

Do It Yourself

My AI Speaker for Watson And Home IoT

v1.5

사용권한

본 문서에 대한 서명은 서명은 수행 및 유지 관리에 대한 책임이 있음을 인정하는 것임.

작성자	최의신	일자	2018-04-14
검토자		일자	

본인은 서명으로써 업무활동 범위 내에서 사용될 것을 인가함.

승인자		일자	
-----	--	----	--

제.개정이력

버전	제.개정일자	제.개정 내용	제.개정자
v1.0	2018/04/14	제정	최의신
v1.1	2018/04/28	진행과정 내용 보완 "개발 1도 모르시지만 흔쾌히 이 문서로 실습을 진행해 주신 장선아 부장님께 깊은 감사를 드립니다. 부장님의 수고가 다른 분들께 도움이 되리라 생각합니다. 감사합니다."	최의신
v1.2	2018/05/05	Naver STT삭제, Watson STT 추가	최의신
v1.3	2018/06/04	Meetup 버전으로 수정 - 공공데이터 이용-날씨(온도,미세먼지), 버스도착 - 멀티 IoT 디바이스 지원	최의신
v1.4	2018/06/13	Meetup 실습 내용을 추가	최의신
v1.5	2018/06/23	- 부팅디스크 만들기 수정 - 오디오 설정 추가	최의신

목 차

0. Prologue	4
1. 준비물	5
2 시스템 구성	6
2.1 하드웨어	6
2.2 소프트웨어	7
3. AI 스피커 만들기.....	8
3.1 하드웨어 연결	8
3.2 이미지 다운로드.....	9
3.3 부팅 디스크 만들기	9
3.4 라즈베리파이 부팅	11
3.5 음악 서비스 설정	12
4. AI 스피커 실행하기	15
4.1 준비작업.....	15
4.2 환경설정.....	24
4.3 Watson Conversation 설정	25
4.4 실행.....	27
5. IoT 디바이스 만들기	30
5.1 하드웨어 연결	31
5.2 개발환경.....	32
6. 프로그램 업로드.....	37
6.1 라이브러리 수정.....	37
6.2 WIFI 정보 수정	38
6.3 컴파일 및 업로드	38
6.4 동작 확인	40
7. 리모컨 데이터 입력하기	41
7.1 리모컨 데이터 캡처	41
7.2 IR_RAW 테이블 입력	42
7.3 IR_RAW 테이블 입력 (Meetup)	46
8. Watson Conversation과 IoT 디바이스.....	48
APPEND: 소스코드	50

0. Prologue

얼마전 "[DIY] 인공지능 스피커와 Home IoT" 타이틀로 인공지능 스피커를 만들었다. 약간의 하드웨어와 소프트웨어를 조합하여 그리 어렵지 않게 만들수 있었다.

다음의 URL에서 내용을 확인 할 수 있다.

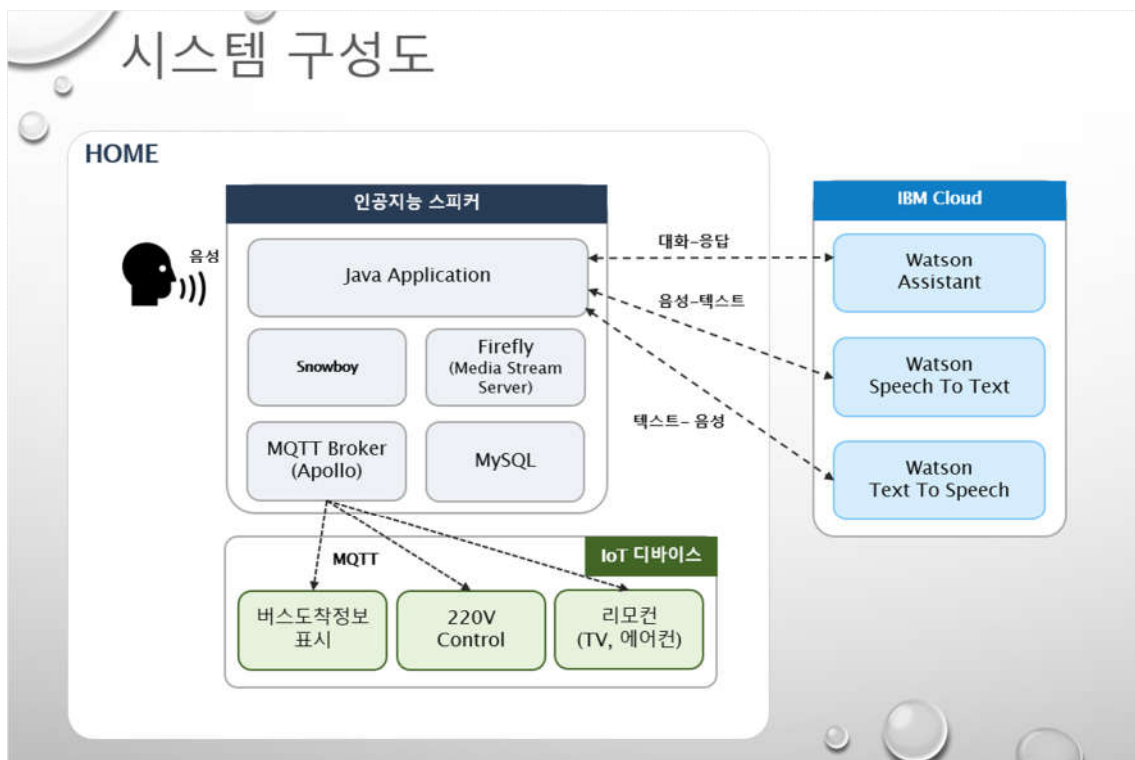
<https://developer.ibm.com/kr/watson/watson-%EC%84%9C%EB%B9%84%EC%8A%A4/2018/02/27/diy-%EC%9D%B8%EA%B3%B5%EC%A7%80%EB%8A%A5-%EC%8A%A4%ED%94%B C%EC%BB%A4%EC%99%80-home-iot/>

이것은 2대의 라즈베리파이 3, 안드로이드 앱, 서비스 서버 등의 구성요소가 많아서 초보자가 따라서 만들기가 쉽지 않다.

그래서 기능은 거의 동일하지만 구성요소 변경을 통해 하나의 라즈베리파이와 약간의 하드웨어로 구성할 수 있는 스피커를 다시 만들었다.

일부 코드는 재사용 되었지만 많은 부분이 바뀌어 새로 개발하였다.

그 이름은 "My AI Speaker for Watson & Home IoT" 이다.



1. 준비물

- micro-SD 카드 리더기



또는



Adapter의 경우 별도의 SD-CARD READER를 준비하거나 PC에 SD-CARD READER 기능이 있는 경우 사용한다.

- micro-SD 8G 이상



- 스피커
- 2A(암페어) 충전기 (Micro 5pin USB 케이블 포함)



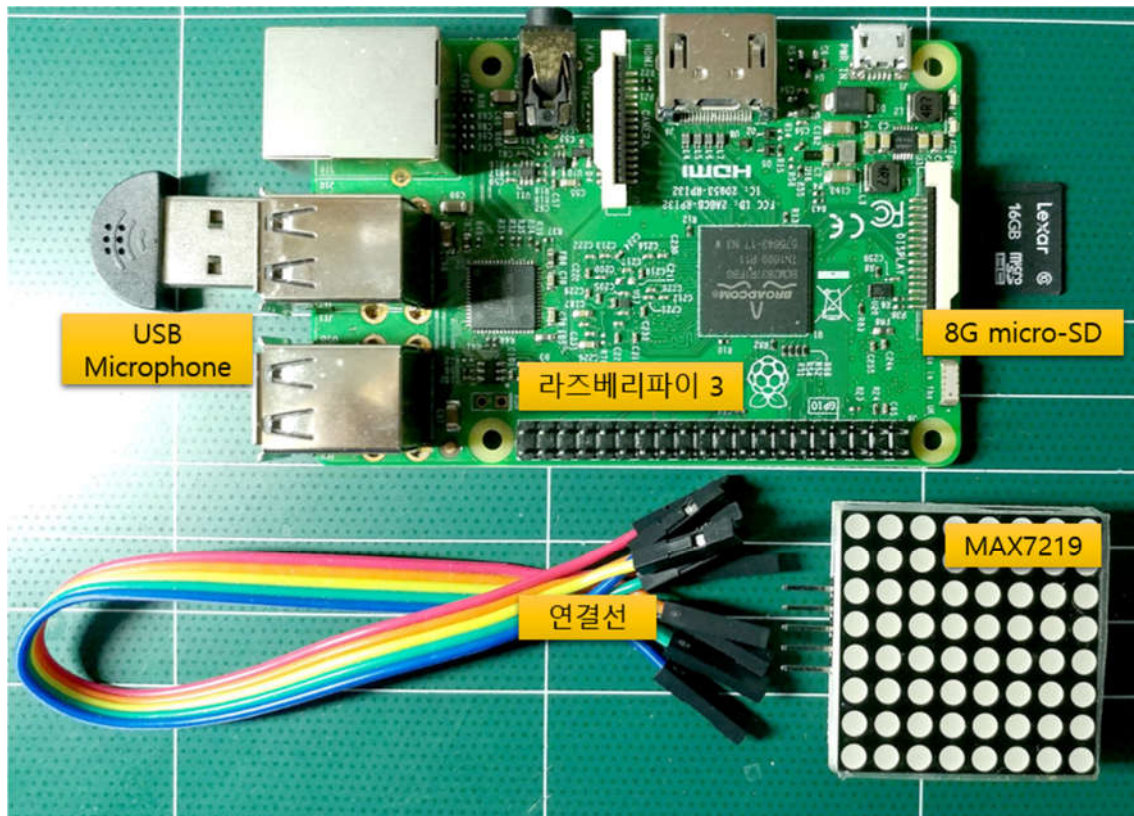
- 리모컨 (TV 또는 에어컨)

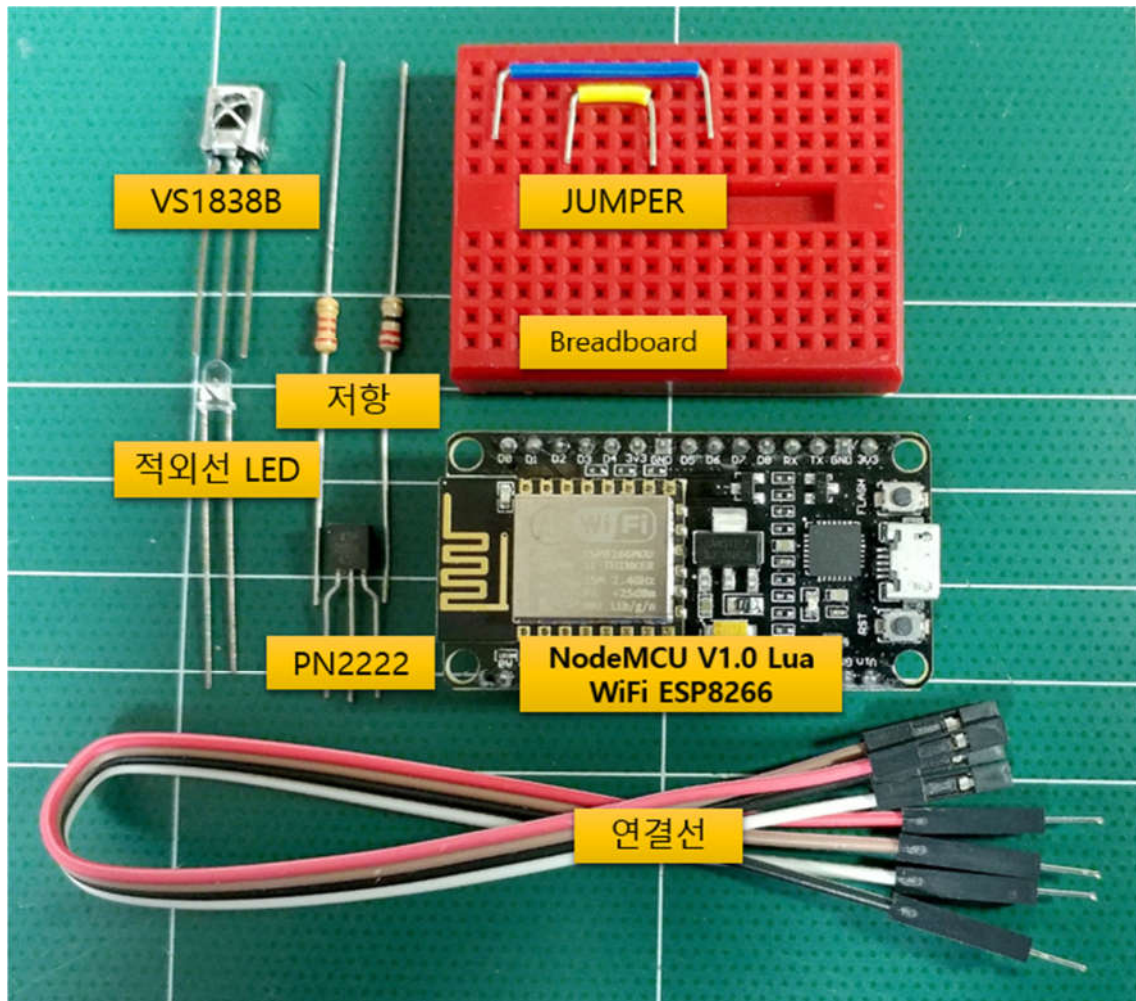
2 시스템 구성

인공지능 스피커(이하 스피커)는 많은 하드웨어와 소프트웨어가 필요하다.
다음은 스피커를 만들기 위한 재료 목록이다.

2.1 하드웨어

스피커 구성	IoT 디바이스
<ul style="list-style-type: none">■ 라즈베리파이 3■ 8G micro-SD■ USB Microphone■ Speaker (AUX 포트 필요)■ MAX7219■ 연결 와이어	<ul style="list-style-type: none">■ NodeMCU V1.0 Lua WiFi ESP8266■ 적외선 발광 LED■ VS1838B■ PN2222■ 저항 (22, 2.2K)■ JUMPER 2개■ 연결 와이어





2.2 소프트웨어

기본 소프트웨어	추가 소프트웨어
<ul style="list-style-type: none"> ■ RASPBIAN ■ MySQL ■ Media Stream Server - firefly ■ MQTT Broker - Apache Apollo 1.7.1 ■ Swig ■ Snowboy 	<ul style="list-style-type: none"> ■ MySQL Workbench ■ SmartTTY 3.1 ■ Arduino ■ Win32 Disk Imager ■ FileZilla Client

Memo

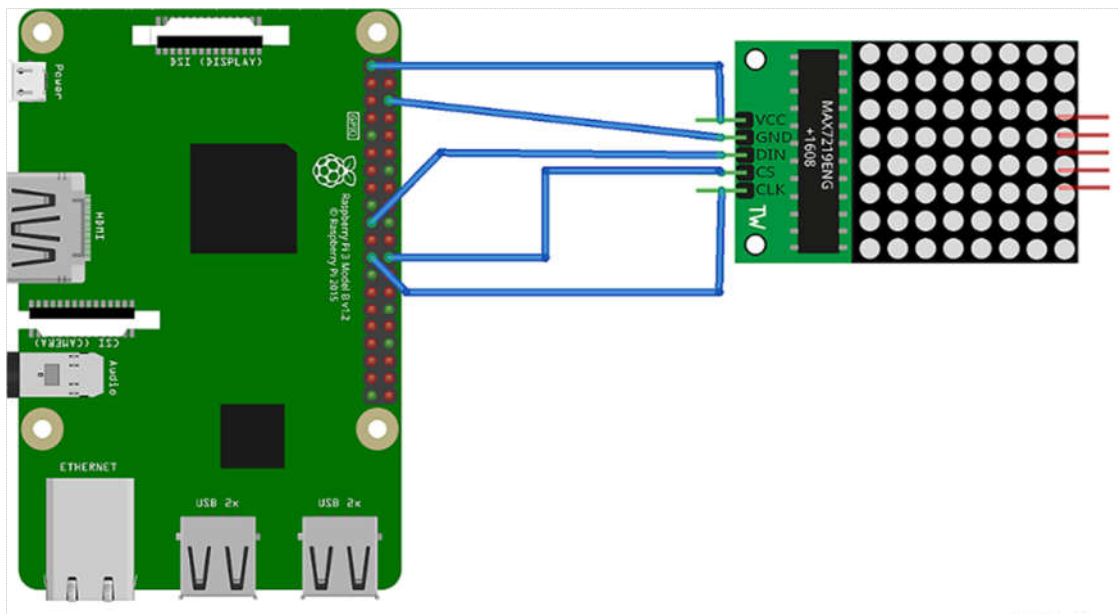
추가 소프트웨어 중에는 윈도우만 지원하는 것이 있으므로 초보자는 가급적 윈도우에서 작업하기를 권장한다.

3. AI 스피커 만들기

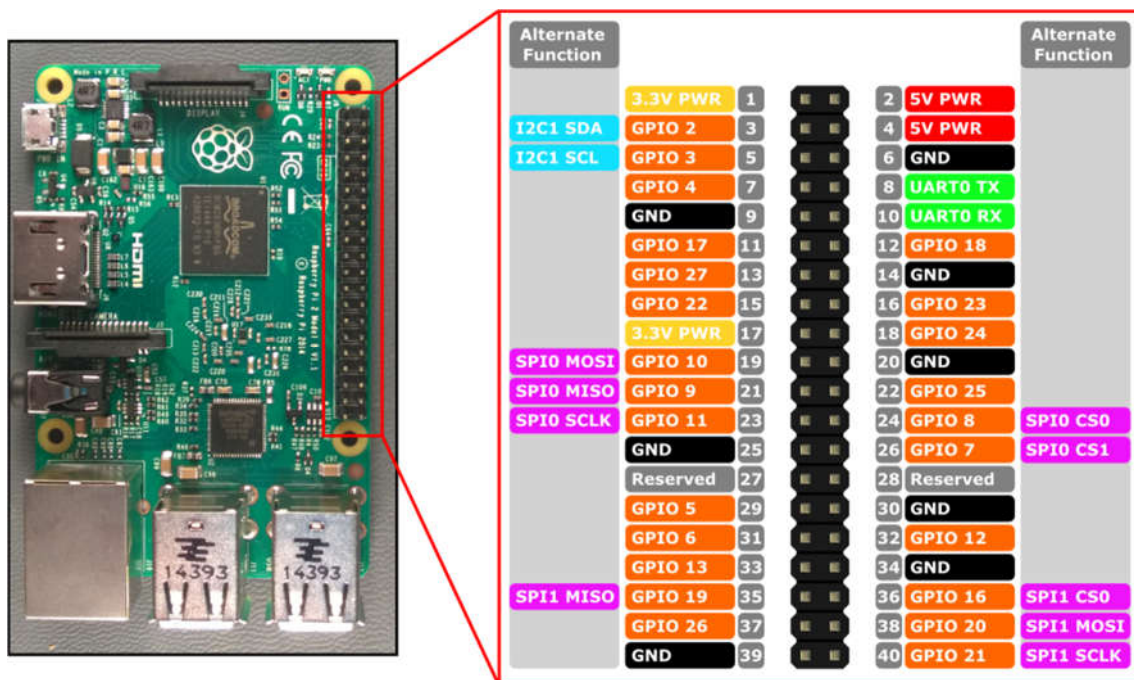
라즈베리파이 3를 이용한 인공지능 스피커는 많은 운영체제 외에도 많은 소프트웨어 설정이 필요하다.

본 문서에서는 이와 관련된 설정이 모두 끝난 이미지를 이용한다. 이 이미지에는 RASPBIAN, MySQL, Media Stream Server(firefly), MQTT Broker(Apache Apollo 1.7.1), Swig, Snowboy 등이 모두 설치되어 있다.

3.1 하드웨어 연결



라즈베리파이 핀번호	MAX7219
1 (3.3V)	VCC
6 (GND)	GND
19 (SPI0 MOSI)	DIN
24 (SPI0 CS0)	CS
23 (SPI0 SCLK)	CLK



3.2 이미지 다운로드

라즈비안을 이용한 소프트웨어 설치가 모두 완료된 이미지는 다음의 URL에서 다운로드 할 수 있다.

<https://goo.gl/Xbk1LP>

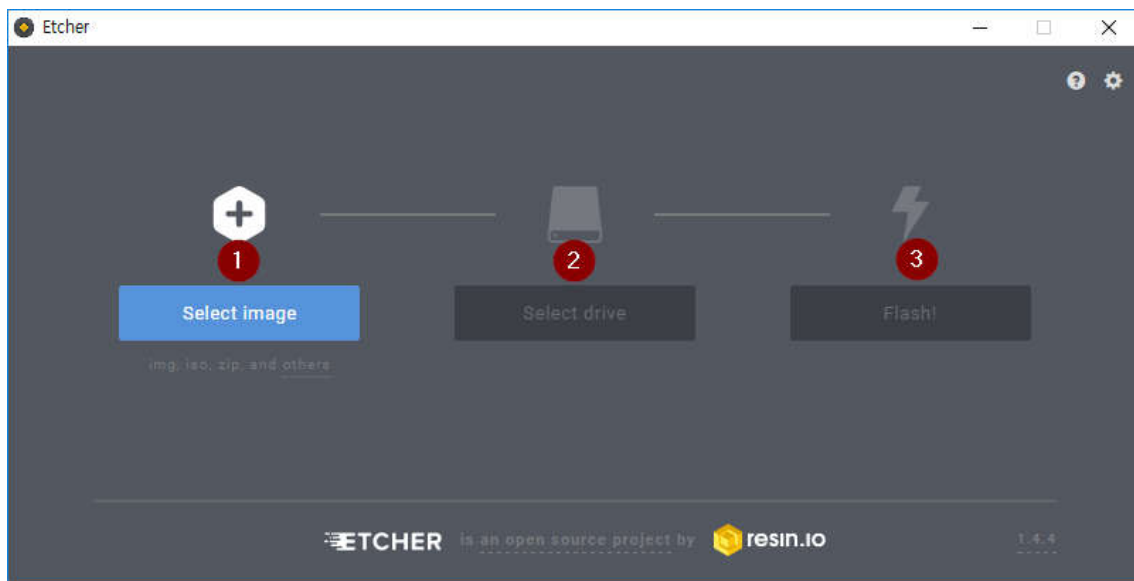
"AI_SPK_0623.zip" 파일을 다운로드하고 압축을 해제한다.

3.3 부팅 디스크 만들기

다운로드 한 이미지를 micro-SD 카드에 설치하기 위해서는 "Etcher"라는 소프트웨어가 필요하다. 다음의 URL에서 다운로드 하여 설치한다.

<https://etcher.io/>

설치가 완료되면 micro-SD 카드를 USB에 연결하고 "Etcher"를 실행한다.

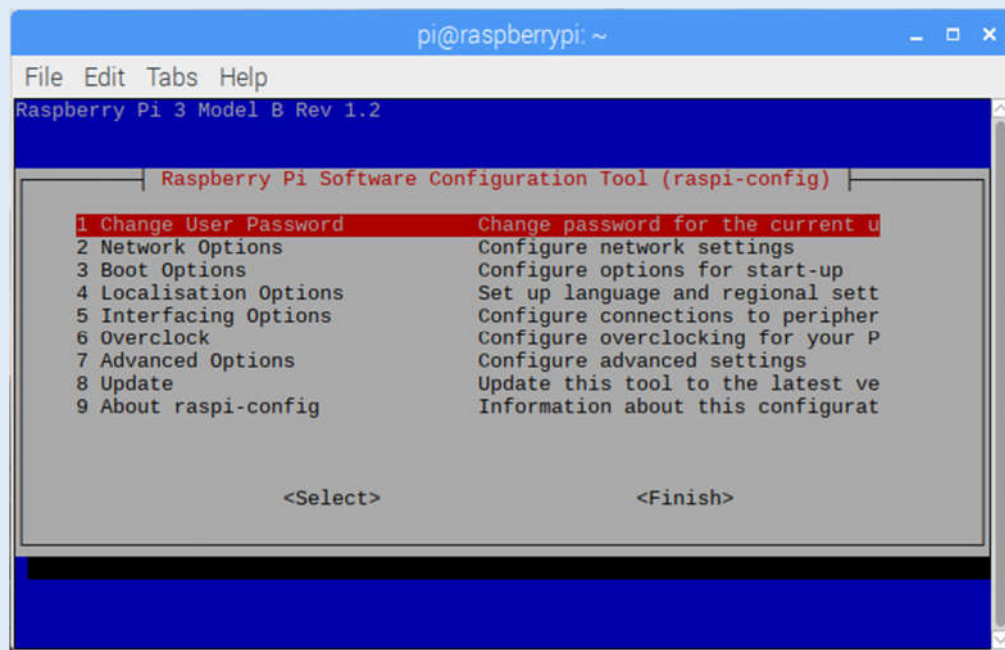


- ① "Select image" 버튼을 클릭하여 **"AI_SPK_0623.IMG"** 파일을 선택한다.
- ② "Select driver" 버튼을 클릭하여 micro-SD 카드를 연결한 드라이브를 선택한다.
- ③ "Flash!" 버튼을 클릭하여 쓰기를 시작한다.

TIP:

8G이상의 micro-SD를 사용하는 경우 모든 공간을 사용하기 위해서는 다음을 수행한다.

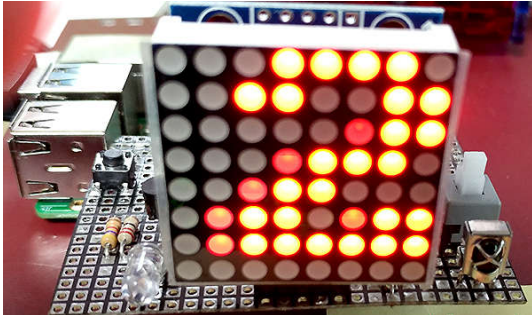
- ① 터미널에서 `sudo raspi-config` 실행한다.



- ② "7 Advanced Options > A1 Expand Filesystem"을 선택한다.

3.4 라즈베리파이 부팅

micro-SD 카드를 라즈베리파이에 연결하고, 유선 랜 포트, 전원 등을 연결하여 부팅을 시작한다. 정상적으로 부팅이 되면 8x8 Dot Matrix LED에 IP 주소가 스크롤 된다.



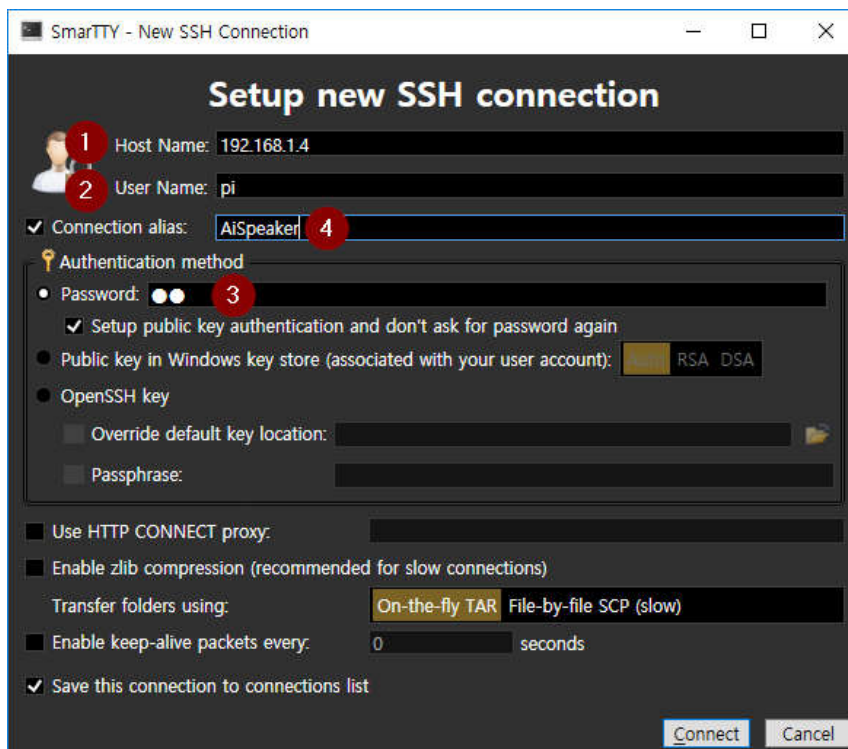
TIP:

키보드와 HDMI 디스플레이를 연결하여 ifconfig로 IP 주소를 확인한다.

SmartTTY를 이용하여 라즈베리파이에 접속하여 작업을 진행해야 한다. SmartTTY는 다음의 URL에서 다운로드 할 수 있다.

<http://smarttty.sysprogs.com/portable/>

SmartTTY를 실행하면 화면 좌측 하단에 "New SSH connection" 버튼을 클릭하여 연결 정보를 입력한다.



- ① 라즈베리파이의 IP 주소를 입력한다.
- ② 사용자 아이디 "**pi**"를 입력한다.
- ③ 비밀번호 "**pi**"를 입력한다.
- ④ 현재 연결정보를 대표하는 이름을 입력한다.

라즈베리파이 계정 정보	
ID	pi
PASSWORD	pi

3.5 음악 서비스 설정

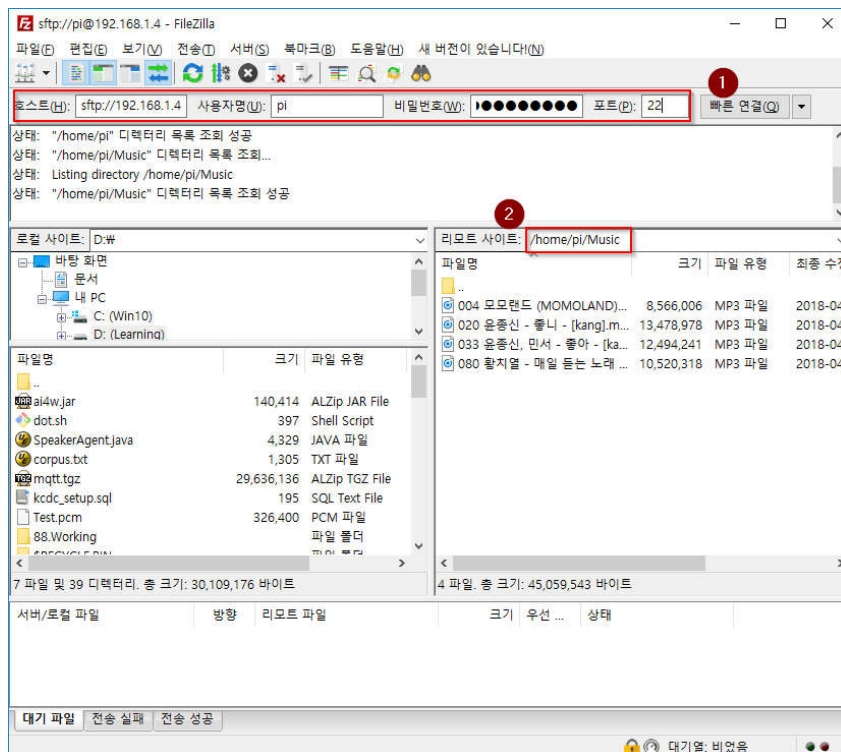
인공지능 스피커에서 사용하는 "firefly"라는 Media Stream Server를 사용한다.

대화로 음악을 검색하기 위해서는 원하는 음악파일을 라즈베리파이 또는 외장 하드 등에 설치하고, 정보를 MySQL에 저장해야 한다. 여기에서는 라즈베리파이에 음악파일을 복사하여 진행한다.

1) MP3 파일 복사

MP3파일은 FileZilla Client를 이용하여 라즈베리파이의 "/home/pi/Music" 폴더로 업로드한다. FileZilla Client는 다음의 URL에서 다운로드 할 수 있다.

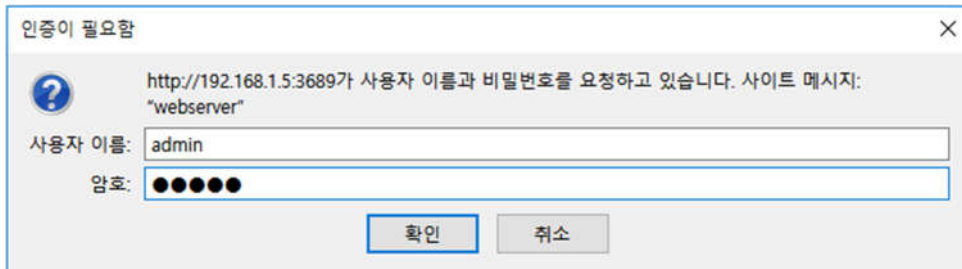
<https://filezilla-project.org/>



- ① 접속 정보를 입력하고 “빠른 연결” 버튼을 클릭한다.
- ② MP3 업로드 폴더

2) Media Server에 MP3 파일 등록

브라우저에서 `http://[라즈베리파일 주소]:3689`로 접속한다. 아이디는 **admin**이고, 비밀번호는 설정에 지정한 내용(**ibmer**)을 입력한다.



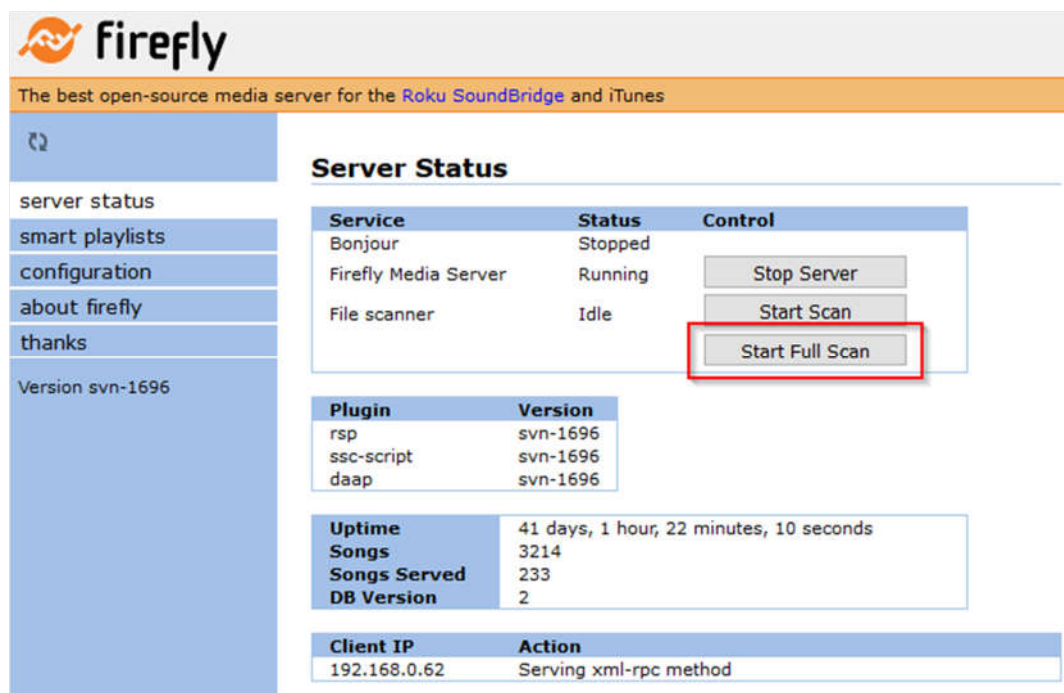
인증이 필요함

http://192.168.1.5:3689가 사용자 이름과 비밀번호를 요청하고 있습니다. 사이트 메시지: "webserver"

사용자 이름: admin

암호: ●●●●●●

확인 취소



firefly
The best open-source media server for the Roku SoundBridge and iTunes

server status
smart playlists
configuration
about firefly
thanks
Version svn-1696

Server Status

Service	Status	Control
Bonjour	Stopped	
Firefly Media Server	Running	Stop Server
File scanner	Idle	Start Scan Start Full Scan

Plugin	Version
rsp	svn-1696
ssc-script	svn-1696
daap	svn-1696

Uptime	41 days, 1 hour, 22 minutes, 10 seconds
Songs	3214
Songs Served	233
DB Version	2

Client IP	Action
192.168.0.62	Serving xml-rpc method

“Start Full Scan” 버튼을 클릭하면 “/home/pi/Music” 폴더의 MP3 파일을 검색한다.

3) MP3 정보 MySQL 데이터베이스에 입력하기

firefly에서 검색된 MP3 정보는 SQLite3 파일에 저장되며, 스피커에서 검색하기 위해서는 MySQL 서버의 “song” 테이블에 정보를 입력해야 한다.

라즈베리파이에 SmartTTY로 접속하여 다음의 명령을 수행한다.

```
cd ~/ai4w
./song.sh
```

TIP:

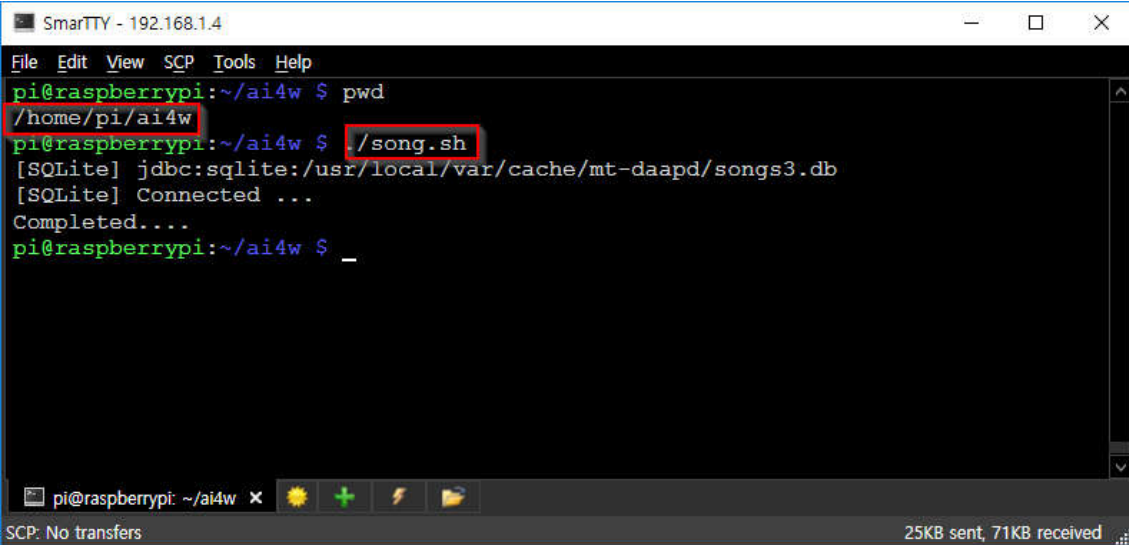
파일이 있는데 실행이 되지 않거나 파일이 없다고 나오는 경우

① 파일이 있는지 확인

ls -l <= 'l'은 소문자 L

② 실행 권한이 없는 경우

chmod +x 파일명



```
SmarTTY - 192.168.1.4
File Edit View SCP Tools Help
pi@raspberrypi:~/ai4w $ pwd
/home/pi/ai4w
pi@raspberrypi:~/ai4w $ ./song.sh
[SQLite] jdbc:sqlite:/usr/local/var/cache/mt-daapd/songs3.db
[SQLite] Connected ...
Completed....
pi@raspberrypi:~/ai4w $ _
```

pi@raspberrypi: ~/ai4w x

SCP: No transfers 25KB sent, 71KB received

4. AI 스피커 실행하기

4.1 준비작업

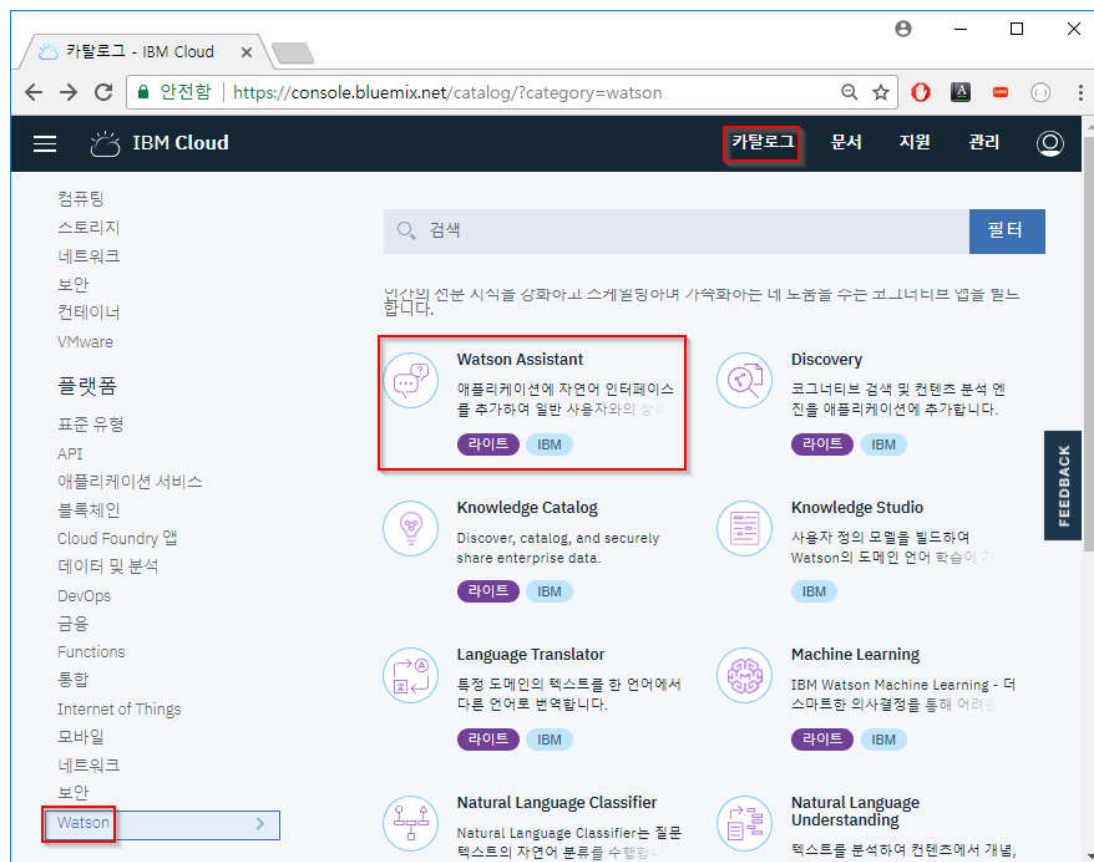
본 스피커는 다음과 같은 Cloud 서비스를 사용한다.

- Watson Assistant (Conversaion)
- Watson Speech To Text
- Watson Text to Speech
- 공공데이터 포털

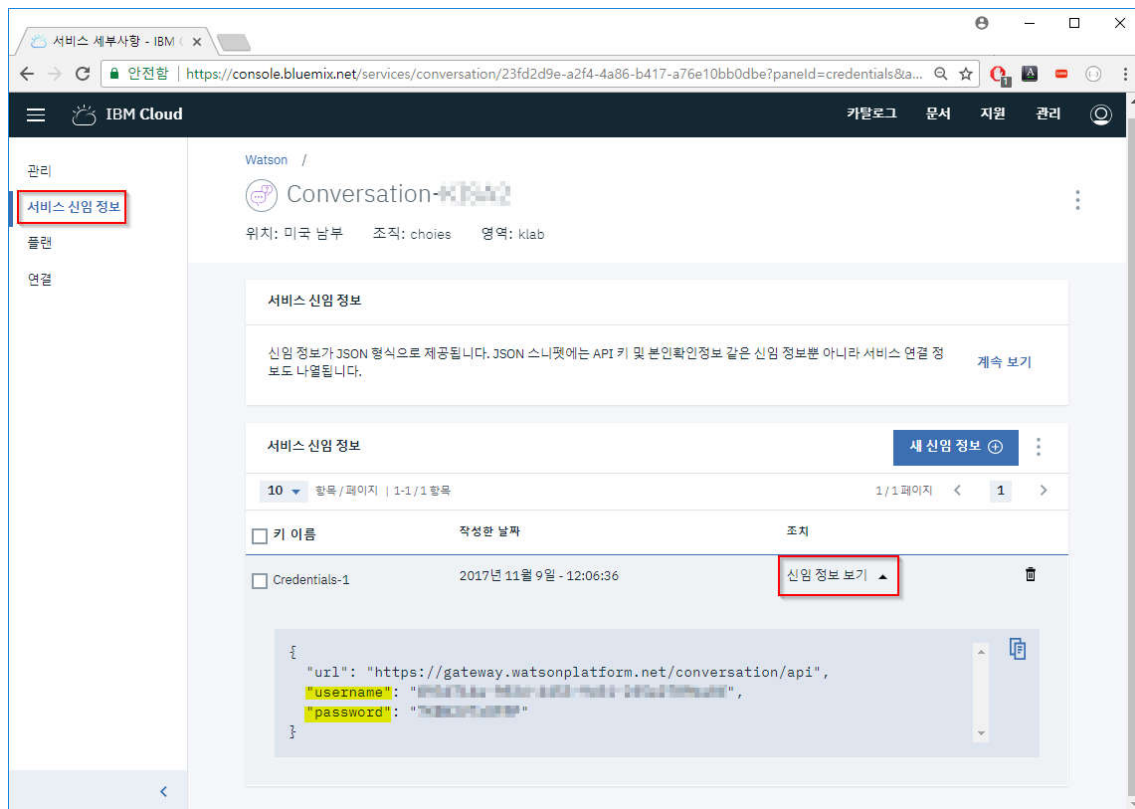
각 서비스 사용을 위해서는 credential 정보를 환경설정(apps.conf) 파일에 지정해야 한다.

1) Watson Assistant

IBM Cloud 에서 카탈로그 "Watson"을 선택하고 "Watson Assistant" 서비스를 선택한다.



정상적으로 서비스 생성이 완료되면 "대시보드"에 생성한 서비스가 나타나며, 클릭하면 다음과 같은 화면이 나타난다.

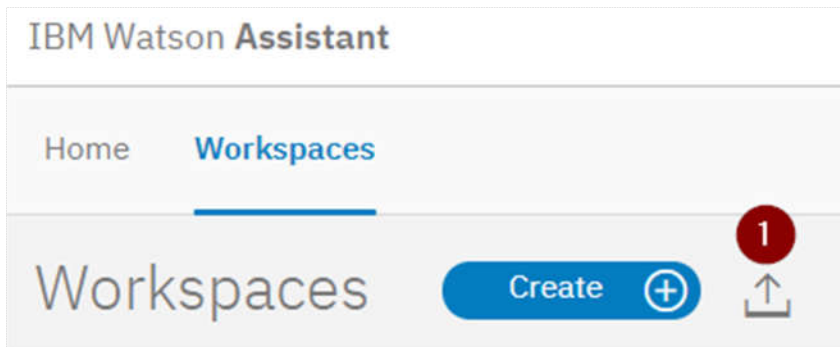


여기서 신임정보 "username", "password"는 환경설정에 사용이 되므로 메모장에 기록해 둔다.

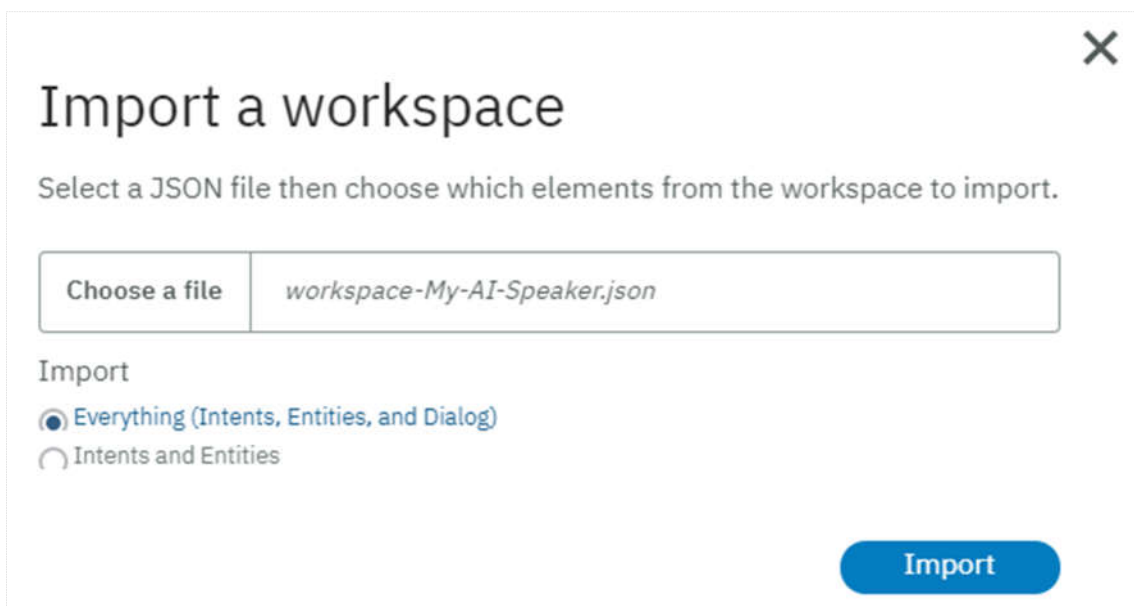
위 화면에서 "관리" 탭을 클릭한 후 "도구 실행"을 클릭한다.



그러면 Waton Assistant 화면이 나타난다.



① 버튼(Import workspace) 버튼을 클릭한다.

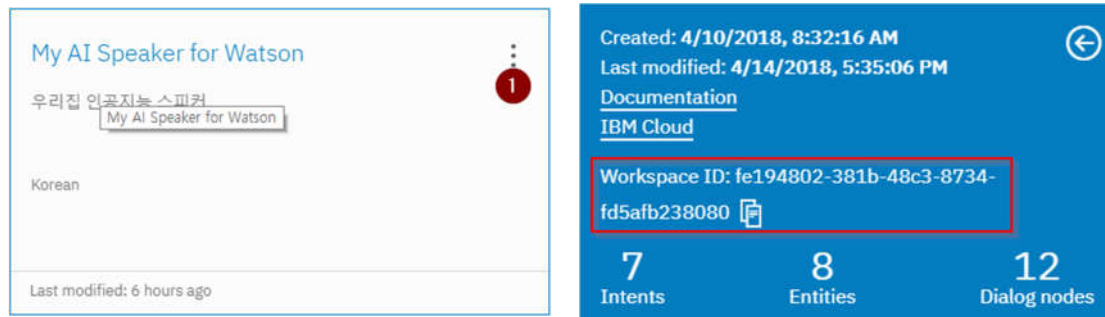


“workspace-My-AI-Speaker.json” 파일을 선택한 후 “Import” 버튼을 클릭한다.

“workspace-My-AI-Speaker.json” 파일은 다음의 URL에서 다운로드 할 수 있다.

<https://goo.gl/Xbk1LP>

정상적으로 수행이 완료되면 다음과 같은 화면이 나타나며, ①아이콘을 클릭하여 “View Details” 메뉴를 선택한다.

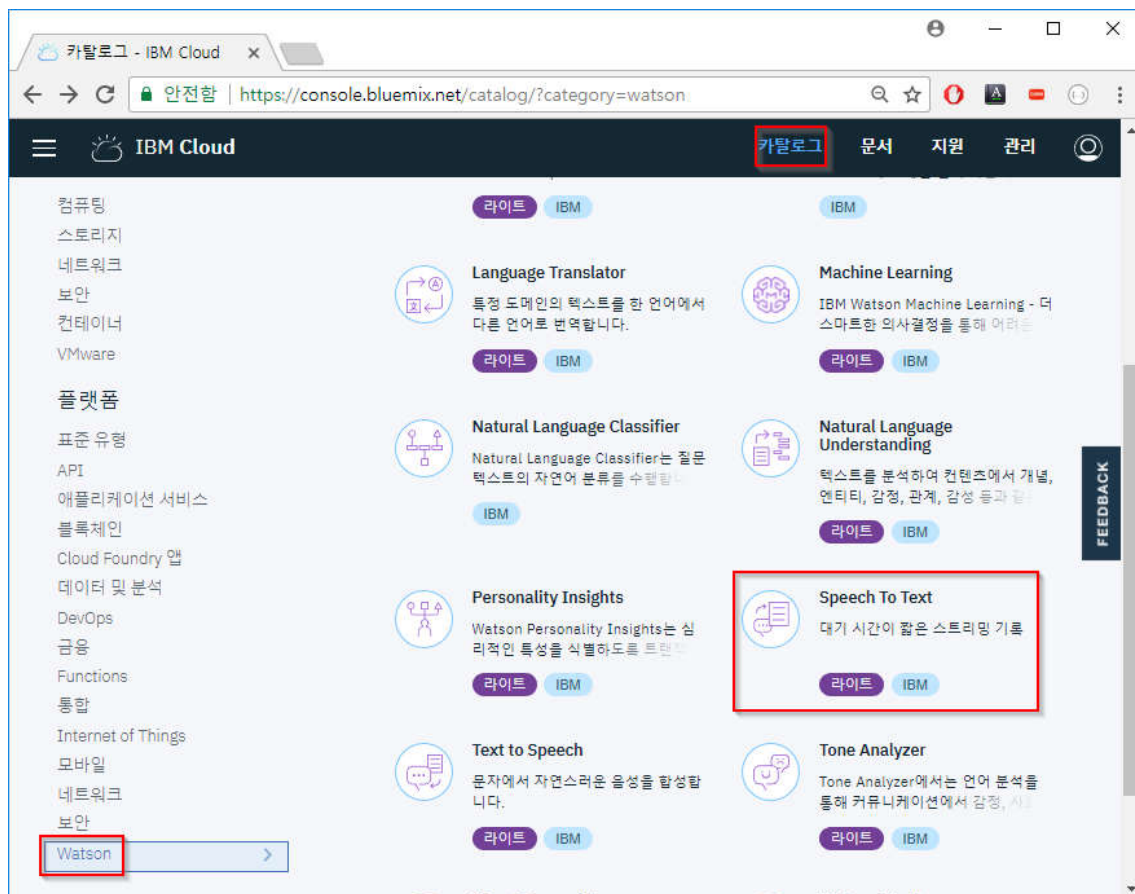


여기에서 Workspace ID를 메모장에 기록해 둔다.

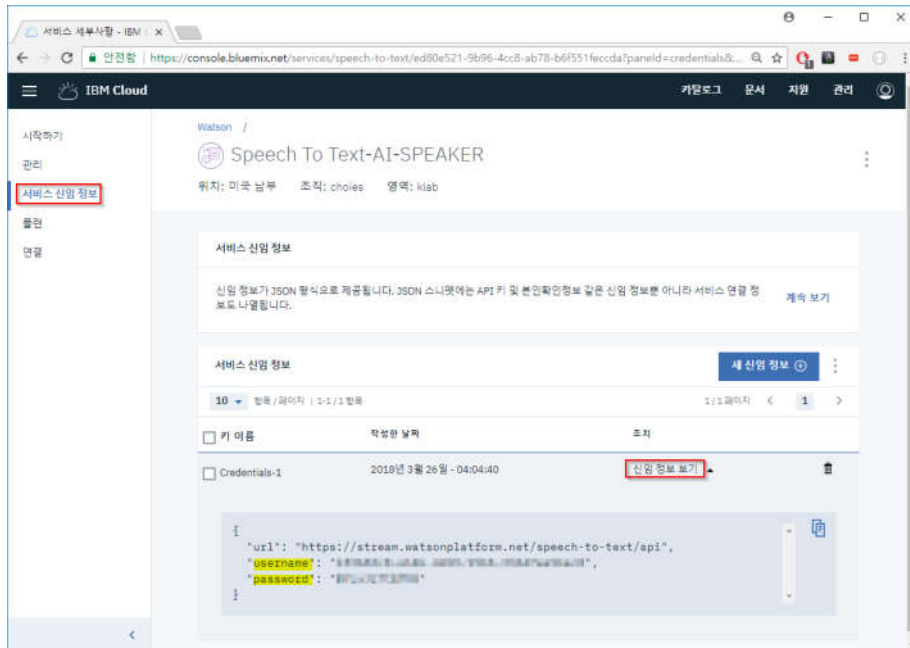
2) Watson Speech To Text

IBM Cloud 에서 카탈로그 "Watson"을 선택하고 "Speech To Text" 서비스를 선택한다.

★★ 언어모델의 학습이 필요한 경우 "Standard"로 생성한다. ★★



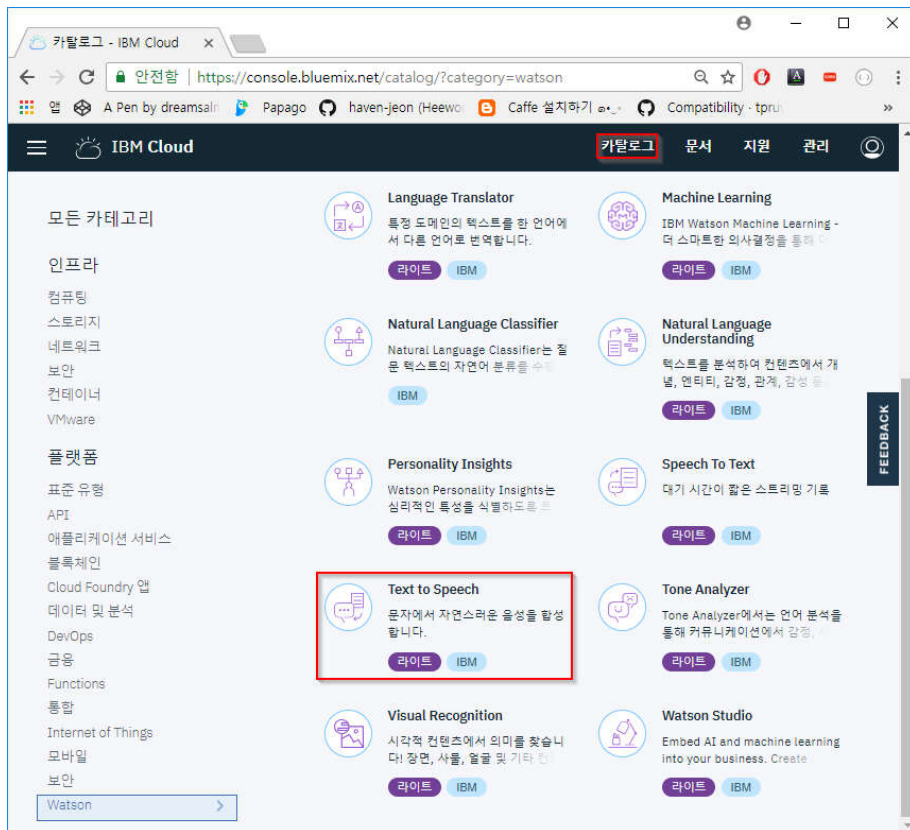
정상적으로 서비스 생성이 완료되면 "대시보드"에 생성한 서비스가 나타나며, 클릭하면 다음과 같은 화면이 나타난다.



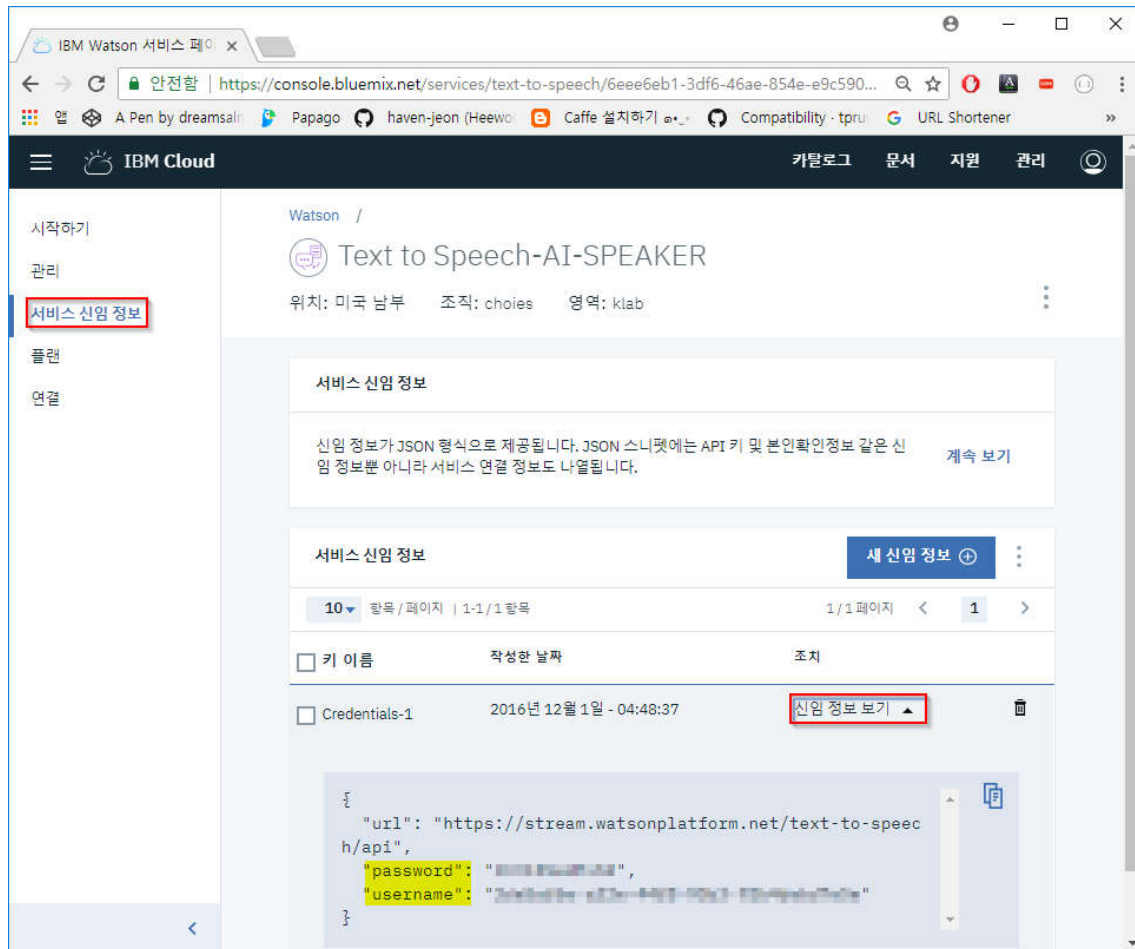
여기서 신임정보 "username", "password"는 환경설정에 사용되므로 메모장에 기록해 둔다.

3) Watson Text to Speech

IBM Cloud 에서 카탈로그 "Watson"을 선택하고 "Text to Speech" 서비스를 선택한다.



정상적으로 서비스 생성이 완료되면 "대시보드"에 생성한 서비스가 나타나며, 클릭하면 다음과 같은 화면이 나타난다.



여기서 신임정보 "username", "password"는 환경설정에 사용되므로 메모장에 기록해 둔다.

4) 공공데이터 사용하기

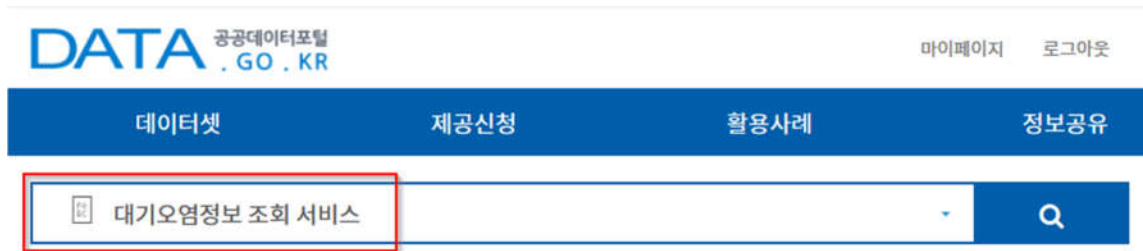
본 스피커에서는 공공데이터포털에서 대기오염 정보와 서울버스 도착 정보를 사용한다. 데이터를 사용하기 위해서는 서비스 신청을 해야하며 승인이 완료된 후에 사용할 수 있다. 아래의 URL에서 회원가입하고 다음의 서비스를 신청을 한다.

- 대기오염정보 조회 서비스
- 버스도착정보조회 서비스 (서울시)

<https://www.data.go.kr>

서비스 신청방법

- ① 홈페이지에서 서비스를 검색한다.



- ② 검색 결과에서 사용 서비스를 클릭한다.



- ③ 상세정보 페이지에서 "활용신청"을 클릭한다.



- ④ "개발계정 신청" 화면에서 정보를 입력하고 "신청" 버튼을 클릭한다.

승인이 완료되면 "마이페이지"에 나타나며, 사용 서비스를 클릭한다.



“개발계정 상세보기” 페이지의 “서비스정보”에 “일반 인증키”가 나타나며, 환경 설정을 위해서 메모한다.

5) 날씨예보 지역 찾기

날씨 정보는 기상청 RSS를 이용한다. 정보를 가져올 지역은 다음의 URL에서 얻는다.

http://www.weather.go.kr/weather/lifenindustry/sevice_rss.jsp

- ① 콤보박스에서 값을 선택한 후 “검색”을 클릭한다.
- ② RSS 버튼을 클릭하면 zone값이 나타나며, Conversation에서 사용하므로 값을 메모한다.

6) 버스 도착정보

버스 도착 정보를 얻기 위해서는 노선 ID와 정류장 번호가 필요하다. 이 정보를 얻는 방법은 다음과 같다. (크롬 브라우저 사용)

- ① <http://topis.seoul.go.kr> 사이트에 접속하여 원하는 노선 번호를 검색한다.

- ② 검색된 노선의 "정류소 출력"을 클릭한다.

- ③ 정류소 정보가 출력되며, 노선ID(routeId)와 정류장번호를 메모한다.

서울시 교통정보 시스템 - TOPIS - Chrome

① topis.seoul.go.kr/map/openPopBusRoute.do?routeId=232000047

8600김포 번 노선ID

운행정보 프린트하기

기점	양곡터미널		종점	시청,서소문청사		버스타입	경기	
버스회사			전화			E-mail		
배차간격	평일	-분	첫차	평일	04:20	막차	평일	23:00
	토요일	-분		토요일	04:20		토요일	23:00
	공휴일	-분		공휴일	04:20		공휴일	23:00

공휴일
정류장번호

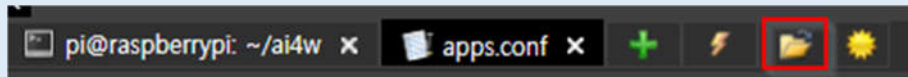
순서	경유정류소	순서	경유정류소	순서	경유정류소
1	양곡터미널(35719)	19	고촌(41004)	37	당산역(19996)
2	김포한강7로(경유)(0)	20	88JC(경유)(0)	38	고촌(41008)

4.2 환경설정

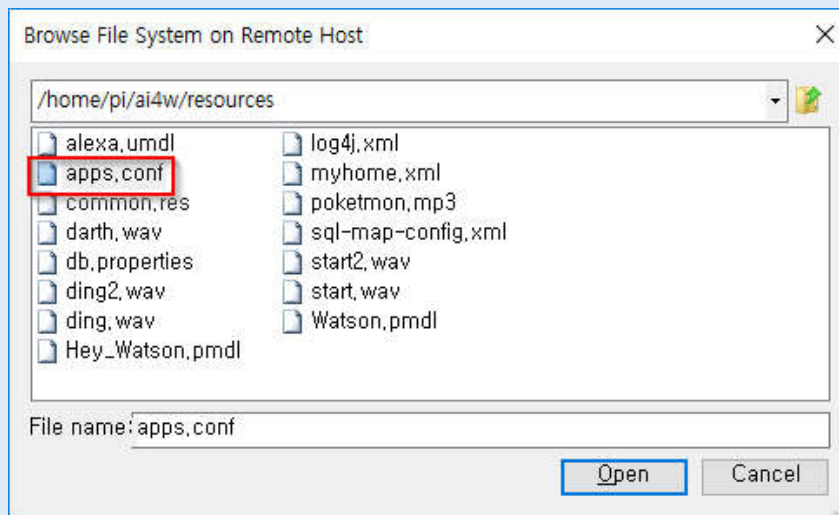
Cloud 서비스 사용을 위한 정보는 /home/pi/ai4w/resources/apps.conf 파일에 기록한다. apps.conf 파일의 항목들은 위에서 등록한 정보로 입력한다. 편집은 vi 명령어 또는 SmartTTY 프로그램에서 제공하는 파일 편집 기능을 이용한다.

TIP:

SmartTTY 화면의 좌측 하단에 있는 오픈 아이콘을 클릭한다.



그리고 "apps.conf" 파일을 선택한 후 "Open" 버튼을 클릭하여 파일을 수정한다.



항목	설명
wcs.user	Watson Assistant 서비스의 username
wcs.passwd	Watson Assistant 서비스의 password
wcs.workid	Watson Assistant 서비스의 Workspace ID
stt.watson.user	Watson STT 서비스의 username
stt.watson.passwd	Watson STT 서비스의 password
tts.watson.user	Watson TTS 서비스의 username
tts.watson.passwd	Watson TTS 서비스의 password
stt.custid	Watson STT 서비스의 Customization Id [아래 STT Training 설명 참조]
air.key	대기오염정보 조회 서비스 일반 인증키

STT Training :

현재 Watson STT는 학습 중이다. 그러므로 인식률을 높이기 위해 현재 사용하는 Conversation에 있는 intent examples를 학습 시킬 것이다.

학습을 위해서는 "apps.conf" 파일에 Conversation, STT 등의 정보가 입력되어 있어야 하며, 라즈베리파이 콘솔 상에서 다음의 명령을 수행한다.

```
cd ~/ai4w
./kr_train.sh
```

수행이 완료되면 다음과 같은 로그가 나타난다.

```
INFO: --> POST https://stream.watsonplatform.net/speech-to-text/api/v1/
allow_overwrite=true http/1.1 (-1-byte body)
Apr 17, 2018 10:00:02 PM okhttp3.internal.platform.Platform log
INFO: <-- 201 Created https://stream.watsonplatform.net/speech-to-text/
omeai?allow overwrite=true (931ms, unknown-length body)
* Customization Id = 01499bb0-164c-447f-a15f-46835cac6cb6
Apr 17, 2018 10:00:13 PM okhttp3.internal.platform.Platform log
INFO: --> POST https://stream.watsonplatform.net/speech-to-text/api/v1/
0-byte body)
Apr 17, 2018 10:00:13 PM okhttp3.internal.platform.Platform log
INFO: <-- 200 OK https://stream.watsonplatform.net/speech-to-text/api/v
unknown-length body)
pi@raspberrypi:~/ai4w $ _
```

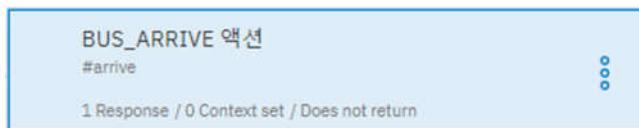
위 그림에서 Customization ID를 apps.conf 파일의 **stt.custid** 항목에 지정한다.

4.3 Watson Conversation 설정

미세먼지 정보, 버스도착 정보를 대화를 통해 얻기 위해서는 Waton Assistant의 Dialog에 앞에서 메모한 정보를 입력해야 한다.

1) 버스도착 정보

- ① "BUS_ARRIVE 액션" 노드를 선택한다.



- ② 우측에는 상세정보를 입력하는 화면이 나타나며, "The Respond with" 부분에 있는 메뉴를 클릭하여 "Open JSON editor"를 선택한다.



- ③ JSON editor에서 action 부분의 "params" 부분을 앞에서 메모한 정보로 수정한다.

```

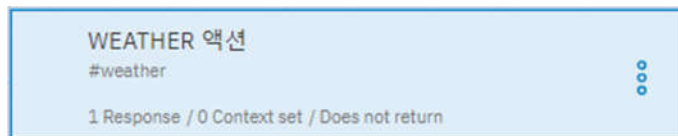
1 {
2   "output": {
3     "text": {
4       "values": [
5         "%BUS_ARRIVE%"
6       ]
7     },
8     "action": {
9       "id": "BUS_ARRIVE",
10      "type": "REPL",
11      "params": {
12        "arsId": "17234",
13        "routeId": "232000067"
14      }
15    }
16  }
17 }

```

- arsId : 정류장 번호
- routeId : 노선 ID

2) 날씨정보

- ① "WEATHER 액션" 노드를 선택한다.



- ② 우측에는 상세정보를 입력하는 화면이 나타나며, "The Respond with" 부분에 있는 메뉴를 클릭하여 "Open JSON editor"를 선택한다.

Then respond with:



- ③ JSON editor에서 action 부분의 "params" 부분을 앞에서 메모한 정보로 수정한다.

```

1 {
2   "output": {
3     "text": {
4       "values": [
5         "%WEATHER%"
6       ]
7     },
8     "action": {
9       "id": "WEATHER",
10      "type": "REPL",
11      "params": {
12        "city": "서울",
13        "zone": "4129052000"
14      }
15    }
16  }
17 }

```

- city : 미세먼지 측정 지역
- zone : 날씨예보 지역

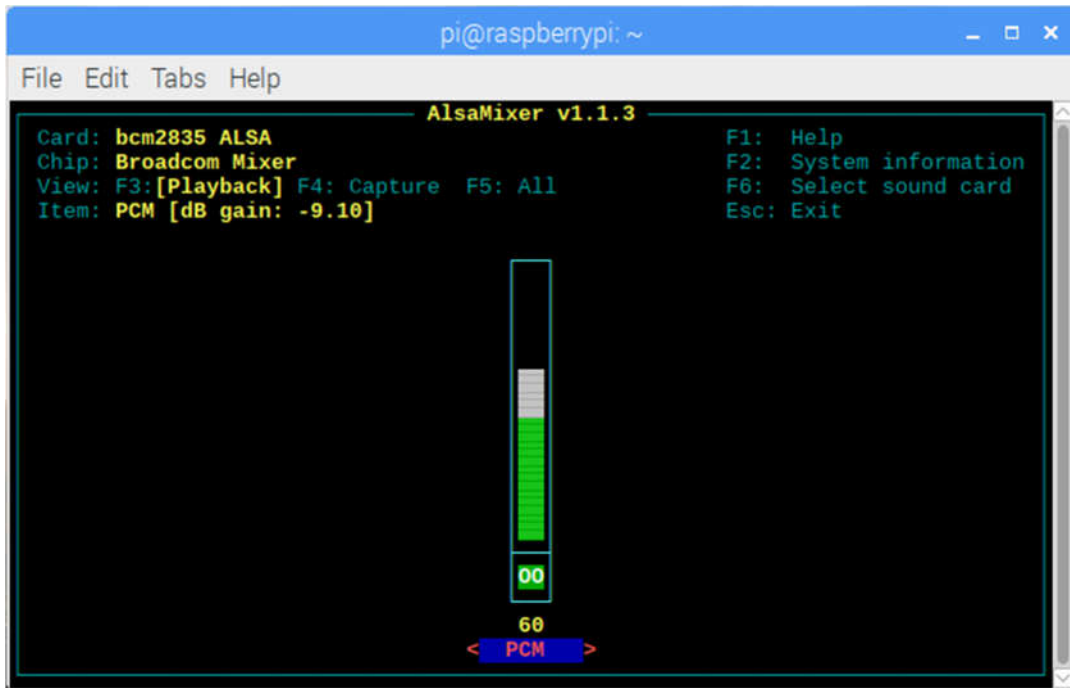
울산,영동,충북,경기남부,영서,서울,전북,충남,부산,경북,대전,세종,제주,경기북부,대구,전남,인천,광주,경남

4.4 실행

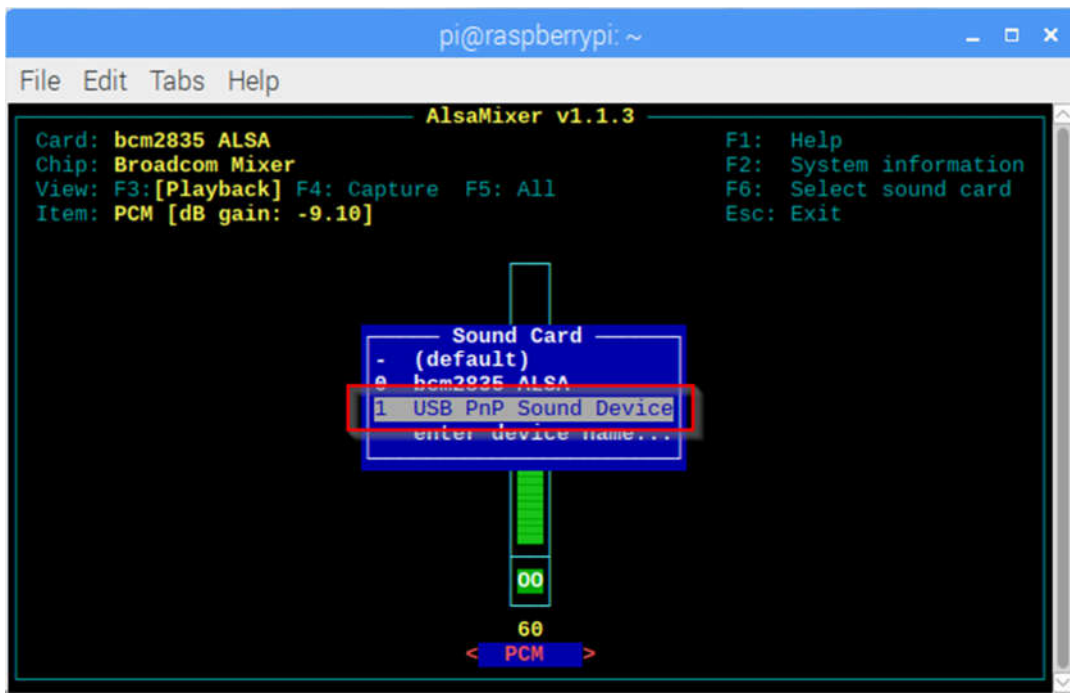
실행 전에 스피커, 마이크의 상태를 점검하기 위해 다음을 수행한다.

```
alsamixer
```

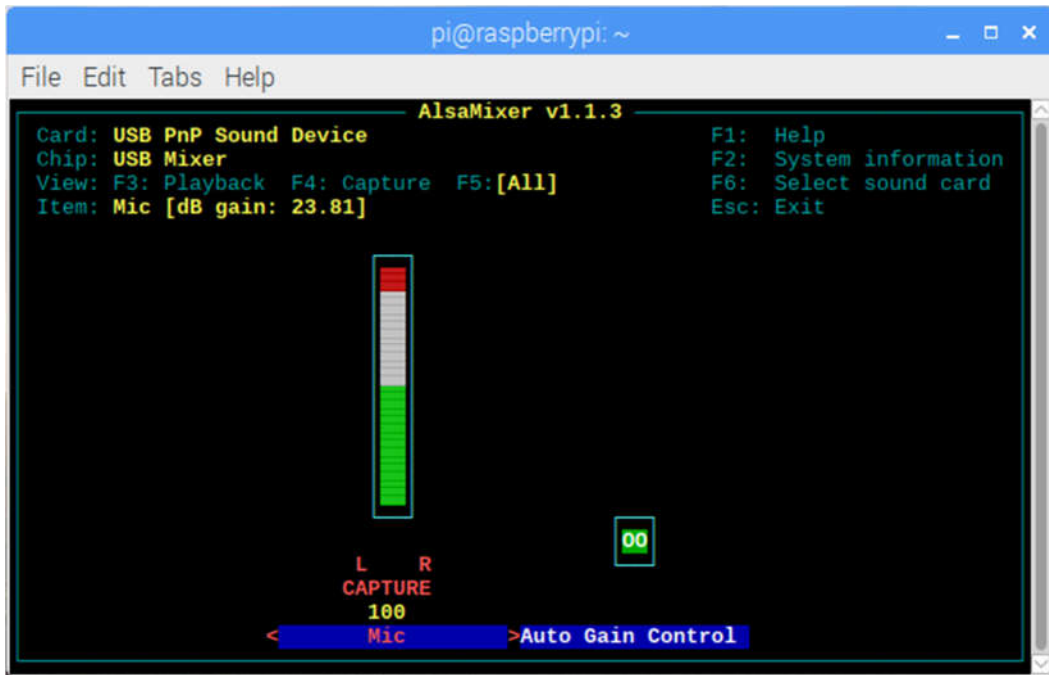
① 볼륨을 UP/DOWN 키로 설정한다.



② F6을 눌러 마이크를 선택한다.



③ F5을 눌러 모든 항목을 표시하고 방향키를 이용하여 감도를 설정한다.



변경된 내용은 다음의 명령을 통해 저장한다.

```
sudo alsactl store 0
```

이제 모든 준비가 완료 되었으므로 Java 어플리케이션을 실행한다.

다음의 위치에서 ai4w.sh를 실행한다.

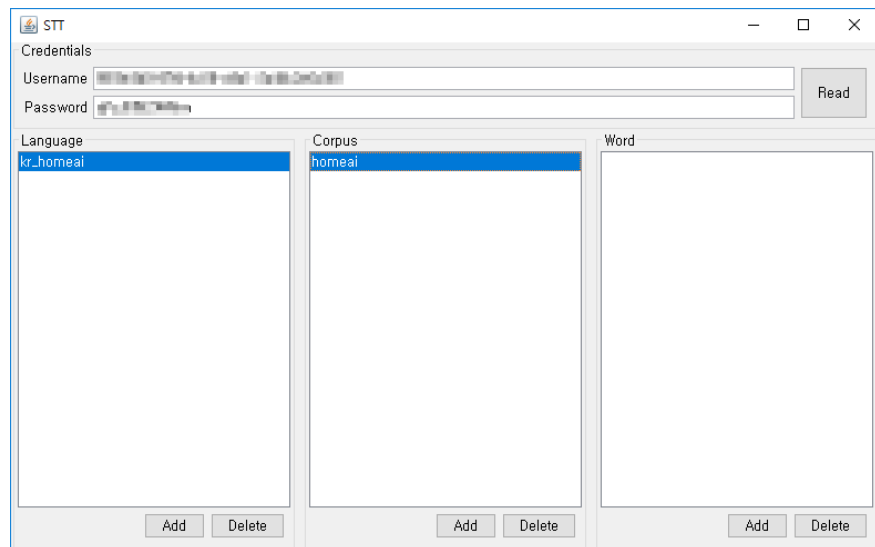
```
cd ~/ai4w
./ai4w.sh
```

```
pi@raspberrypi:~/ai4w $ ./ai4w.sh
[MIC] Device [plughw:1,0]
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
Apr 15, 2018 6:44:06 PM okhttp3.internal.platform.Platform log
INFO: --> POST https://gateway.watsonplatform.net/conversation/api/v1/workspaces/f
02-16 http/1.1 (51-byte body)
Apr 15, 2018 6:44:07 PM okhttp3.internal.platform.Platform log
INFO: <-- 200 OK https://gateway.watsonplatform.net/conversation/api/v1/workspaces
8-02-16 (842ms, 333-byte body)
==[INPUT]
==[OUTPUT]
안녕하세요? 무엇을 도와드릴까요?
Stream player Status -> NOT_SPECIFIED
@.@ STARTED !! 3200
Connected.. {}
```

- Hotword : **헤이 왓슨**
- 테스트 문장
 - 노래 틀어
 - 티브이 켜
 - 티브이 꺼
 - 티비엔 틀어
 - KBS 틀어
 - 버스 언제와

그리고 홈 폴더에는 다음과 같은 기능을 수행하는 프로그램이 있다.

셸 파일	기능
ai4w.sh	인공지능 스피커 메인 어플리케이션
song.sh	MP3 정보를 MySQL에 저장
dot.sh	8x8 Dot Matrix 코드 생성 도구
ip.sh	라즈베리파이 IP 주소 표시
kr_train.sh	Watson STT 트레이닝
stt.sh	STT 학습 정보를 삭제



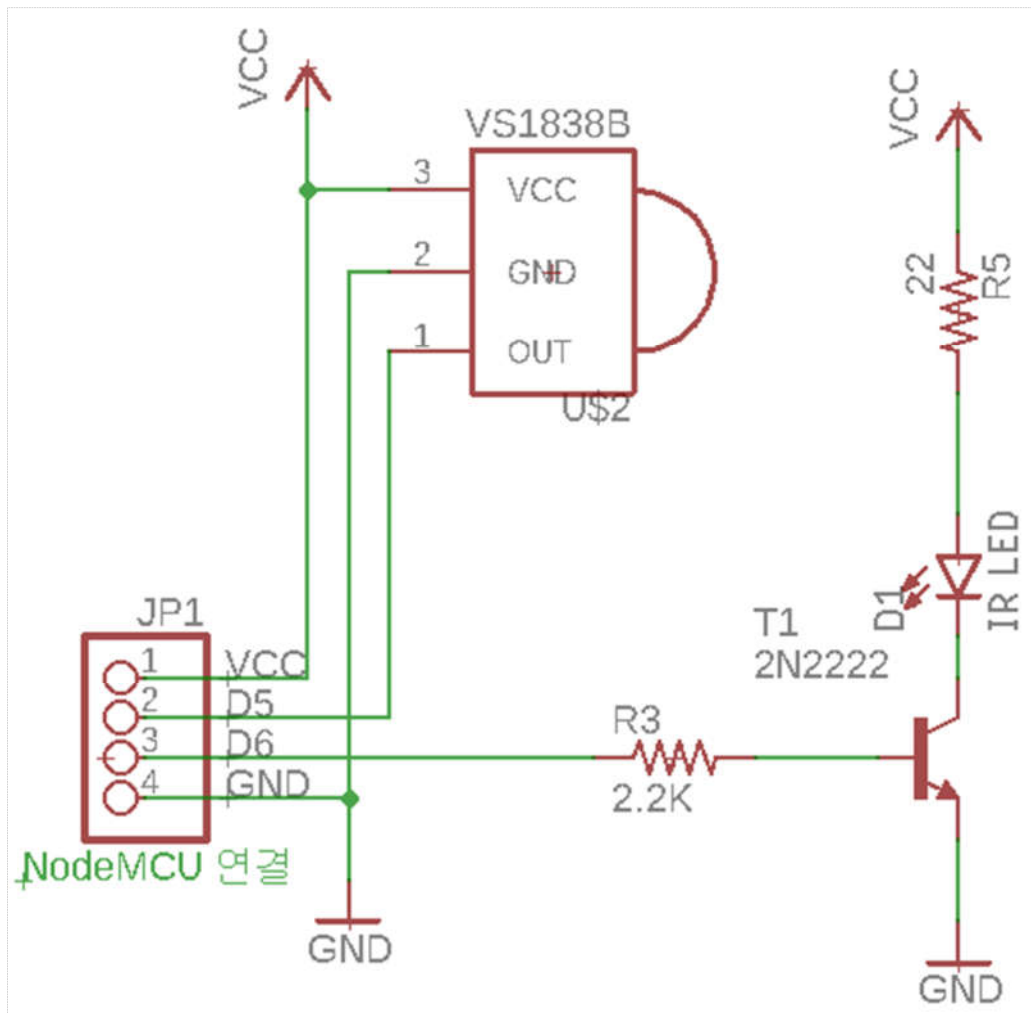
5. IoT 디바이스 만들기

본 스피커에서는 음성을 이용하여 텔레비전, 에어컨 등을 제어할 것이다. 그러기 위해서는 리모컨을 제어해야 한다. 하지만 아쉽게도 서비스에 딱 맞는 제품이 없어 별도로 제작해야 한다. 그나마 다행(?)인 것은 그리 복잡하지 않게 만들 수 있다는 것이다.

여기에서 만들 IR 디바이스는 두 가지의 기능을 갖는다.

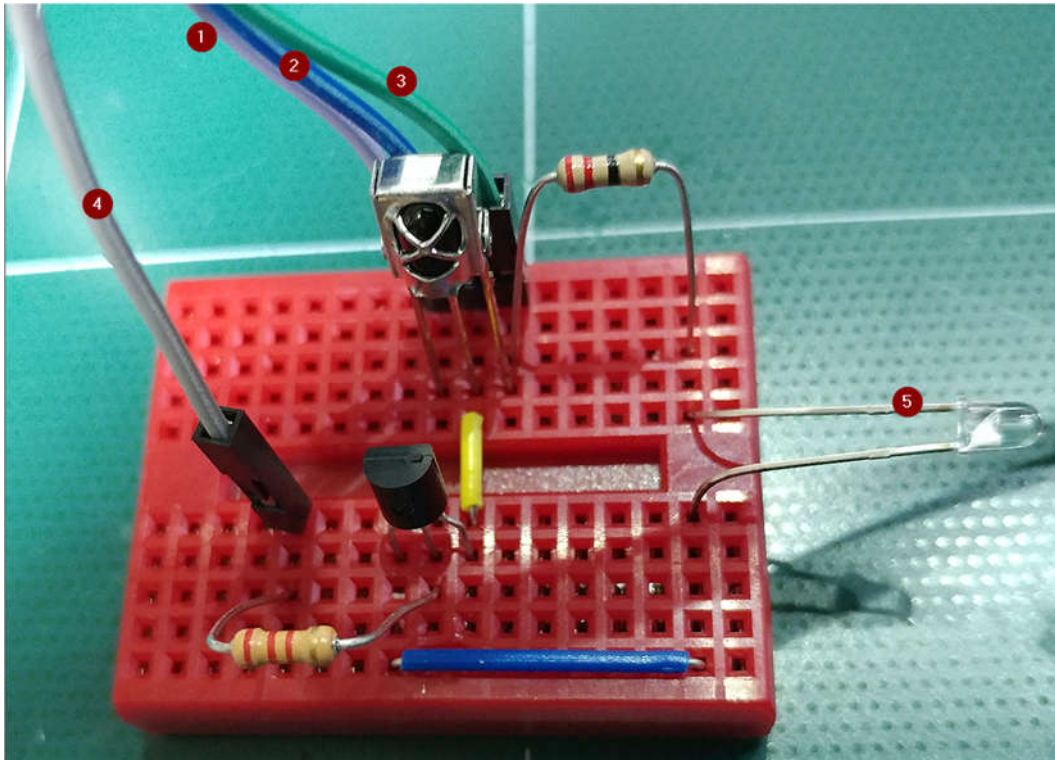
- 리모컨의 IR 코드값 발신
- 리모컨의 IR 코드값 기록(Capture)

Capture 기능은 가정에서 사용 중인 리모컨의 코드값을 저장하고, 이 코드값을 음성 서비스에서 사용할 것이다.

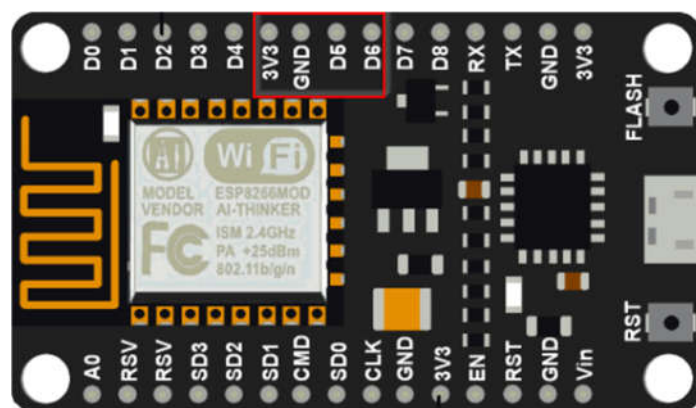


[회로도]

5.1 하드웨어 연결



번호	연결
1 (SIGNAL)	NodeMCU 보드의 D5 핀에 연결한다.
2 (GND)	NodeMCU 보드의 GND 핀에 연결한다.
3 (VCC)	NodeMCU 보드의 3v3 핀에 연결한다.
4	NodeMCU 보드의 D6 핀에 연결한다.
5	LED의 "+" 핀을 연결한다. 다리가 긴 쪽이 "+" 이다.



Meetup

Meetup에서는 조립이 완료된 디바이스를 제공한다.

5.2 개발환경

여기서 만들 디바이스는 MQTT를 사용하며, 기능의 구현은 Arduino(이하 아두이노)를 이용한다. 아두이노는 다음의 URL에서 다운로드 할 수 있다.

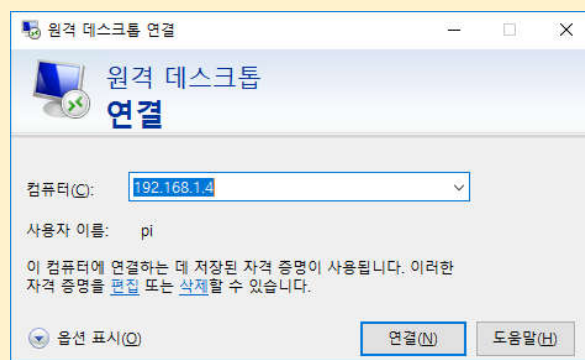
<https://www.arduino.cc/en/Main/Software#>



압축파일을 해제하고 arduino.exe를 실행한다.

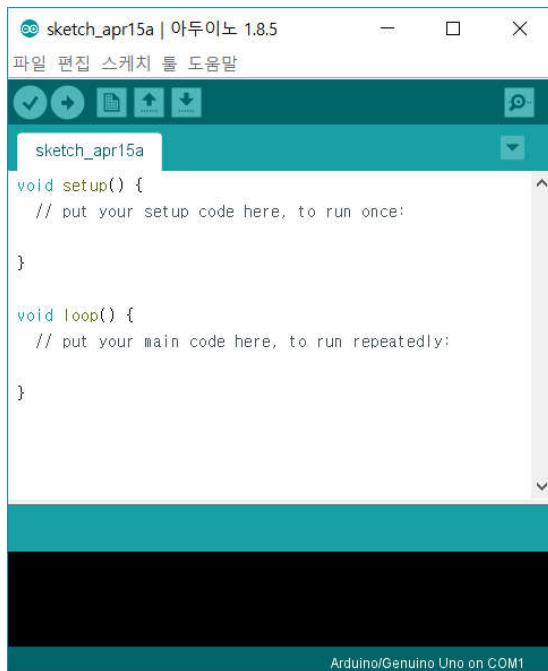
Meetup

라즈베리파이는 원격접속이 가능하도록 설정되어 있다. 그러므로 “원격 데스크톱 연결”을 수행할 수 있다.



연결이 완료되면 바탕화면의 “Arduino IDE”를 실행한다.

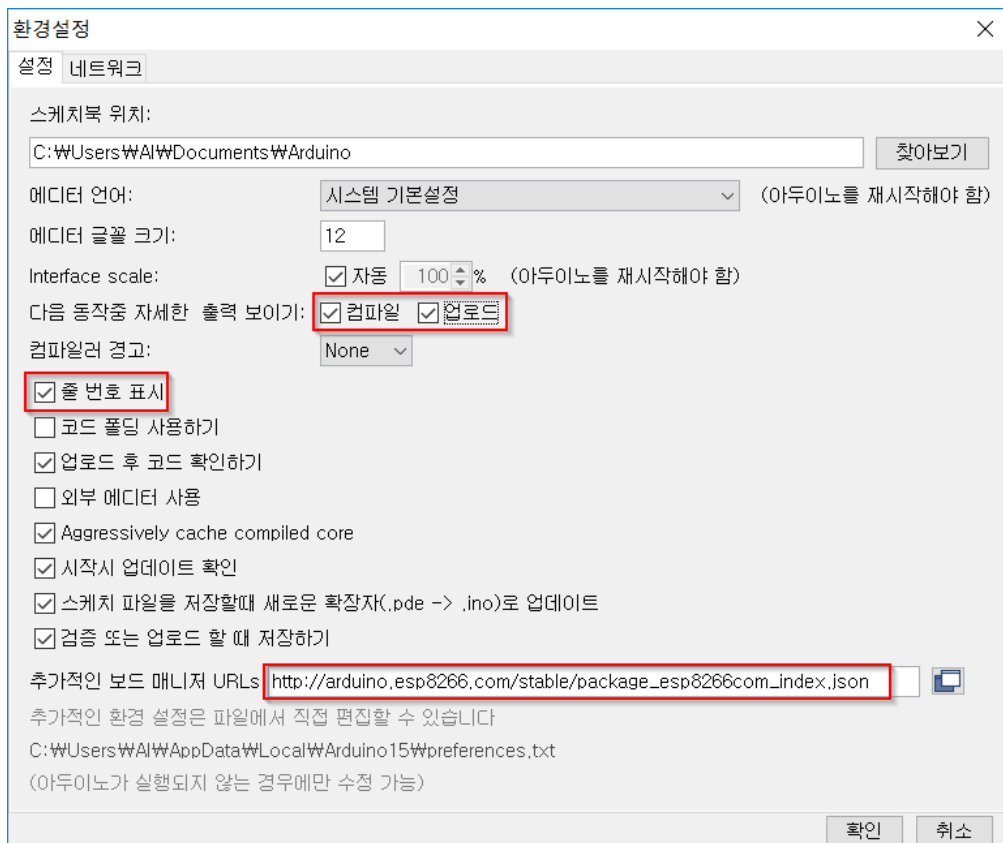




ESP8266을 이용하여 개발하기 위해서는 보드 및 사용 라이브러리를 설치해야 한다.

1) ESP8266 보드 설치

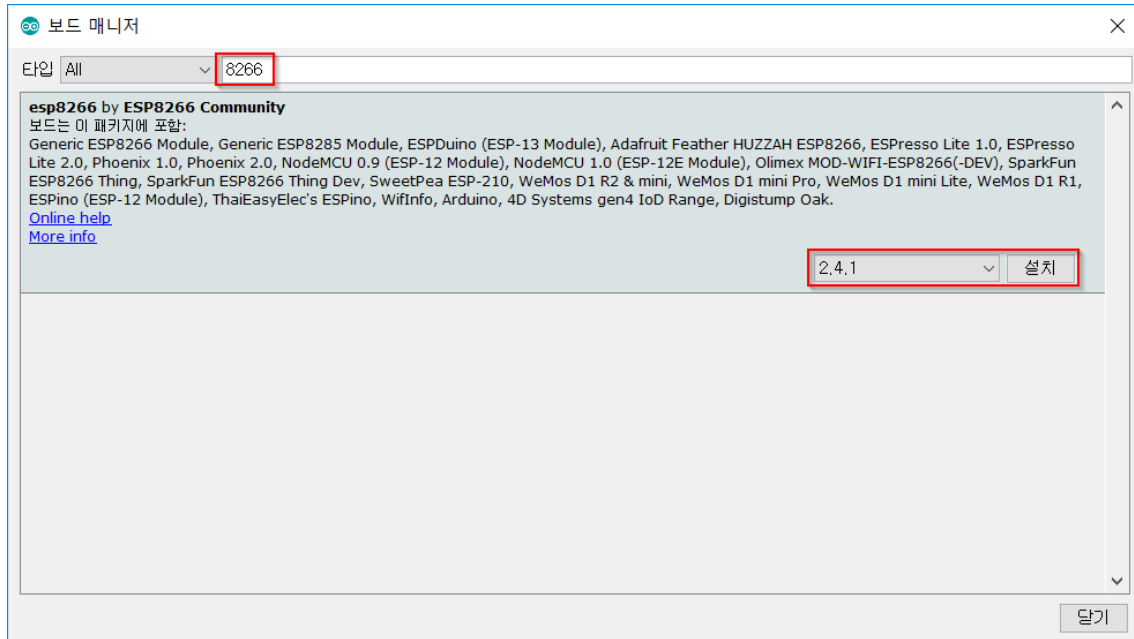
메뉴에서 "파일 > 환경설정"을 선택한다.



“추가적인 보드 매니저 URLs”에 다음 URL을 입력한다.

http://arduino.esp8266.com/stable/package_esp8266com_index.json

메뉴 “툴 > 보드 > 보드 매니저”를 선택한다. 화면이 나타나면 필터에 8266을 입력하고 버전 2.4.1을 선택하고 “설치” 버튼을 클릭한다.



TIP:

혹시 위 그림과 같이 “설치” 버튼이 보이지 않는다면 “Online help” 링크를 클릭하세요.

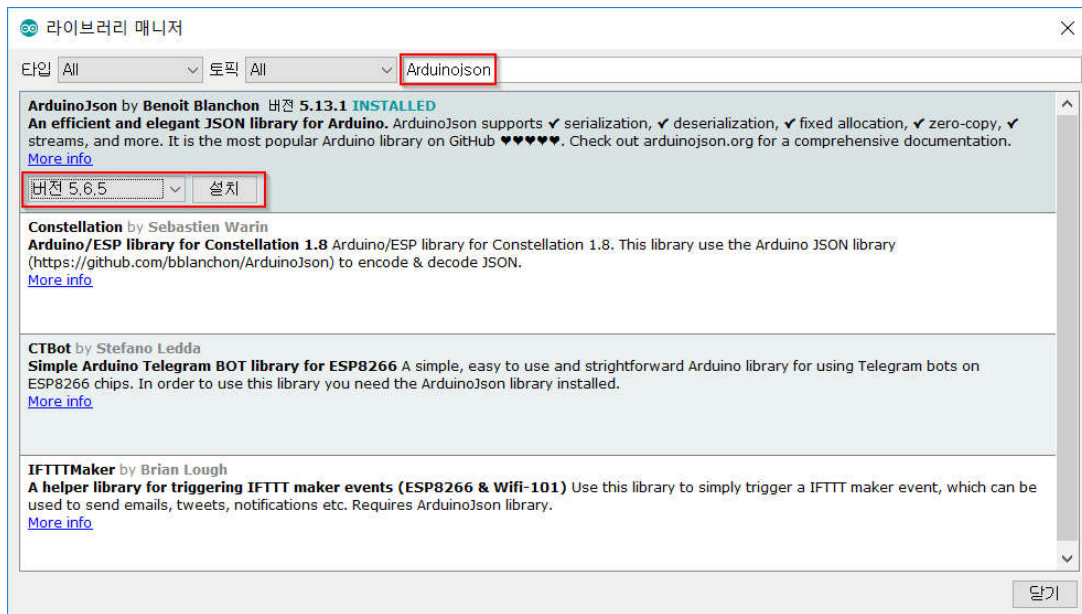
2) 사용 라이브러리 설치

IR 디바이스는 다음의 라이브러리를 필요로 한다.

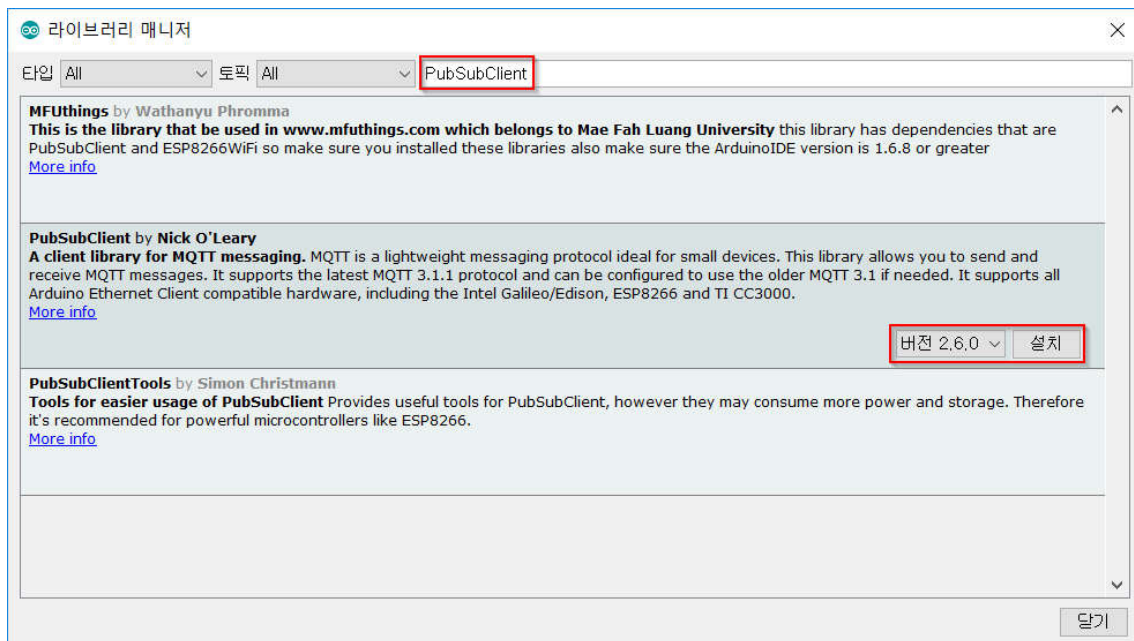
- PubSubClient (2.6.0)
- Arduinojson (5.6.5)
- IRremoteESP8266 (2.3.1)

메뉴 “스케치 > 라이브러리 포함하기 > 라이브러리 관리”를 선택한다.

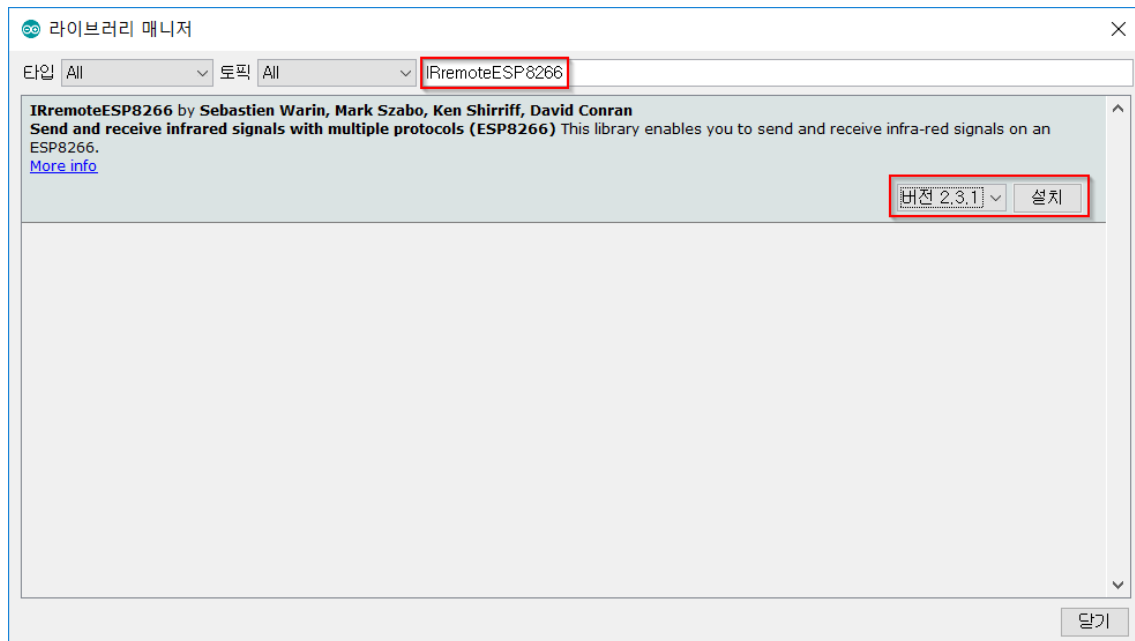
필터에 “Arduinojson”를 입력하고 설치한다.



필터에 "PubSubClient"를 입력하고 설치한다.



필터에 "IRremoteESP8266"를 입력하고 설치한다.



6. 프로그램 업로드

이제 개발환경 구성이 완료 되었으므로 아두이노 프로그램을 업로드 해야 한다.

여기서 사용하는 보드는 CP2102 USB to UART를 사용하므로 관련 드라이버를 설치해야 한다.

다음의 URL에서 드라이버를 다운로드 하여 설치한다.

<https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers#windows>

그리고 아두이노 프로그램은 다음의 URL에서 다운로드 할 수 있다.

"IR_DEVICE.zip" 파일을 다운로드 하여 압축을 해제한다.

<https://goo.gl/Xbk1LP>

6.1 라이브러리 수정

MQTT로 전송되는 리모컨 정보는 라이브러리가 제공하는 크기보 크다. 그러므로 라이브러리의 패킷 크기를 변경해야 한다.

라이브러리는 다음의 위치에 존재하다.

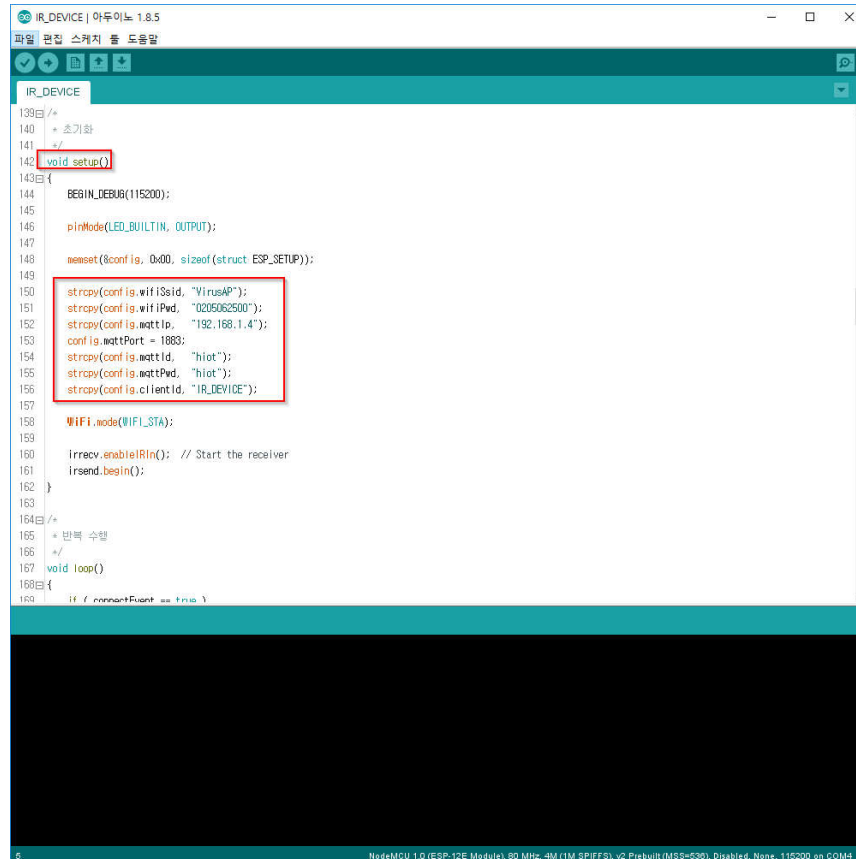
[사용자 홈]\Documents\Arduino\libraries\PubSubClient\src\PubSubClient.h

파일을 텍스트 편집기로 열어 패킷 사이즈를 1024로 변경한다.

```
// MQTT_MAX_PACKET_SIZE : Maximum packet size
#ifndef MQTT_MAX_PACKET_SIZE
#define MQTT_MAX_PACKET_SIZE 1024
#endif
```

6.2 WIFI 정보 수정

아두이노 IDE에서 “파일 > 열기” 메뉴를 선택하여 압축 해제한 **IR_DEVICE.ino** 파일을 엽니다. 소스에서 “setup” 함수에 있는 WIFI, MQTT 연결정보를 수정한다.



변수	설명
wifiSsid	무선 공유기의 SSID를 입력한다.
wifiPwd	무선 공유기의 비밀번호를 입력한다.
mqttIp	라즈베리파이의 IP 주소를 입력한다.
mqttId	MQTT Broker에 설정된 ID로 “hiot”를 입력한다.
mqttPwd	MQTT Broker에 설정된 비밀번호로 “hiot”를 입력한다.

6.3 컴파일 및 업로드

정상적인 컴파일을 위해서는 대상 보드를 정확하게 지정해야 한다.

메뉴 “툴 > 보드 > **NodeMCU 1.0 (ESP-12E Module)**”를 선택한다.

그리고 보드가 연결된 시리얼 포트를 지정해야 한다.

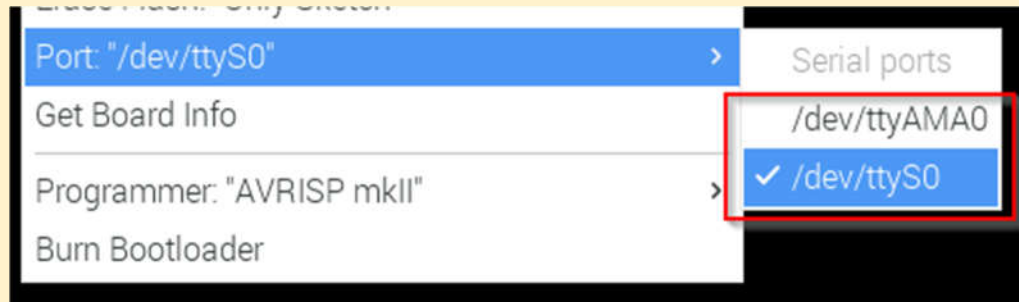
포트는 메뉴 “툴 > 포트 > COM?”에서 지정할 수 있다.

Meetup

라즈베리파이의 시리얼 포트는 윈도우와 다르게 COM? 형식으로 되지 않는다.

그러므로 "Tool > Port" 포트 메뉴에서 "/dev/tty?" 를 선택한다.

그리고 보드는 메뉴 "툴 > 보드 > Generic ESP8266 Module"을 선택한다.



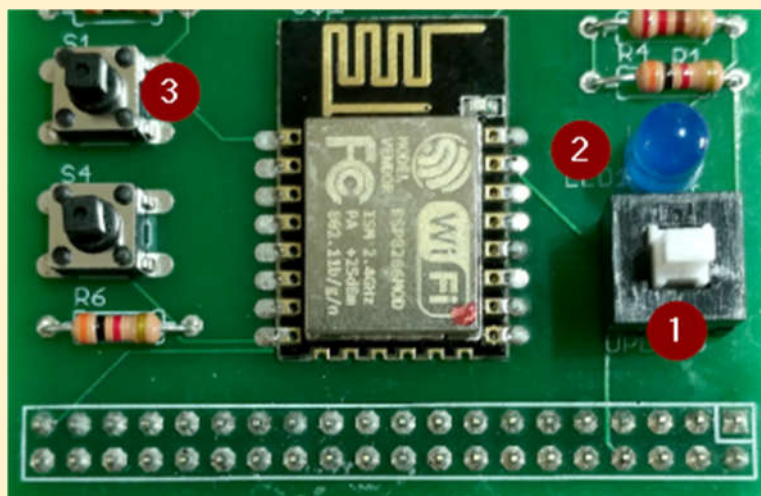
이제 메뉴에서 "스케치 > 확인/컴파일"을 선택하면 컴파일을 시작한다.



Meetup

업로드를 위해서는 ESP-12E의 상태를 펌웨어 업로드 상태로 바꾸어야 한다.

다음 그림에서 (1) 버튼을 누르면 (2) LED가 켜지며 (3) 버튼을 눌러 ESP-12E를 RESET 한다.



컴파일이 정상적으로 끝나면 "스케치 > 업로드"를 선택하여 펌웨어를 업로드 한다.

```

업로딩...
head_sector_count: 16
adjusted_sector_count: 55
erase_size: 037000
espcomm_send_command: sending command header
espcomm_send_command: sending command payload
setting serial port timeouts to 15000 ms
setting serial port timeouts to 1000 ms
espcomm_send_command: receiving 2 bytes of data
writing flash
..... [ 28% ]
..... [ 56% ]

```

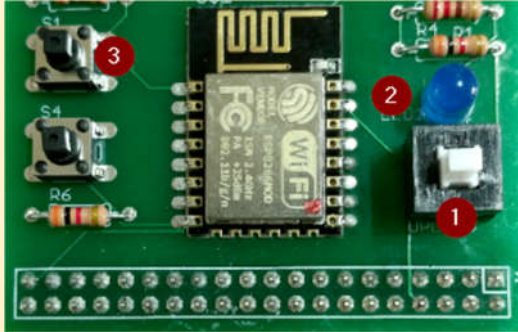
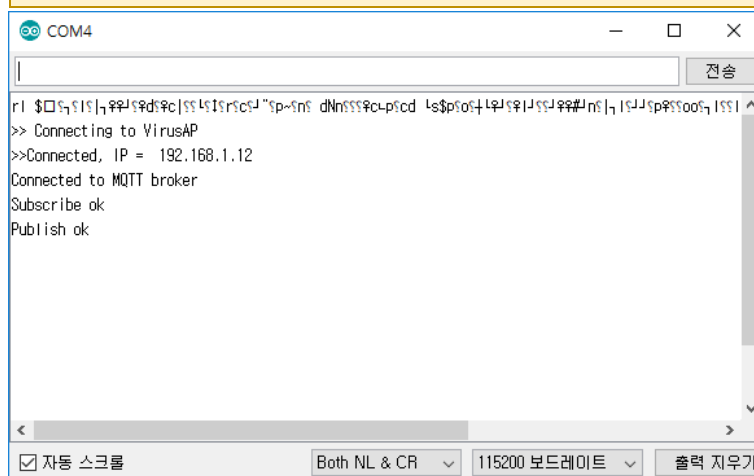
6.4 동작 확인

펌웨어가 업로드 되고나면 디바이스는 재시작을 하게 된다. 동작 상태는 시리얼 포트로 출력이 되는데, 메뉴의 "툴 > 시리얼 모니터"를 선택하면 다음과 같은 화면이 나타난다.

NodeMCU 보드의 "RST" 버튼을 클릭한다.

Meetup

그림에서 (1) 버튼을 눌러 (2) LED를 끄고 (3) 버튼을 눌러 ESP-12E를 RESET 한다.

WIFI, MQTT Broker에 연결이 완료되면 보드의 LED가 5회 점멸한다. 그러면 디바이스는 정상적으로 동작하는 것이다.

7. 리모콘 데이터 입력하기

앞에서 만든 디바이스로 텔레비전, Set-top box, 에어컨 등을 제어하기 위해서는 사용 중인 리모콘의 데이터를 입력해야 한다. 데이터를 저장하는 테이블은 IR_RAW이며, 입력 방법은 다음과 같다.

먼저 다음의 URL에서 "IR 입력 양식.xlsx" 파일을 다운로드 한다.

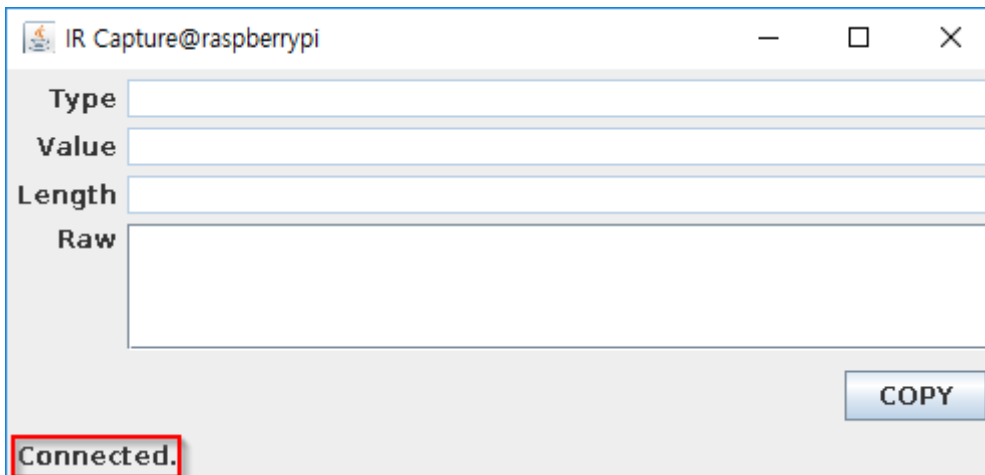
<https://goo.gl/Xbk1LP>

7.1 리모콘 데이터 캡처

리모콘 디바이스에 전원을 입력하고, 라즈베리파이에 SmartTTY로 접속하여 다음의 명령을 실행한다.

```
cd ~/ai4w  
./ir.sh
```

다음과 같은 화면이 나타나며, "Connected"라는 문구가 출력되면 MQTT Broker에 정상적으로 접속이 된 것이다.



IR Capture@raspberrypi

Type

Value

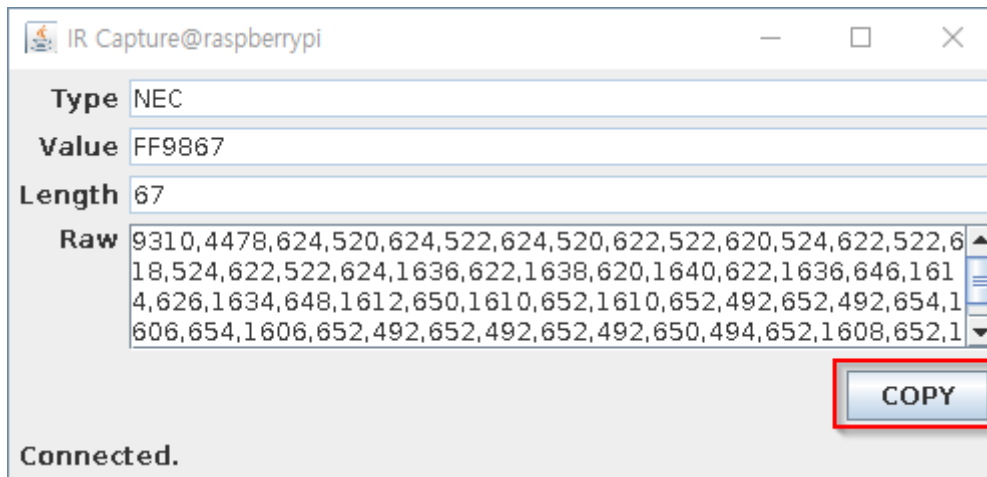
Length

Raw

COPY

Connected.

이제 사용 중인 리모콘을 눌러보자. 그러면 캡처된 정보가 화면에 나타난다.



이제 다운로드 한 엑셀 파일을 엽니다. 그리고 위 그림에서 "COPY" 버튼을 클릭하여 내용을 클립보로 복사합니다. 이제 엑셀 파일을 "B 열(TYPE)"에서 "붙여넣기"를 수행합니다. 그러면 데이터가 입력된 것을 확인할 수 있습니다.

KEY	TYPE	VALUE	LENGTH	RAW DATA
IPTV_0	NEC	FF02FD	67	9306,4466,662,482,662,484,662,484,662,482,654,490,660,484,662,482,654,492,664,1596,658,1604,666,1594,656,1604,666,1594,666,1596,666,1596,664,1596,662,484,664,480,664,480,664,482,662,482,662,482,664,1598,664,480,664,1596,654,1608,666,1596,670,1596,660,1602,664,1598,660,484,664,1598,664
IPTV_1	NEC	FF38C7	67	9236,4480,648,496,648,496,646,496,648,494,650,494,648,496,648,496,648,494,648,1612,648,1610,648,1610,646,1614,646,1612,644,1614,646,1612,646,1612,648,496,646,496,646,1612,648,1610,644,1612,646,496,646,498,644,496,646,1610,626,1634,624,520,648,496,648,496,626,1636,646,1612,650,1612,648
IPTV_2				
IPTV_3				
IPTV_4				

이런 방식으로 사용하려는 리모컨의 키를 캡처하여 엑셀을 채웁니다.

엑셀에는 IPTV_0 ~ IPTV_9, TV_0 ~ TV_9 항목이 존재합니다. 이것을 케이블 TV와 일반 TV를 사용하는 경우를 나타냅니다. 만일 케이블 TV가 없다면 리모컨 캡처 정보를 TV 부분이 아닌 IPTV 부분에 입력합니다. 이것은 Watson Conversation의 엔티티 부분에서 IPTV를 사용하기 때문입니다.

7.2 IR_RAW 테이블 입력

엑셀에 있는 데이터를 테이블에 입력하기 위해 MySQL Workbench를 사용할 것이다. 이것은 다음의 URL에서 다운로드 할 수 있다.


<https://dev.mysql.com/downloads/workbench/>

MySQL Workbench 6.3.10

Select Operating System:
 Microsoft Windows ▼

[Looking for previous GA versions?](#)

Recommended Download:



MySQL Installer for Windows

All MySQL Products. For All Windows Platforms. In One Package.

Starting with MySQL 5.6 the MySQL Installer package replaces the standalone MSI packages.

Windows (x86, 32 & 64-bit), MySQL Installer MSI

[Go to Download Page >](#)

Other Downloads:

Windows (x86, 64-bit), MSI Installer	6.3.10	28.2M	Download
(mysql-workbench-community-6.3.10-winx64.msi)			
MD5: a445c5f1a362f360da1ad25ab4d2de0f Signature			

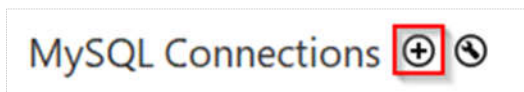
[Login »](#)
 using my Oracle Web account

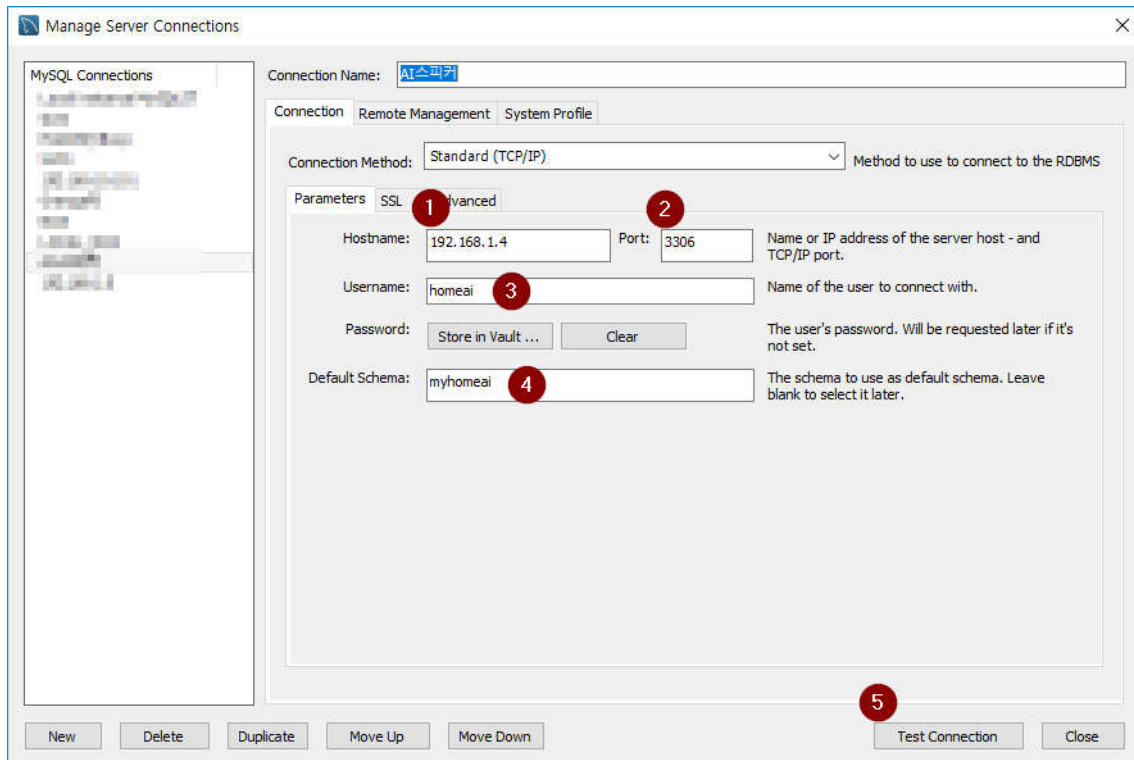
[Sign Up »](#)
 for an Oracle Web account

MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions.

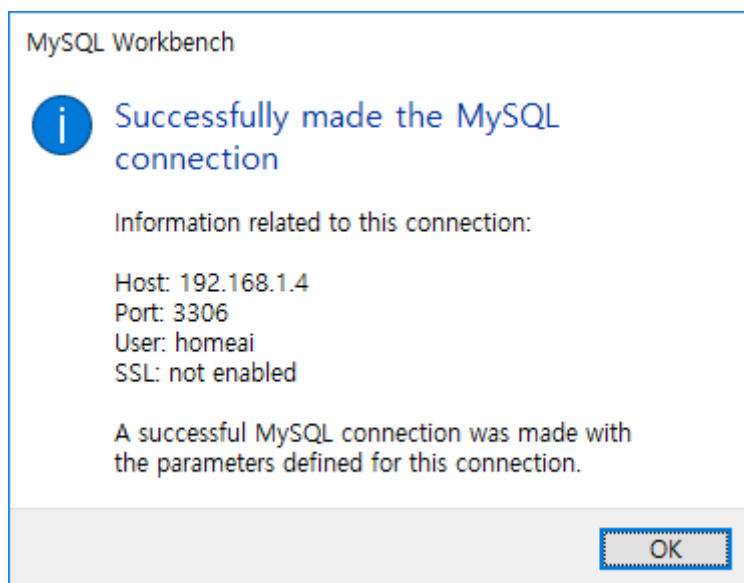
[No thanks, just start my download.](#)

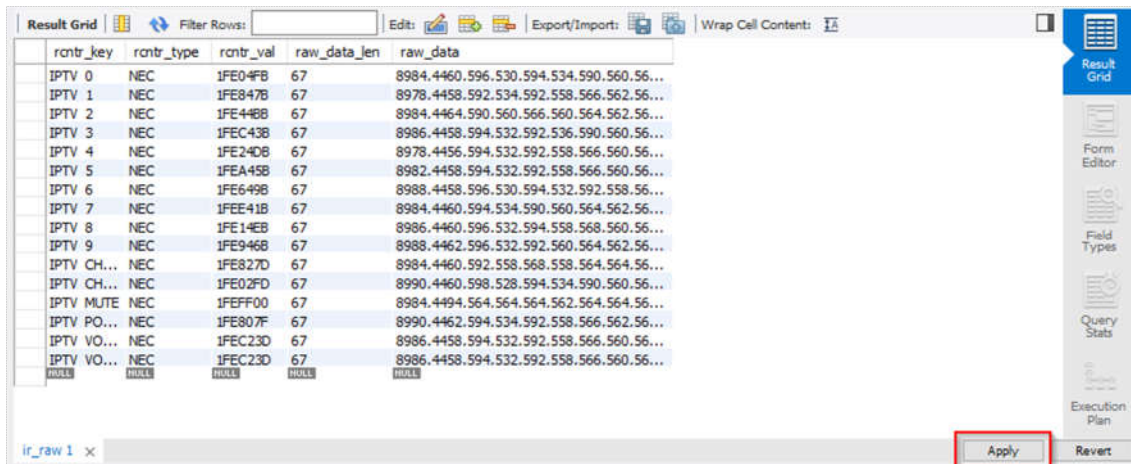
설치가 완료되면 Workbench를 실행한다.
 먼저 라즈베리파이의 MySQL 연결을 위해 연결정보를 입력해야 한다.
 화면에서 (+) 버튼을 클릭한다.





- ① 라즈베리파이의 IP 주소를 입력한다.
- ② MySQL 서버의 포트 3306을 입력한다.
- ③ MySQL 서버 사용자 ID "**homeai**"를 입력한다.
- ④ 스키마 "**myhomeai**"를 입력한다.
- ⑤ 입력된 정보로 연결을 테스트 한다. 이때 비밀번호 "**homeai**"를 입력한다.
정상적으로 연결이 되면(경고 무시) 다음과 같은 화면이 나타난다.



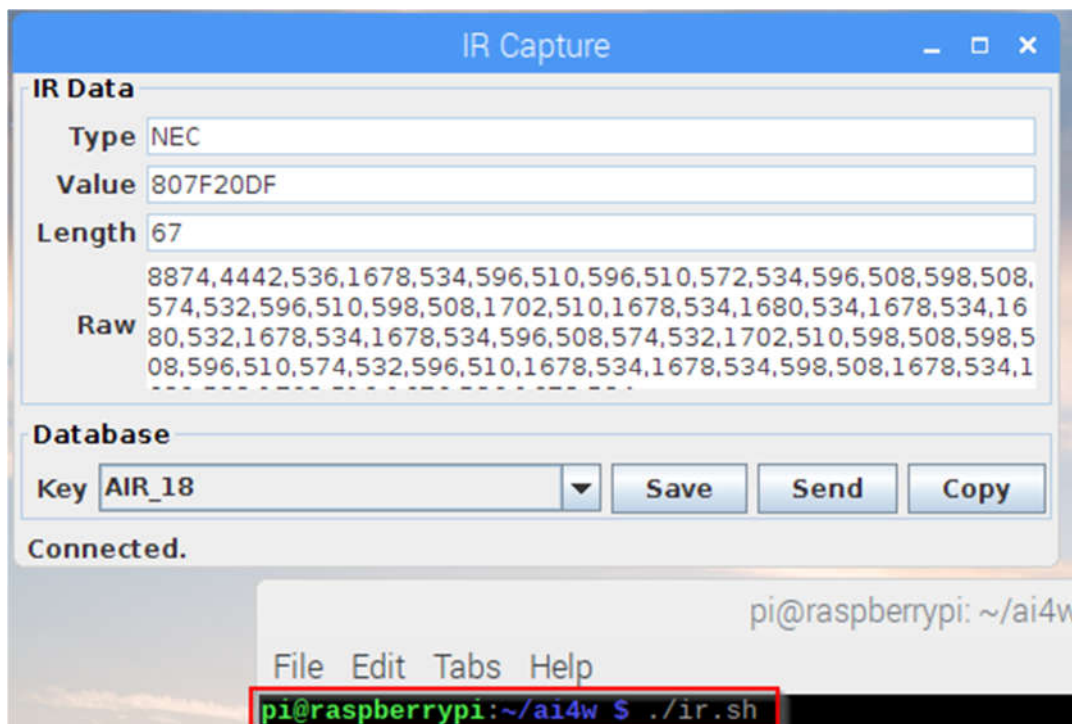


"Apply" 버튼을 클릭하여 데이터베이스에 저장한다.

7.3 IR_RAW 테이블 입력 (Meetup)

원격 데스크톱 연결을 하고 터미널에서 다음의 명령을 실행한다.

```
cd ~/ai4w
./ir.sh
```



화면 하단에 "Connected"라는 문구가 출력되면 MQTT Broker에 정상적으로 접속이 된 것이다. 캡처한 리모컨 데이터를 "IR_RAW" 테이블에 저장하는 방법은 다음과 같다.

- ① 캡처를 원하는 리모컨 버튼을 누른다. 화면에 리모컨 정보가 나타난다.
- ② "Key" 부분에서 맵핑시킬 키를 선택한다.
- ③ "Save" 버튼을 클릭하여 정보를 저장한다.

만일 현재 저장된 키외에 추가적으로 키를 넣기 원하는 경우 "IR_RAW" 테이블의 "rcntr_key" 컬럼에 키를 추가하고 "ir.sh"를 다시 실행한다.

8. Watson Conversation과 IoT 디바이스

Watson Conversation은 자연어에서 인텐트(Intent)와 엔티티(Entity)를 분석하고, 설계된 다이얼로그를 이용하여 적절한 응답을 한다.

여기서 인텐트는 자연어가 어떤 의미 인지를 판단하고, 엔티티는 그 의미의 대상이 되는 것을 나타낸다,

예를 들어 "텔레비전 켜 주세요", "내방 전등을 켜 주세요" 라는 문장이 있는 경우, 인텐트는 어떤 것을 켜다는 의미로 정의할 수 있으며, 엔티티는 텔레비전, 전등 등으로 정의할 수 있다.

스피커에서 음성을 리모컨 버튼으로 매핑하기 위해 다음의 규칙을 사용한다.

- 인텐트로 음악을 들려줄지 아니면 텔레비전, 에어컨 등을 제어 해야 할지 판단한다.
- 엔티티로 대상은 무엇인지 어떤 동작을 해야 하는지를 판단한다.
- 엔티티 정보를 리모컨 키 코드와 매핑시킨다.

① 해석된 의도에 따라 음악 관련된 기능, TV 제어 등의 기능을 수행하는 액션을 호출한다.

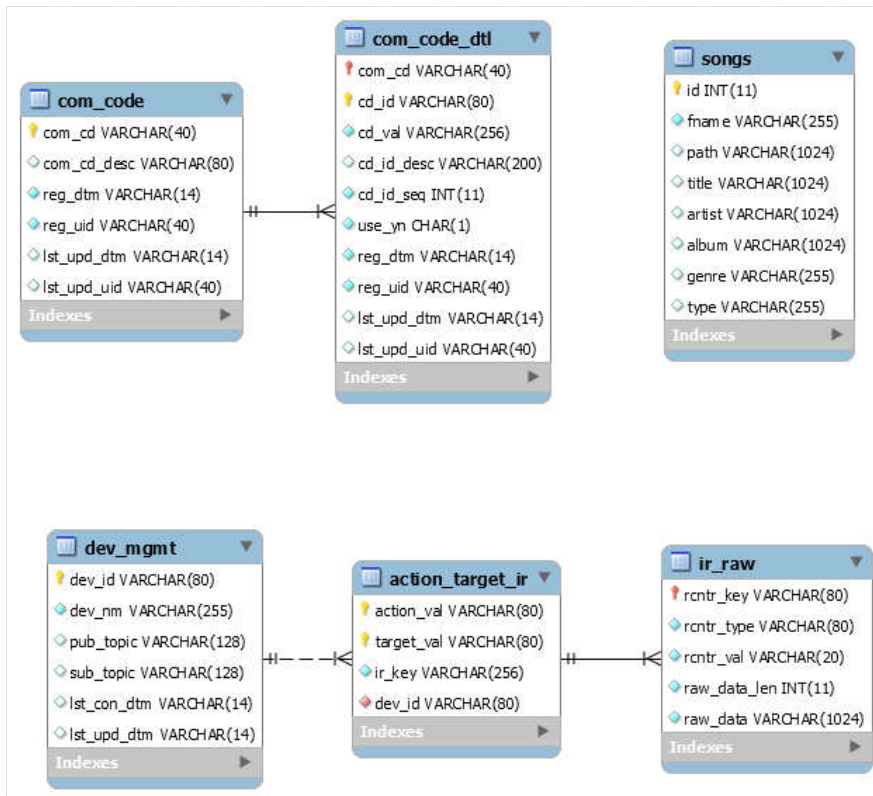
② 제어를 위해 3가지의 엔티티를 정의한다.

- @action 엔티티: 어떤 동작(turnon, turnoff, play 등)을 해야 하는지를 나타낸다.
- @target 엔티티: 동작 대상(TV, 에어컨)이 무엇인지를 나타낸다.
- @channel 엔티티: 케이블 TV의 채널을 나타낸다. (의미상으로는 @target과 동일)
- @temp 엔티티: 에어컨의 온도를 나타낸다.

action	target	channel		temp	
down	volumn	CH_9	CH_13	AIR_18	AIR_24
turnoff	music	CH_7	CH_170	AIR_19	AIR_25
turnon	aircon	CH_23	CH_16	AIR_20	AIR_26
play	tv	CH_5	CH_17	AIR_21	AIR_27
up	light	CH_53	CH_15	AIR_22	AIR_28
set		CH_55	CH_11	AIR_23	AIR_29

③ action_target_ir 테이블에는 action과 target이 어떤 리모컨 키와 연결되는지를 정의한다.

테이블 구조는 비교적 간단하며, Waton Conversation의 엔티티 값으로 키를 찾을 수 있도록 하였다.



1 • `SELECT * FROM myhomeai.action_target_ir;`

action_val	target_val	ir_key
plav	CH 11	IPTV 1.IPTV 1
plav	CH 13	IPTV 1.IPTV 3
plav	CH 15	IPTV 1.IPTV 5
plav	CH 16	IPTV 1.IPTV 6
plav	CH 17	IPTV 1.IPTV 7
plav	CH 170	IPTV 1.IPTV 7.IPTV 0
plav	CH 178	IPTV 1.IPTV 7.IPTV 8
plav	CH 179	IPTV 1.IPTV 7.IPTV 9
plav	CH 18	IPTV 1.IPTV 8
plav	CH 19	IPTV 1.IPTV 9
plav	CH 20	IPTV 2.IPTV 0
plav	CH 200	IPTV 2.IPTV 0.IPTV 0
plav	CH 202	IPTV 2.IPTV 0.IPTV 2
plav	CH 203	IPTV 2.IPTV 0.IPTV 3
plav	CH 204	IPTV 2.IPTV 0.IPTV 4
plav	CH 23	IPTV 2.IPTV 3
plav	CH 24	IPTV 2.IPTV 4
plav	CH 30	IPTV 3.IPTV 0
plav	CH 32	IPTV 3.IPTV 2
plav	CH 5	IPTV 5
plav	CH 51	IPTV 5.IPTV 1
plav	CH 52	IPTV 5.IPTV 2
plav	CH 53	IPTV 5.IPTV 3
plav	CH 54	IPTV 5.IPTV 4
plav	CH 55	IPTV 5.IPTV 5
plav	CH 7	IPTV 7
plav	CH 9	IPTV 9
plav	CH 95	IPTV 9.IPTV 5
turnoff	aircon	AIR POW OFF
turnoff	tv	TV POWER.IPTV POWER
turnon	aircon	AIR POW ON
turnon	tv	TV POWER.IPTV POWER

ir_key는 콤마(,)로 구분하여 2개 이상을 입력할 수 있다.

APPEND: 소스코드

1) AI Speaker for Watson

https://github.com/TechOrgg/ai_speaker4_watson.git

2) Android & Watson Assistant (with NAVER)

Branch: master ▾	New pull request	Create new file	Upload files	Find file	Clone or download ▾
Choi Eui Shin - SFA 문서 추가 ...					Latest commit 2fa9f38 on 27 Feb
android/MyHomeAISpeaker		소스 및 문서	4 months ago		
doc		- SFA 문서 추가	4 months ago		
iot		소스 및 문서	4 months ago		
server		- SFA 문서 추가	4 months ago		
README.md		Update README.md	4 months ago		

https://github.com/TechOrgg/ai_speaker_iot.git