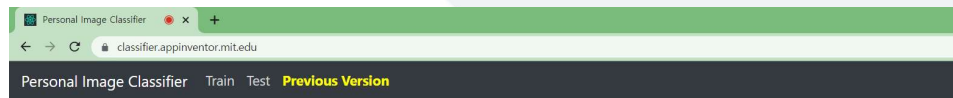
사람이 학습하는 것처럼 제공된 데이터를 입력으로
스스로 학습을 통해 예측을 할 수 있도록 모델을 구축하여
이 모델을 통해서 입력되지 않은 정보에 대해서도 판단을 할 수 있도록 함



인공지능 마스크인식

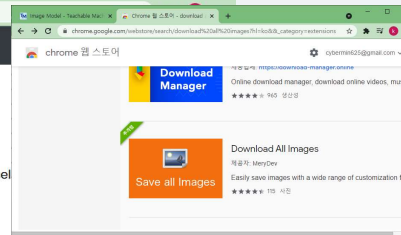
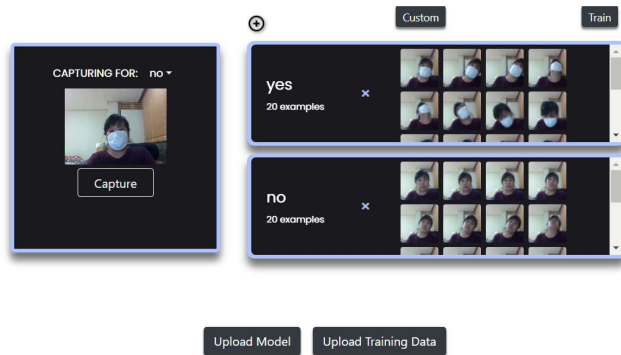
앱인벤터 학습모델 - PersonallImage Classifier

<https://classifier.appinventor.mit.edu/>



Training Page

To get started, click the plus icon to add a classification and then use the "Capture" button or drag images into the capture box to add images to the set. You can also upload previously generated data and models using the buttons below. When done, hit "Train"



- 크롬 웹스토어
<https://chrome.google.com/webstore/category/extensions?hl=ko&>
- Download All Images 추가

이미지 모으기

모델 테스트

학습(Train)

학습모델 내보내기

인공지능 마스킹인식

앱인벤터 학습모델 확장기능 추가



<https://mit-cml.github.io/extensions/>

MIT App Inventor Extensions

This is the official resource for the MIT App Inventor Extensions, use them within your own projects. Explore, create and share new functionality through App Inventor Extensions.

[Try App Inventor Extensions >](#)

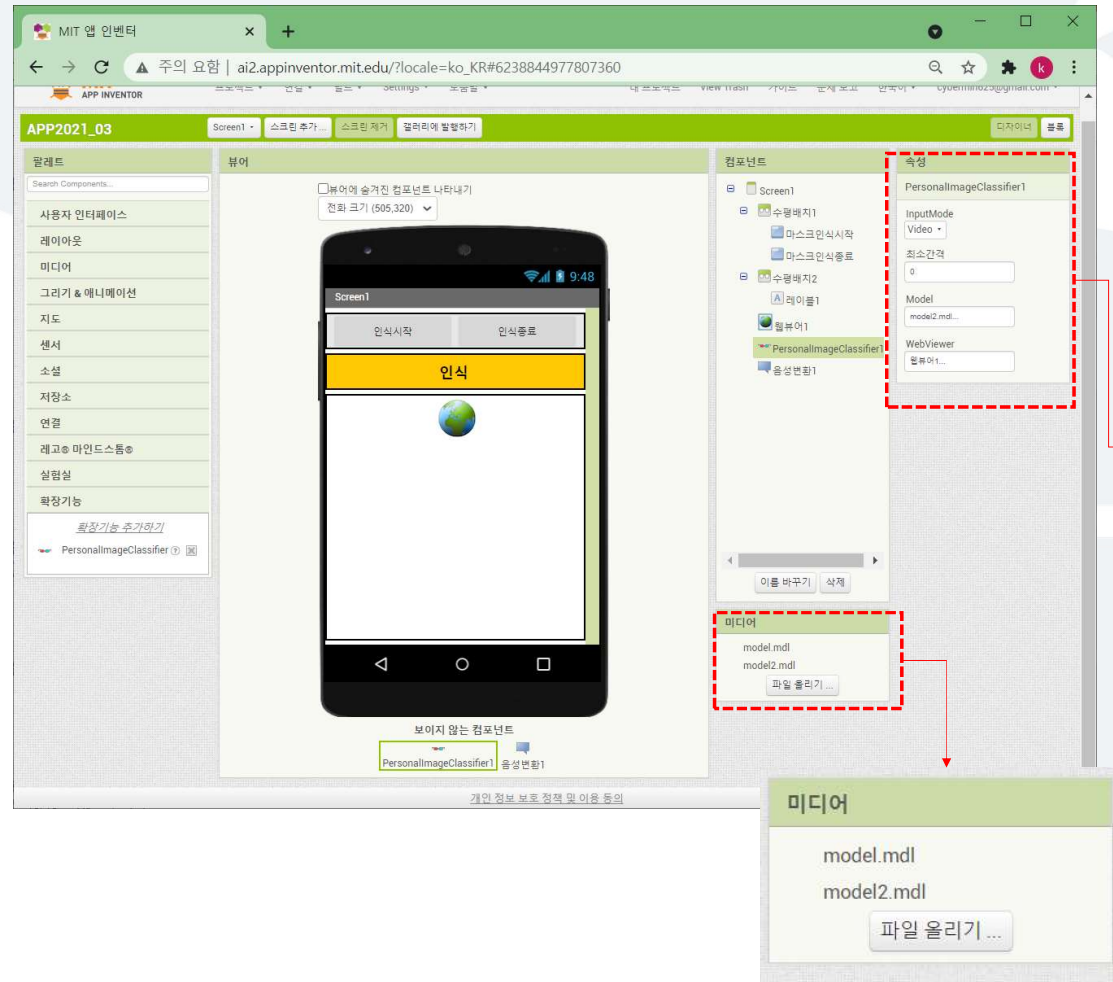
Note: Make sure you have the latest MIT AI2 Companion application. Click on Help in the menu bar of your App Inventor screen and select Companion Information to get a QR code or link that can be used to download and install the MITAI2Companion.apk on your phone.

Supported:

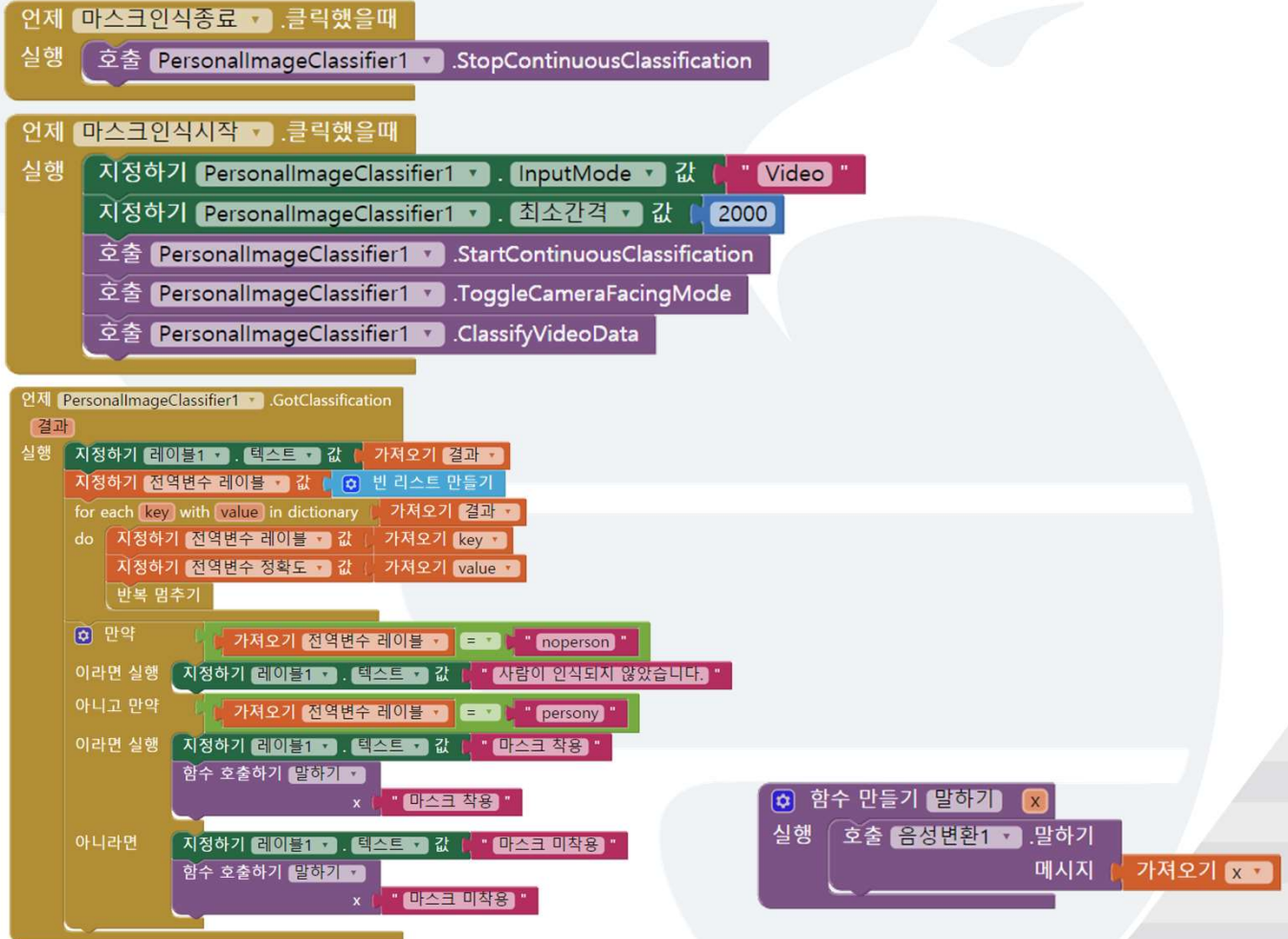
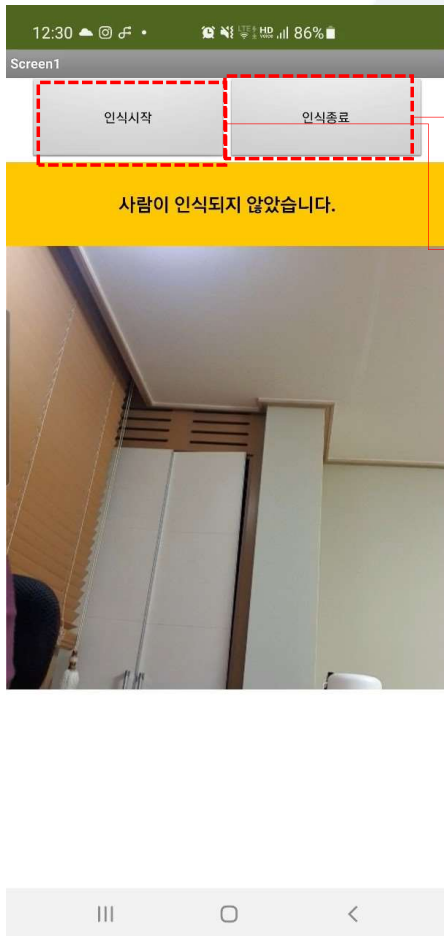
Name	Description	Author	Version	Download .aix File	Source Code
BluetoothLE	Adds as Bluetooth Low Energy functionality to your applications. See BluetoothLE Documentation and Resources for more information.	MIT App Inventor	20200828	BluetoothLE.aix	Via GitHub
LookExtension	Adds object recognition using a neural network compiled into the extension.	MIT App Inventor	20181124	LookExtension.aix	Via GitHub
PersonalAudioClassifier	Use your own neural network classifier to recognize sounds with this extension.	MIT App Inventor	20200904	PersonalAudioClassifier.aix	Via GitHub
PersonallImageClassifier	Use your own neural network classifier to recognize images with this extension.	MIT App Inventor	20210315	PersonallImageClassifier.aix	Via GitHub
PosenetExtension	Estimate pose with this extension.	MIT App Inventor	20200226	Posenet.aix	Via GitHub
FaceMeshExtension	Estimate face landmarks with this extension.	MIT App Inventor	20210414	Facemesh.aix	Via GitHub

Note: The BluetoothLE extension was made possible, in part, by a grant given by the University Program Office at Intel Corporation.

인공지능 마스크인식 앱인벤터 마스크인식기



인공지능 마스크인식 앱인벤터 마스크인식기



실습

인공지능 마스크인식기

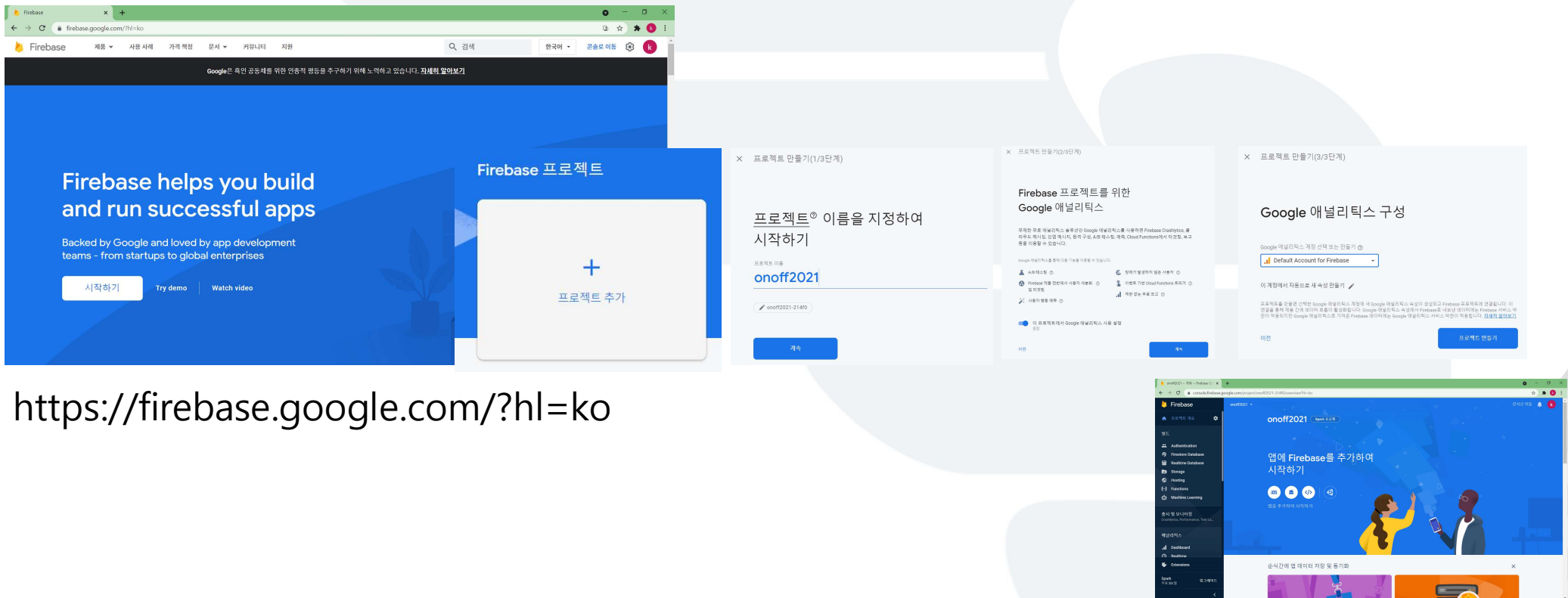
인공지능 학습모델을 활용한 앱 만들기

A stylized graphic of a bird in flight, rendered in light blue and white. The bird is positioned on the right side of the frame, facing left. Its wings are spread, and several horizontal, rounded rectangular bars of varying lengths are layered across the background, creating a sense of motion or a stylized representation of feathers. The overall aesthetic is clean and modern.

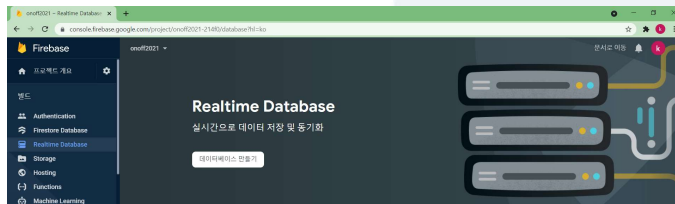
파이어베이스와 연동

파이어베이스(Firebase)

- **2011년** 파이어베이스(**Firestore, Inc**)사가 개발하고 **2014년** 구글에 인수된 모바일 및 웹 애플리케이션 개발 플랫폼



파이어베이스와 연동



데이터베이스 설정

1 데이터베이스 옵션

2 보안 규칙

위치 설정은 실시간 데이터베이스 데이터가 저장되는 위치입니다.

실시간 데이터베이스 위치

미국(us-central1)

취소 다음

데이터베이스 설정

1 데이터베이스 옵션

2 보안 규칙

데이터 구조를 정의한 후 데이터의 보안을 강화하는 규칙을 작성해야 합니다.
[자세히 알아보기](#)

☒ **잠금 모드**로 시작

데이터는 기본적으로 비공개됩니다. 클라이언트 읽기/쓰기 액세스 권한은 보안 규칙에서 지정한 대로만 부여됩니다.

☐ **테스트 모드**에서 시작

백트 설정을 위해 기본적으로 데이터가 공개됩니다. 하지만 90일 내에 보안 규칙을 업데이트하여 장기적 클라이언트 읽기/쓰기 액세스 권한을 사용 설정해야 합니다.

```
{  "rules": {    ".read": false,    ".write": false  }}
```

모든 제3자 읽기 및 쓰기가 거부됩니다.

취소 사용 설정

Realtime Database

데이터 규칙 백업 사용량

규칙 수정 규칙 모니터링

게시되지 않은 변경사항

게시

삭제

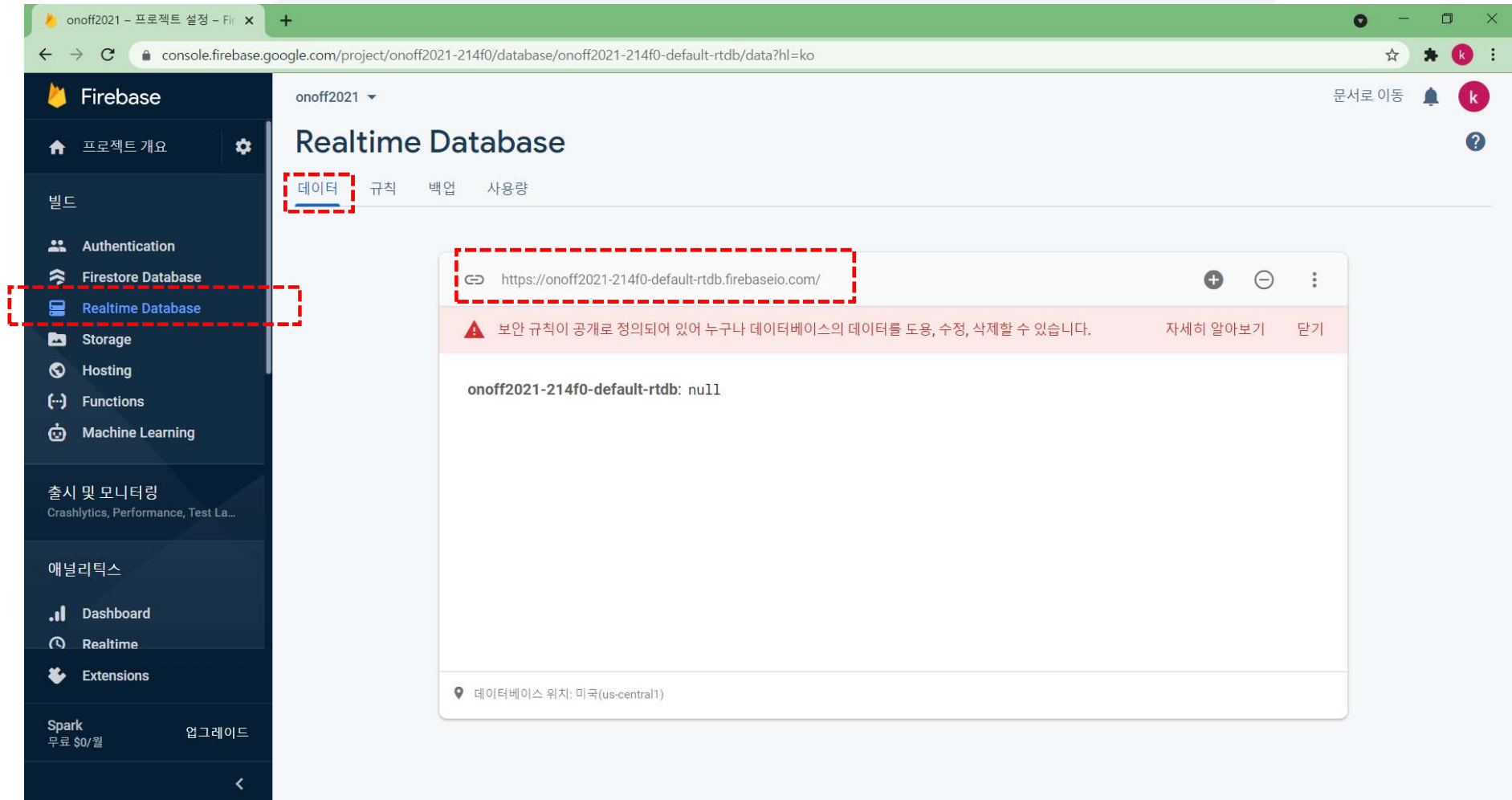
```
1 {  2   "rules": {  3     ".read": true,  4     ".write": true  5   }  6 }
```

파이어베이스와 연동

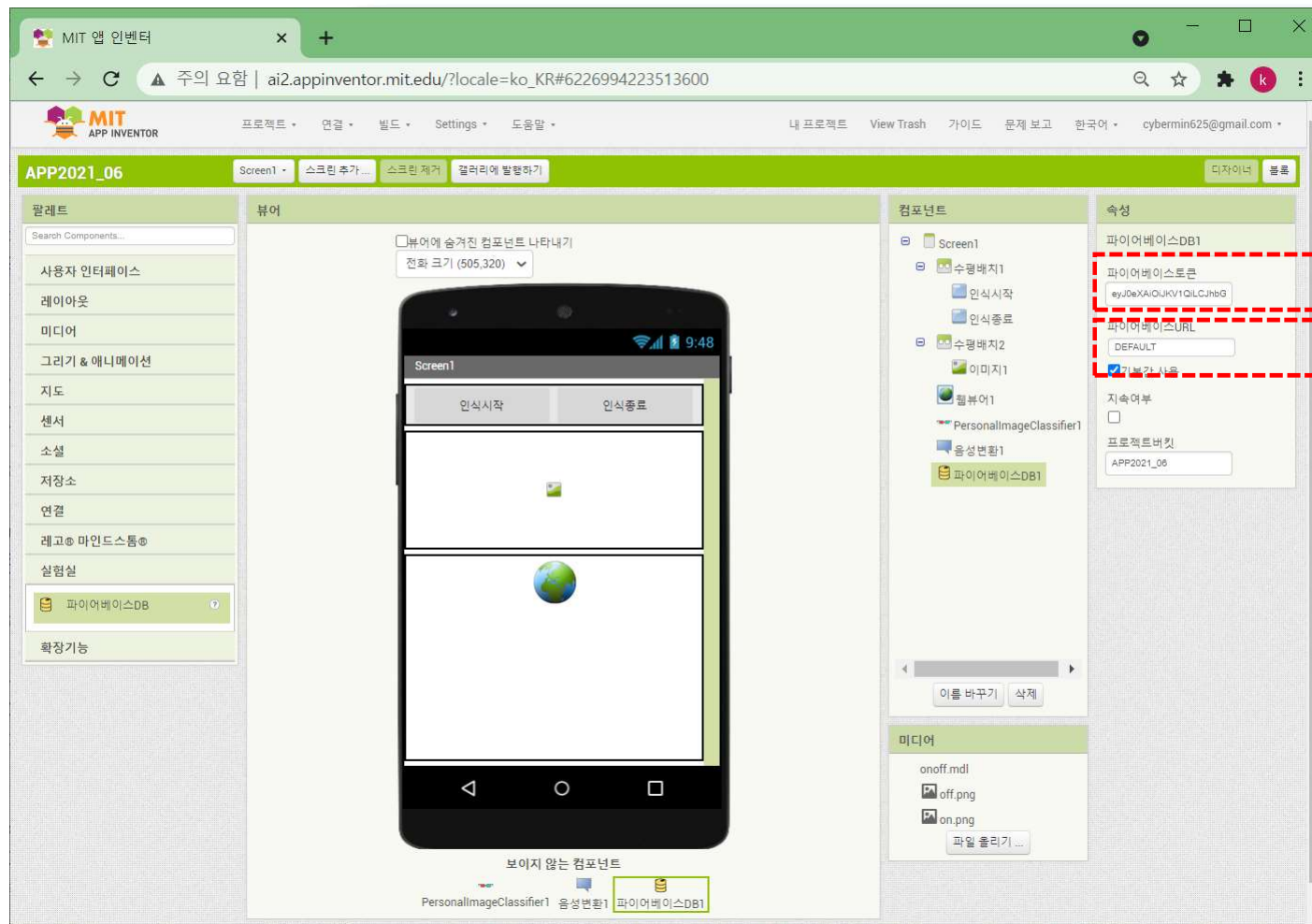
The screenshot shows the Firebase console interface for the project 'onoff2021'. The left sidebar contains the 'Project settings' (프로젝트 설정) menu, which is highlighted with a red dashed box. The 'Service accounts' (서비스 계정) tab is also highlighted with a red dashed box. The main content area displays the 'Database secrets' (데이터베이스 비밀번호) section. A warning message states: '데이터베이스 비밀번호는 현재 지원이 중단되었으며 기존 Firebase 토큰 생성기를 사용합니다. 소스 코드를 Firebase Admin SDK로 업데이트하세요.' (Database passwords are no longer supported, and the existing Firebase token generator should be used. Update the source code to the Firebase Admin SDK). Below the warning, there is a table of database secrets. The table has two columns: 'Database' (데이터베이스) and 'Secret' (비밀번호). The first row shows the database 'onoff2021-214f0-default-rtdb' and its corresponding secret, which is highlighted with a red dashed box. The secret is represented by a series of dots. A 'Add secret' (비밀번호 추가) button is visible in the top right corner of the table.

데이터베이스	비밀번호
onoff2021-214f0-default-rtdb

파이어베이스와 연동



파이어베이스와 연동



데이터베이스 비밀번호

RealtimeDatabase url

파이어베이스 값 추가

Realtime Database

데이터 규칙 백업 사용량

https://onoff2021-214f0-default-rt

⚠ 보안 규칙이 공개로 정의되어 있습니다

onoff2021-214f0-default-rt

off: 0
on: 0

Realtime Database

데이터 규칙 백업 사용량

https://onoff2021-214f0-default-rt

⚠ 보안 규칙이 공개로 정의되어 있습니다

onoff2021-214f0-default-rt

off: "10"
on: "1"

언제 PersonallImageClassifier1 .GotClassification

결과

실행 for each key with value in dictionary 가져오기 결과
do 지정하기 전역변수 레이블 값 가져오기 key
지정하기 전역변수 정확도 값 가져오기 value
반복 멈추기

지정하기 이미지1 사진 값 합치기 가져오기 전역변수 레이블

" .png "

지정하기 파이어베이스DB1 프로젝트버킷 값 " "

호출 파이어베이스DB1 .값가져오기
태그 가져오기 전역변수 레이블
찾는태그가없을경우 " "

언제 파이어베이스DB1 .값을받았을때

태그 값

실행 만약 가져오기 값 = " "
이러면 실행 호출 파이어베이스DB1 .값저장하기
태그 가져오기 태그
저장할값 1

아니라면 호출 파이어베이스DB1 .값저장하기
태그 가져오기 태그
저장할값 가져오기 값 + 1

파이어베이스 값 추가

Realtime Database

데이터 규칙 백업 사용량

<https://onoff2021-214f0-default-rtdb.firebaseio.com/>

⚠ 보안 규칙이 공개로 정의되어 있어 누구나 데이터베이스의 데이터를

onoff2021-214f0-default-rtdb

User_Data

2021-08-01 14:19:26: "\"off\""
2021-08-01 14:19:28: "\"off\""
2021-08-01 14:19:30: "\"off\""
2021-08-01 14:19:32: "\"off\""
2021-08-01 14:19:34: "\"off\""
2021-08-01 14:19:36: "\"off\""
2021-08-01 14:19:38: "\"off\""
2021-08-01 14:19:40: "\"off\""
2021-08-01 14:19:42: "\"off\""
2021-08-01 14:20:34: "\"off\""

언제 PersonImageClassifier1 .GotClassification

결과

실행 for each key with value in dictionary 가져오기 결과
do 지정하기 전역변수 레이블 값 가져오기 key
지정하기 전역변수 정확도 값 가져오기 value
반복 멈추기
지정하기 이미지1 사진 값 합치기 가져오기 전역변수 레이블
" .png "
지정하기 레이블1 텍스트 값 호출 시계1 날짜시간형식으로바꾸기
인스턴트 호출 시계1 시스템시간가져오기
패턴 " yyyy-MM-dd HH:mm:ss "
지정하기 파이어베이스DB1 프로젝트버킷 값 " User_Data/ "
호출 파이어베이스DB1 값저장하기
태그 레이블1 텍스트
저장할값 가져오기 전역변수 레이블

파이썬으로 파이어베이스 자료 처리

Python JSON 표준 라이브러리

```
import json
```

#파일 읽기

```
with open('/content/drive/MyDrive/교육/앱인벤터/onoff2021.json', 'r') as fp:  
    data = json.load(fp)
```

```
data = data['User_Data']  
data
```

```
{'2021-08-01 14:19:26': 'off',  
'2021-08-01 14:19:28': 'off',  
'2021-08-01 14:19:30': 'off',  
'2021-08-01 14:19:32': 'off',  
'2021-08-01 14:19:34': 'off',  
'2021-08-01 14:19:36': 'off',  
'2021-08-01 14:19:38': 'off',  
'2021-08-01 14:19:40': 'off',  
'2021-08-01 14:19:42': 'off',  
'2021-08-01 14:20:34': 'off',  
'2021-08-01 14:20:36': 'off',  
'2021-08-01 14:20:38': 'off',  
'2021-08-01 14:20:39': 'off',  
'2021-08-01 14:20:40': 'off',  
'2021-08-01 14:20:42': 'off',  
'2021-08-01 14:20:44': 'off',  
'2021-08-01 14:20:46': 'off',  
'2021-08-01 14:20:48': 'off',  
'2021-08-01 14:20:50': 'off',  
'2021-08-01 14:20:52': 'off',  
'2021-08-01 14:20:55': 'on',  
'2021-08-01 14:20:57': 'on',  
'2021-08-01 14:20:59': 'on',  
'2021-08-01 14:21:01': 'off',  
'2021-08-01 14:21:03': 'on',  
'2021-08-01 14:21:05': 'on'}
```

```
for k, v in data.items():  
    if v == "on" : data[k] = 1  
    else : data[k] = 0
```

data

```
{'2021-08-01 14:19:26': 0,  
'2021-08-01 14:19:28': 0,  
'2021-08-01 14:19:30': 0,  
'2021-08-01 14:19:32': 0,  
'2021-08-01 14:19:34': 0,  
'2021-08-01 14:19:36': 0,  
'2021-08-01 14:19:38': 0,  
'2021-08-01 14:19:40': 0,  
'2021-08-01 14:19:42': 0,  
'2021-08-01 14:20:34': 0,  
'2021-08-01 14:20:36': 0,  
'2021-08-01 14:20:38': 0,  
'2021-08-01 14:20:39': 0,  
'2021-08-01 14:20:40': 0,  
'2021-08-01 14:20:42': 0,  
'2021-08-01 14:20:44': 0,  
'2021-08-01 14:20:46': 0,  
'2021-08-01 14:20:48': 0,  
'2021-08-01 14:20:50': 0,  
'2021-08-01 14:20:52': 0,  
'2021-08-01 14:20:55': 1,  
'2021-08-01 14:20:57': 1,  
'2021-08-01 14:20:59': 1,  
'2021-08-01 14:21:01': 0,  
'2021-08-01 14:21:03': 1,  
'2021-08-01 14:21:05': 1}
```

시각화 모듈

- matplotlib.pyplot : 파이썬에서 매트랩과 유사한 그래프 표시를 가능케 하는 패키지

```
import matplotlib.pyplot as plt
```

```
onoff = list(data.values())
```

```
plt.plot(onoff)
```

```
#눈금을 비워두고 축 눈금과 레이블을 보이지 않게 함  
ax = plt.gca()  
ax.axes.xaxis.set_ticks([])  
ax.axes.yaxis.set_ticks([])
```

```
plt.xlabel('Time')  
plt.ylabel('On Off')  
plt.show()
```

