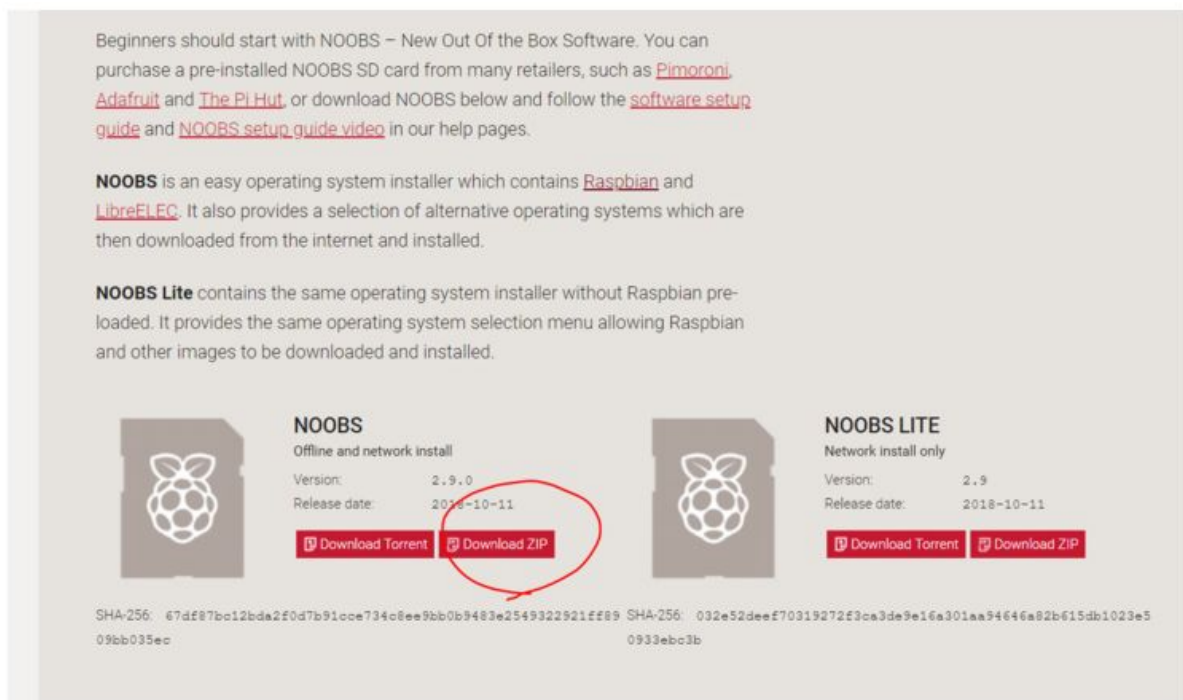
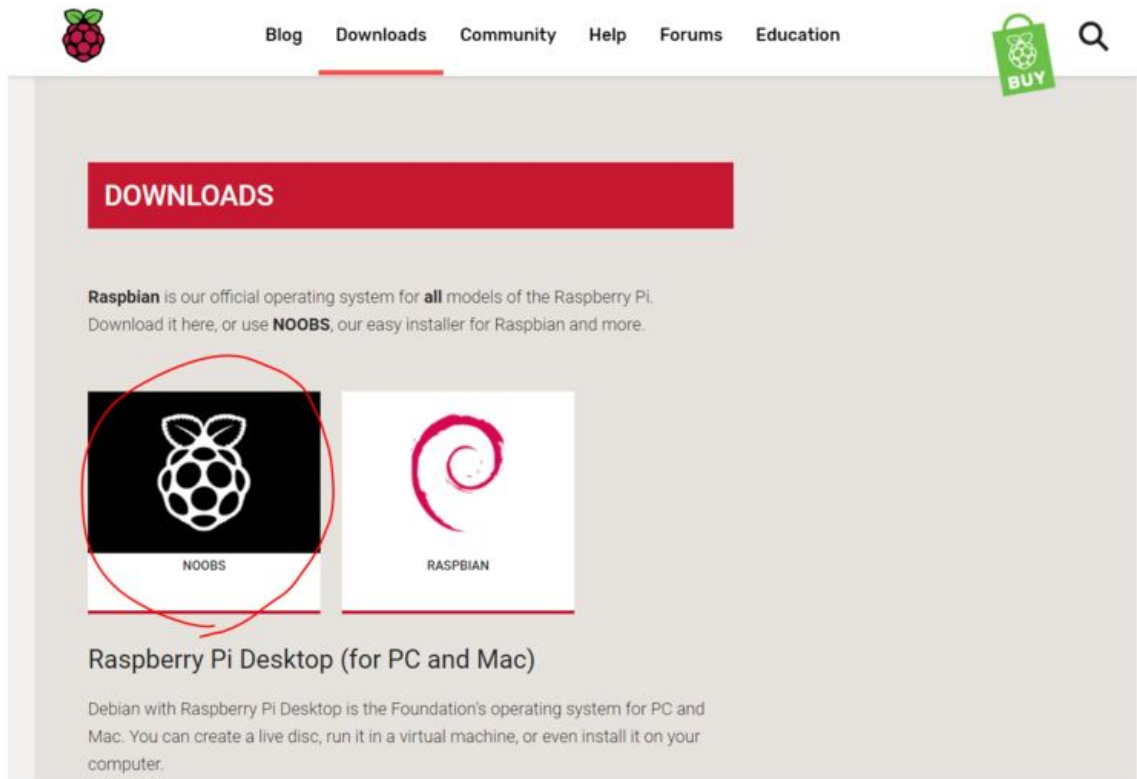


1. 라즈베리파이에 라즈비안 설치하기

- a. 컴퓨터에 sd카드를 인식하기
- b. <https://www.raspberrypi.org/downloads/> 링크에서 noobs.zip을 다운받아서 압축풀기



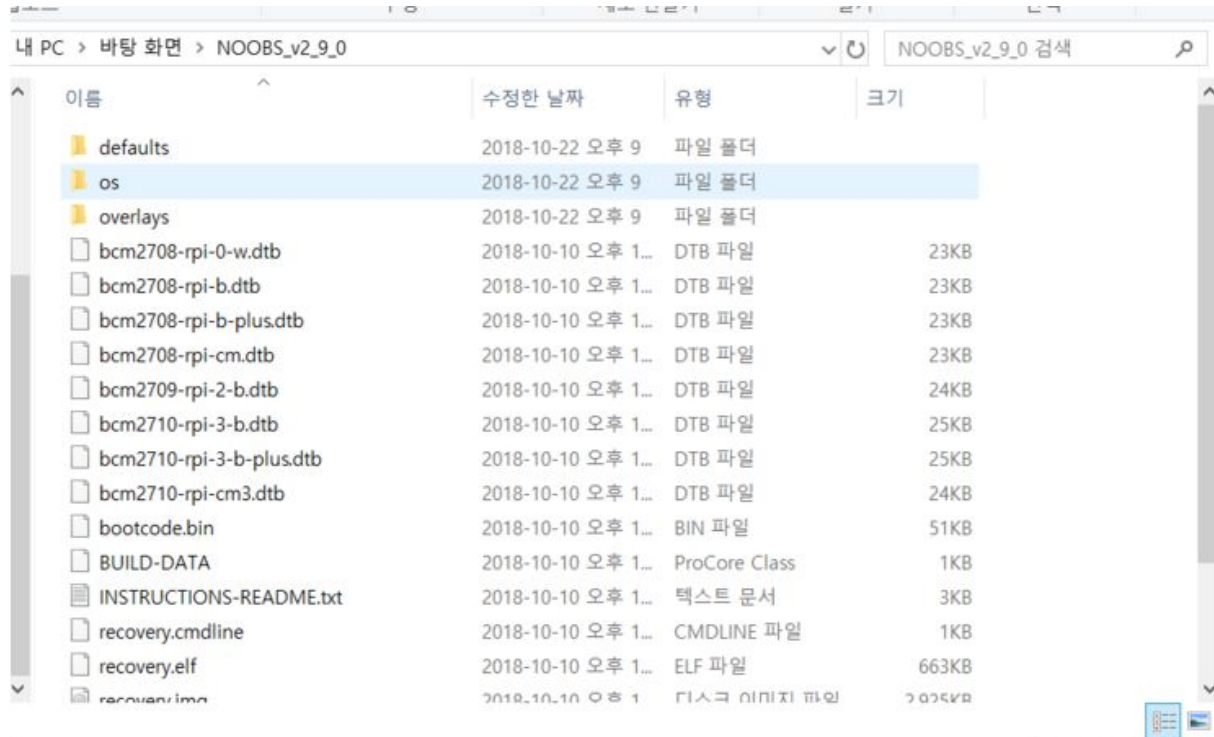
* Noobs vs Noobs Lite

두 버전의 차이점은 설치할 때 인터넷이 필요하냐 안필요하냐의 차이입니다.

Lite 버전이 라즈비안을 설치할 때 인터넷을 필요로 합니다.

2. 압축을 풀고 난 파일들을 sd 카드에 복사해 줍니다.

이때 주의해야 할 점은 압축을 풀고 생성된 폴더를 복사하는 것이 아니라 압축이 풀린 순수한 파일들만 sd카드에 옮겨주어야 합니다.

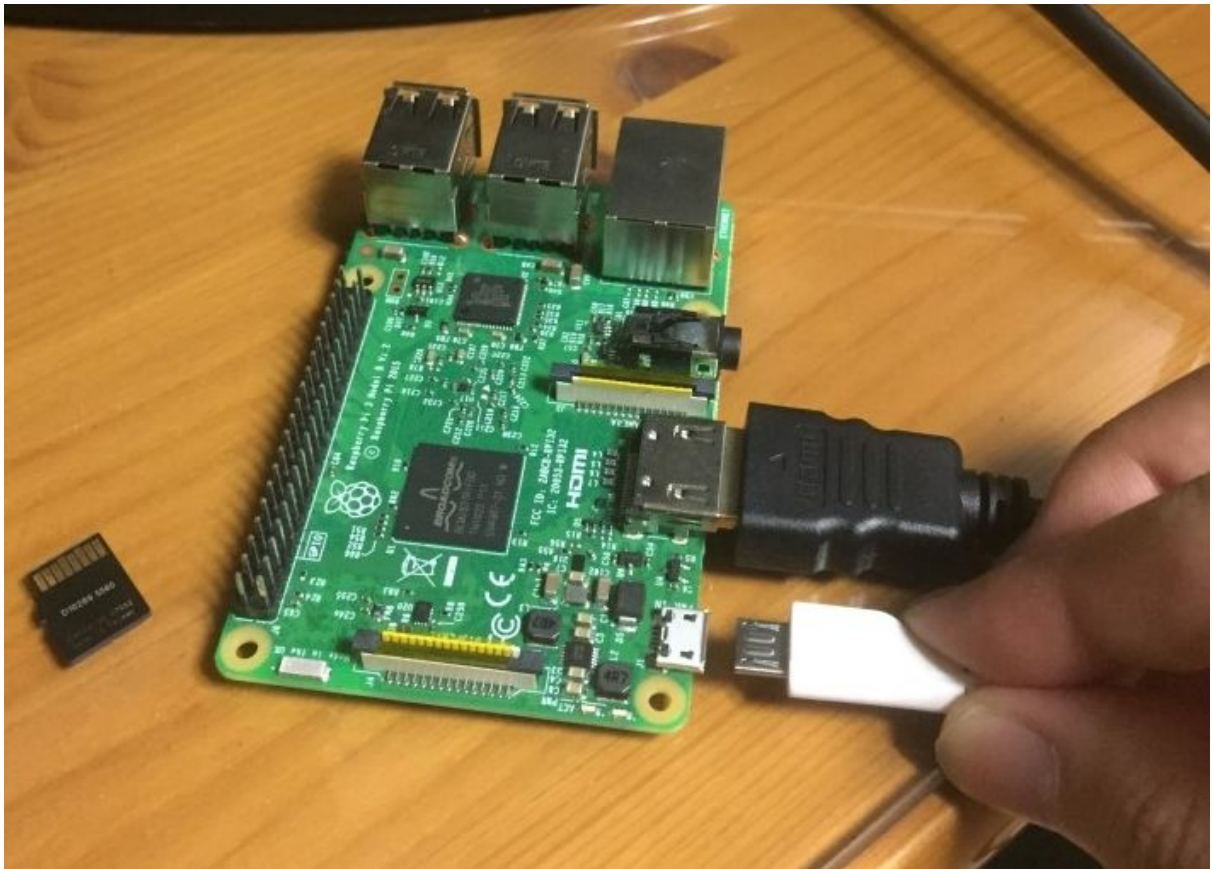


이름	수정된 날짜	유형	크기
defaults	2018-10-22 오후 9	파일 폴더	
os	2018-10-22 오후 9	파일 폴더	
overlays	2018-10-22 오후 9	파일 폴더	
bcm2708-rpi-0-w.dtb	2018-10-10 오후 1...	DTB 파일	23KB
bcm2708-rpi-b.dtb	2018-10-10 오후 1...	DTB 파일	23KB
bcm2708-rpi-b-plus.dtb	2018-10-10 오후 1...	DTB 파일	23KB
bcm2708-rpi-cm.dtb	2018-10-10 오후 1...	DTB 파일	23KB
bcm2709-rpi-2-b.dtb	2018-10-10 오후 1...	DTB 파일	24KB
bcm2710-rpi-3-b.dtb	2018-10-10 오후 1...	DTB 파일	25KB
bcm2710-rpi-3-b-plus.dtb	2018-10-10 오후 1...	DTB 파일	25KB
bcm2710-rpi-cm3.dtb	2018-10-10 오후 1...	DTB 파일	24KB
bootcode.bin	2018-10-10 오후 1...	BIN 파일	51KB
BUILD-DATA	2018-10-10 오후 1...	ProCore Class	1KB
INSTRUCTIONS-README.txt	2018-10-10 오후 1...	텍스트 문서	3KB
recovery.cmdline	2018-10-10 오후 1...	CMDLINE 파일	1KB
recovery.elf	2018-10-10 오후 1...	ELF 파일	663KB
recovery.img	2018-10-10 오후 1...	디스크 이미지 파일	7,975KB

3.sd카드에 noobs 파일을 저장했다면 라즈베리파이에 디스플레이를 연결해 라즈비안 설치를 합니다.(모니터와 라즈베리파이를 연결하기 위해 hdmi 연결선을 준비합니다.)



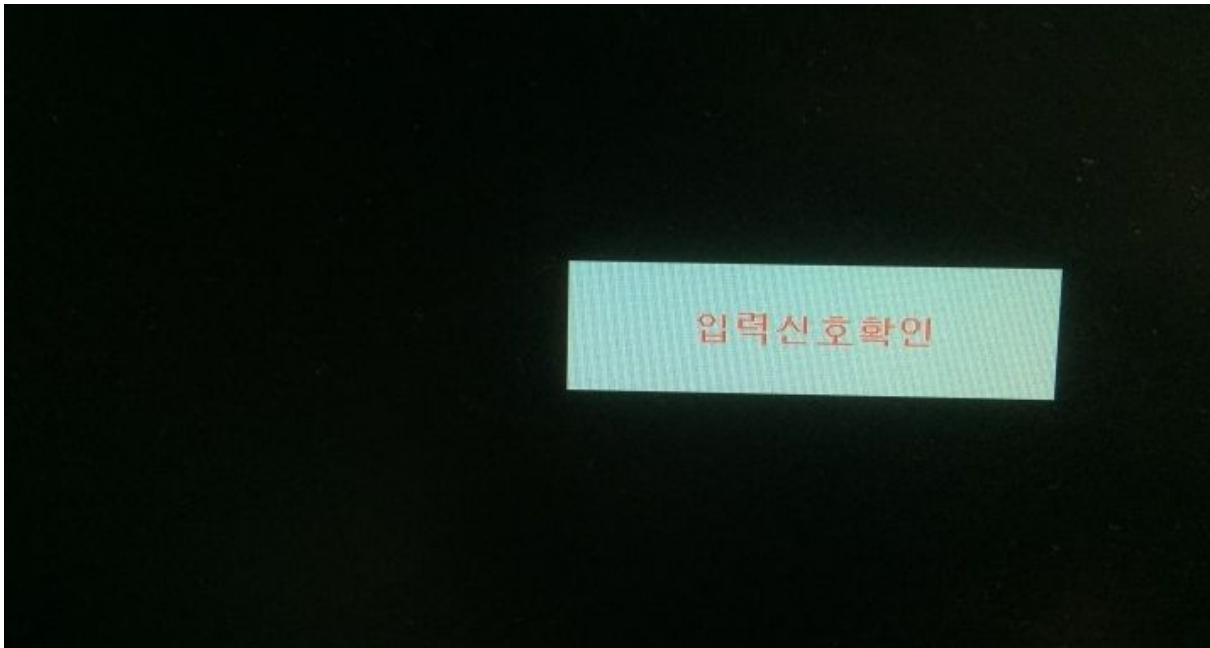
4.라즈베리 파이와 모니터를 연결합니다.



5.라즈베리파이에 micro 5핀 충전기를 이용해 전원을 공급합니다.

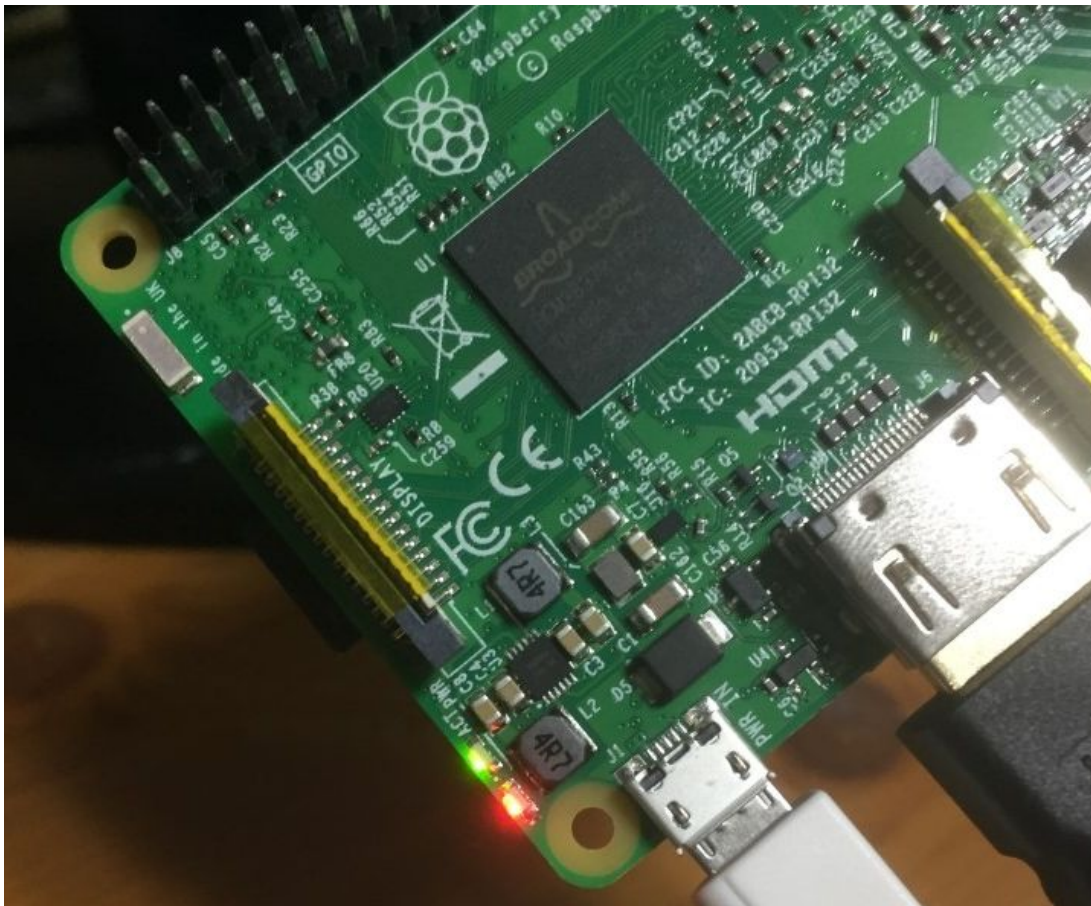


6.전원을 인가하면 빨간 불이 들어옵니다.



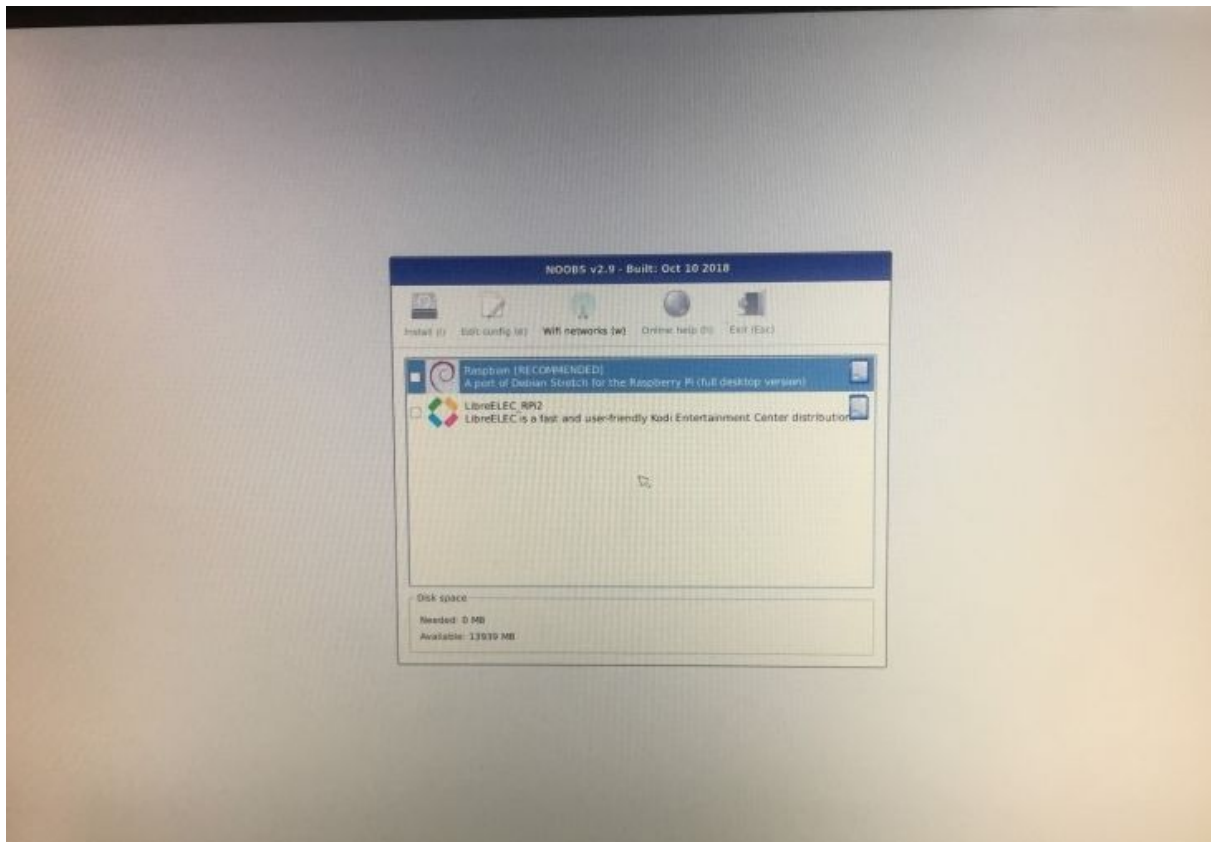
7. 하지만 모니터는 켜지지 않습니다.
이때 아까 noobs를 저장해 놓은 sd카드를 라즈베리파이에 삽입합니다.

라즈베리파이 usb포트에 마우스, 키보드를 연결해 줍니다.

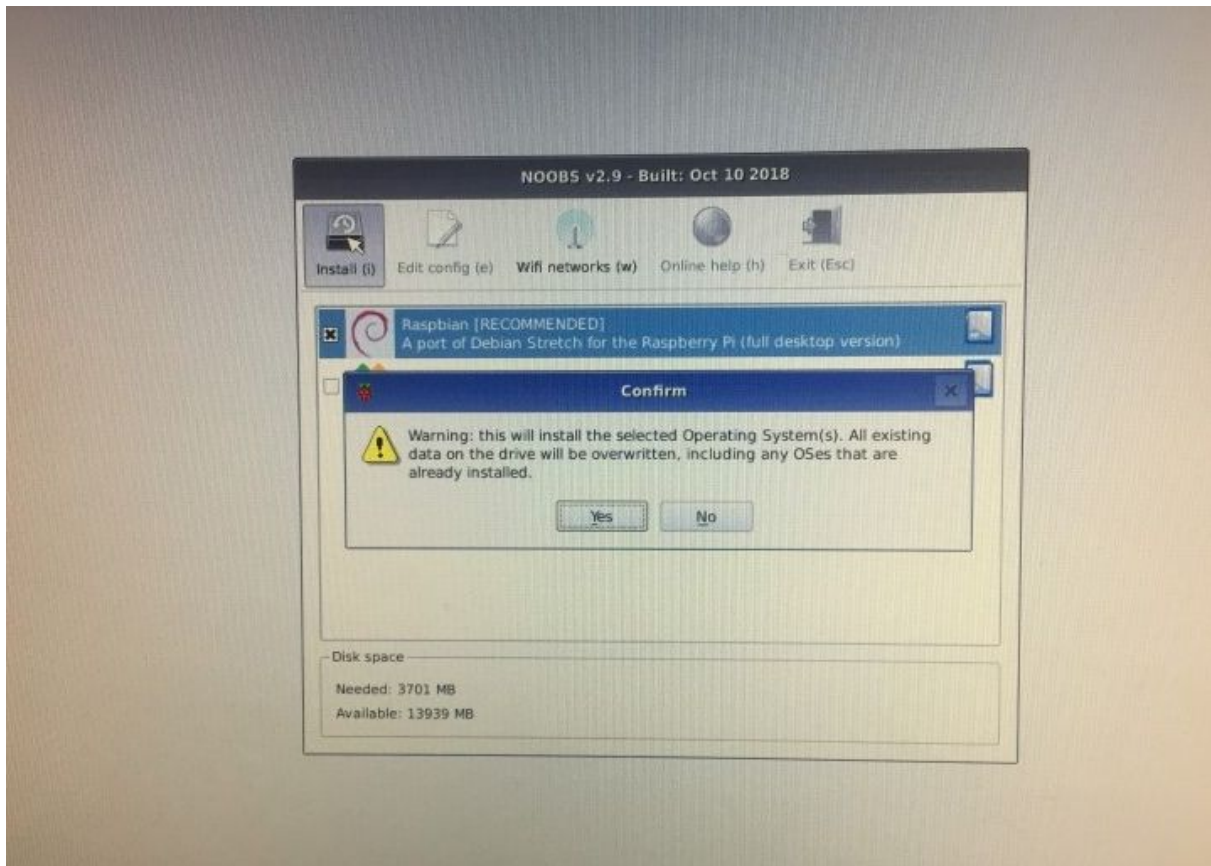


8. 빨간불 옆에 초록불이 켜졌습니다.
이제 라즈베리파이가 정상적으로 부팅됩니다.

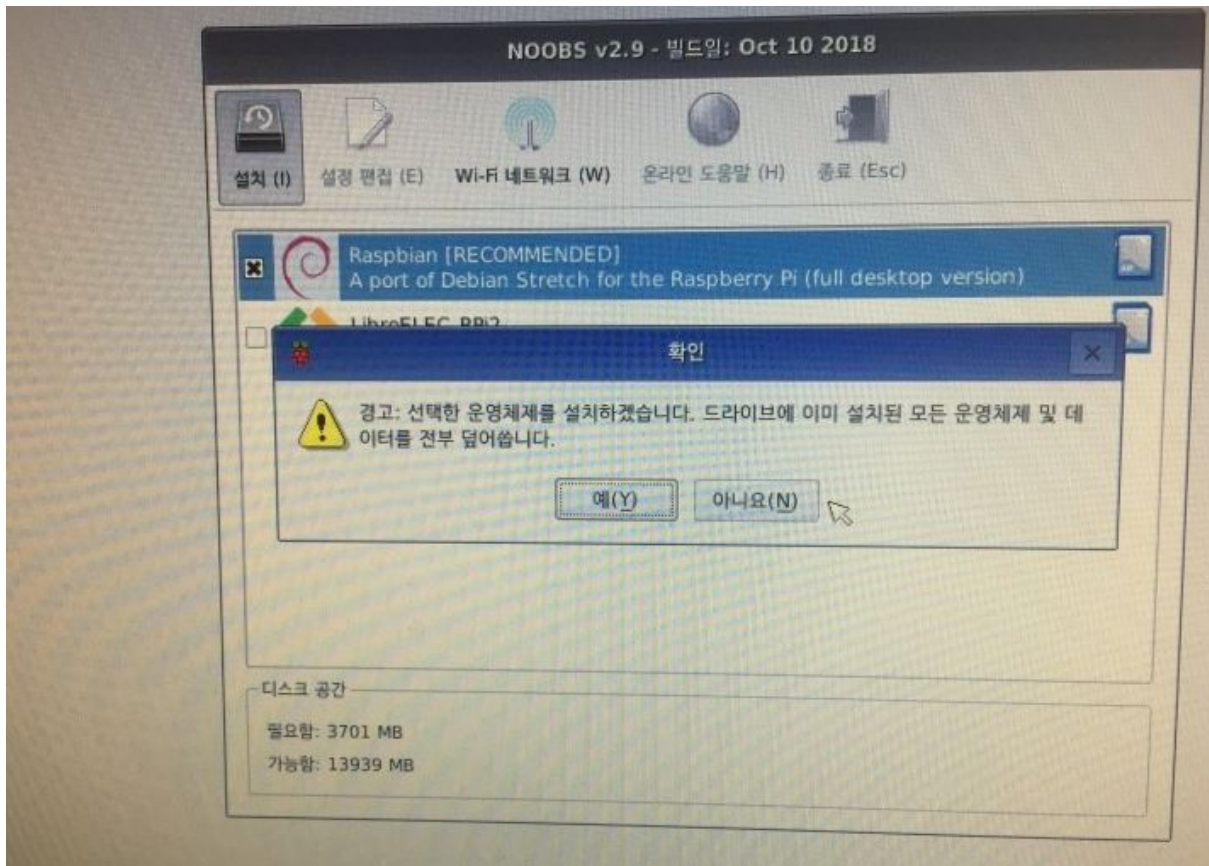
9.sd카드를 꽂은 라즈베리파이가 동작하면 아래와 같은 화면이 출력됩니다.



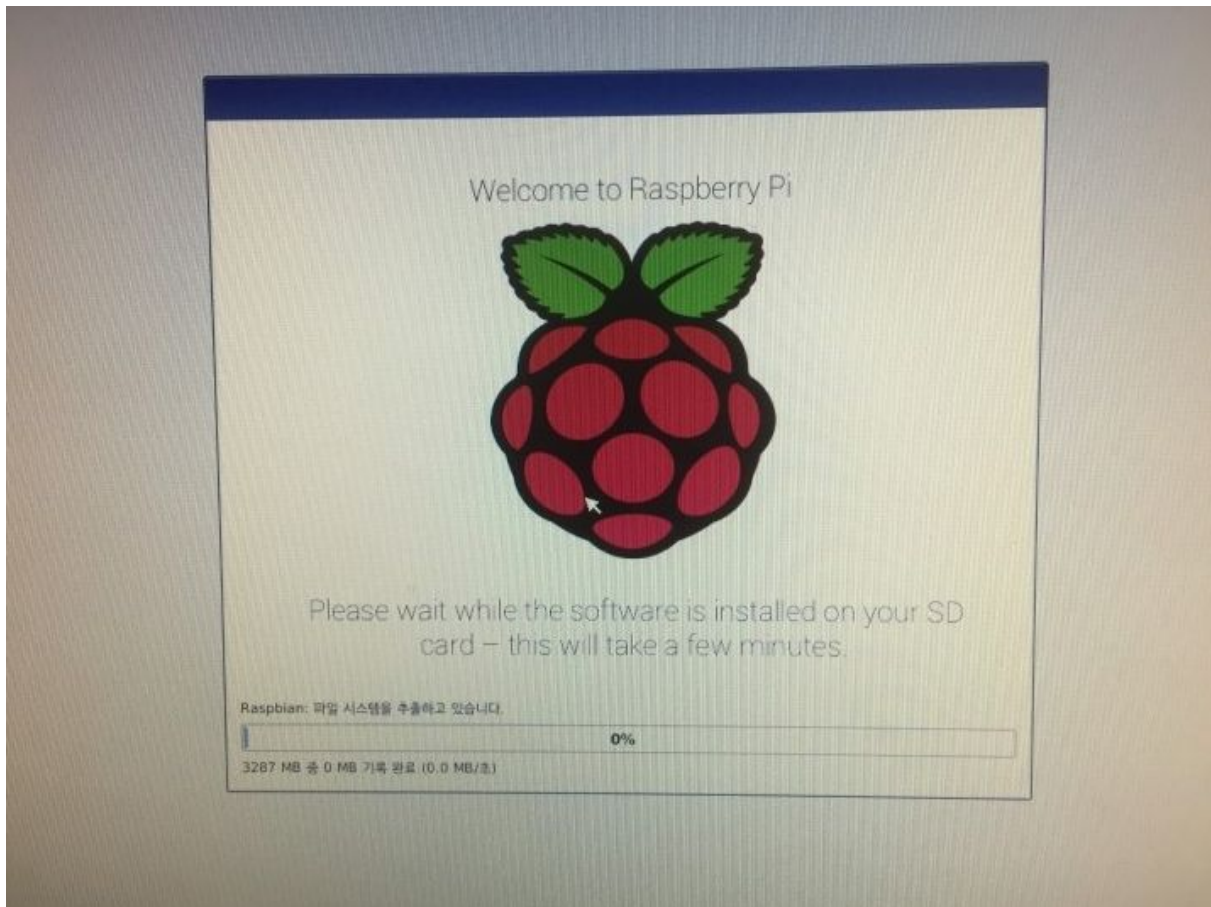
10.어떤 운영체제를 설치할지 물어봅니다. 라즈비안을 골라줍니다.



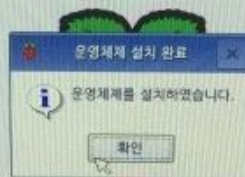
11.왼쪽 위의 install 버튼을 눌러 설치를 시작합니다. yes를 눌러주면 됩니다.




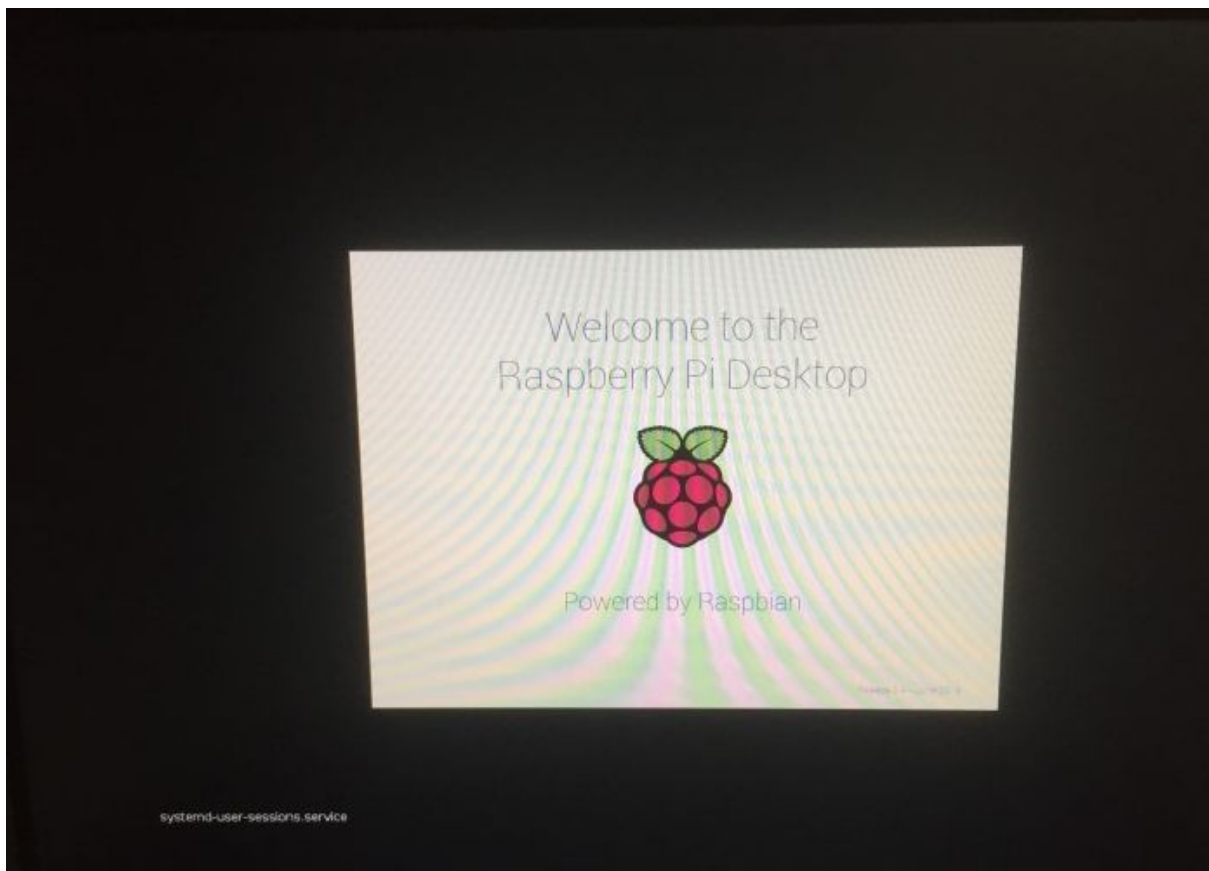
12. 화면 하단에 언어선택 기능을 선택하여 한국어를 선택해 줍니다.

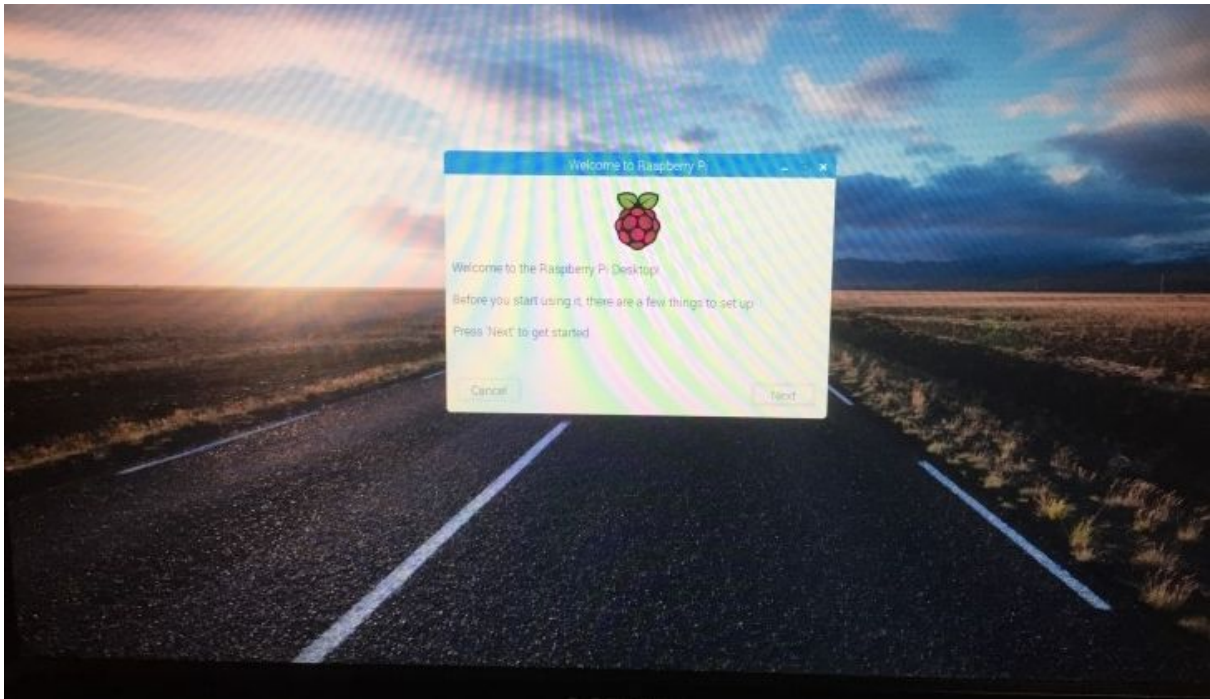


13. 설치가 시작됩니다.

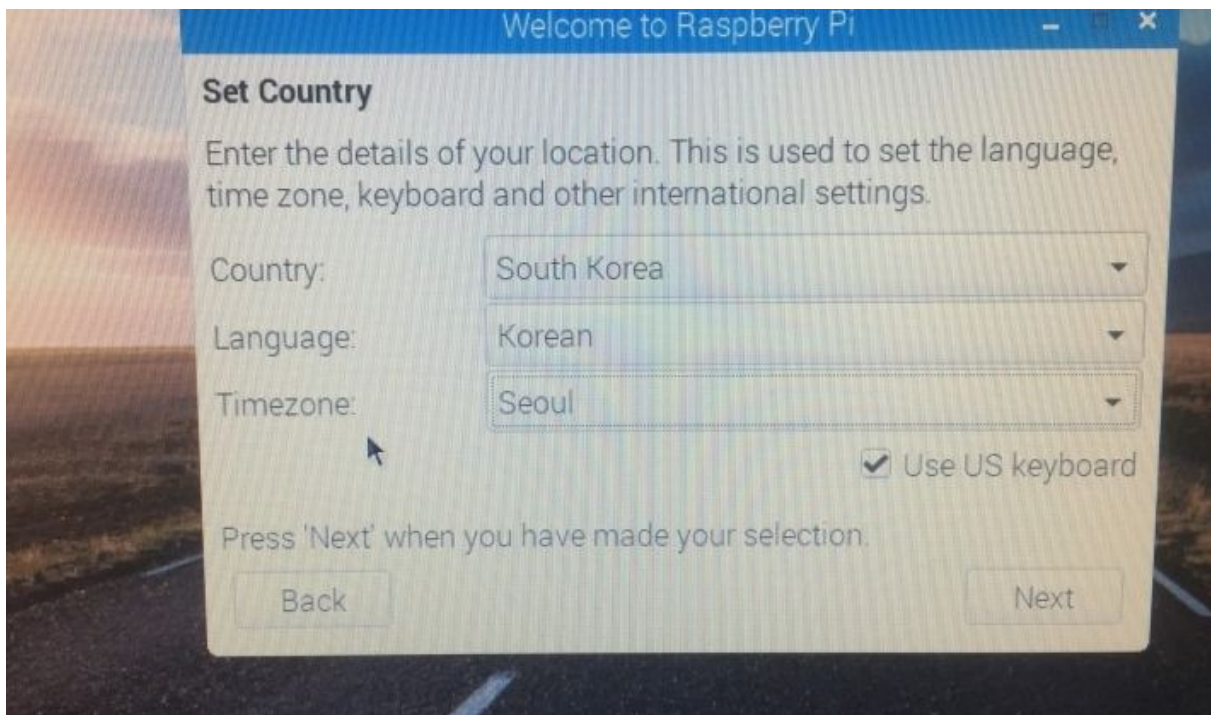


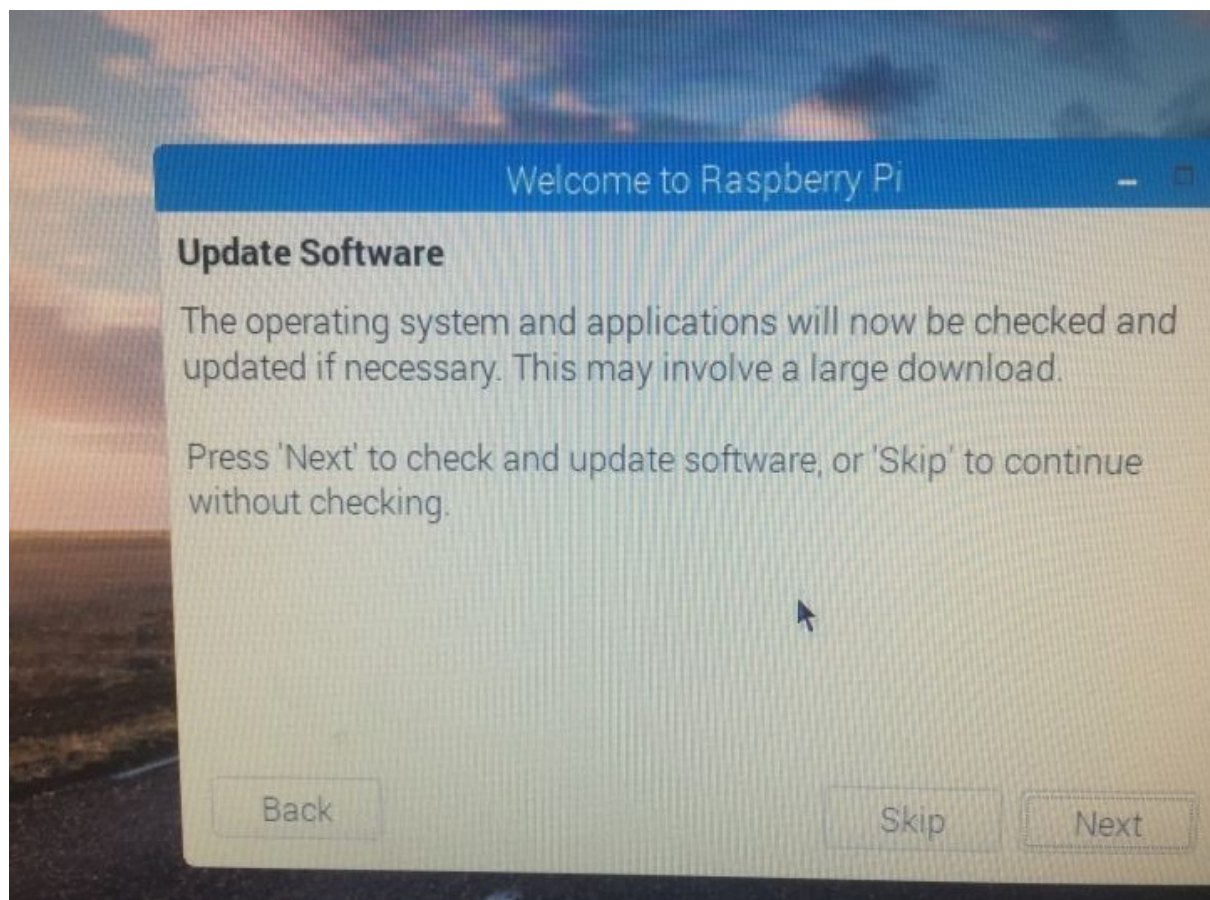
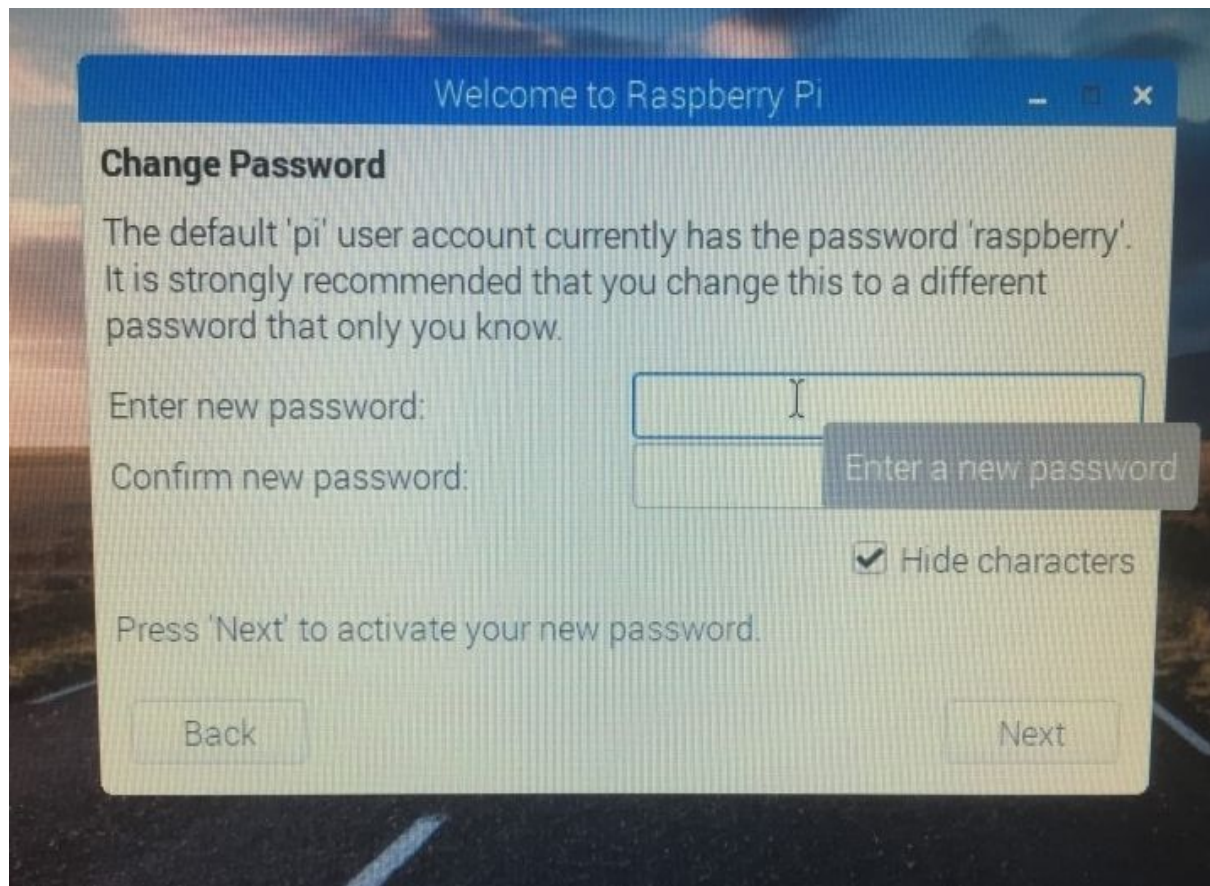
For recovery mode, hold  Shift

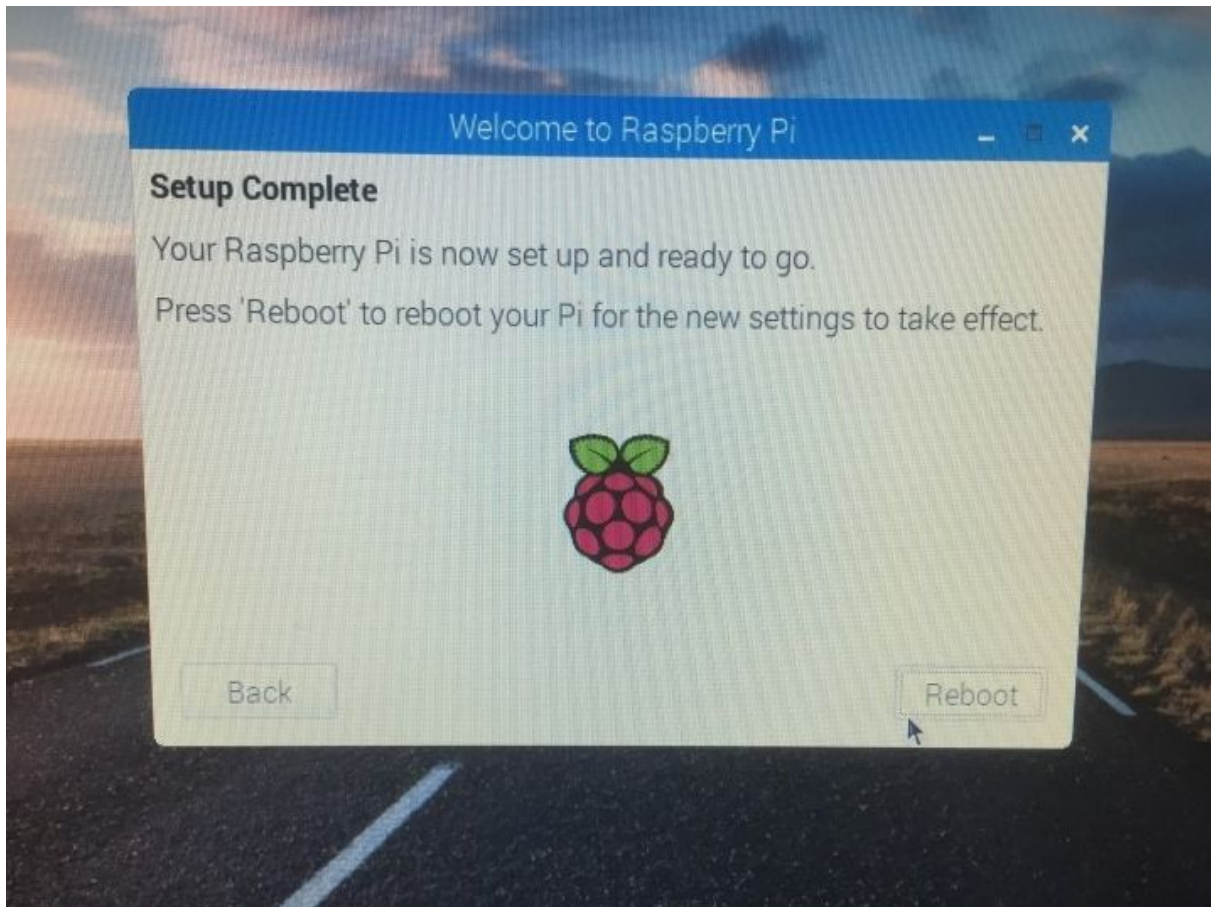




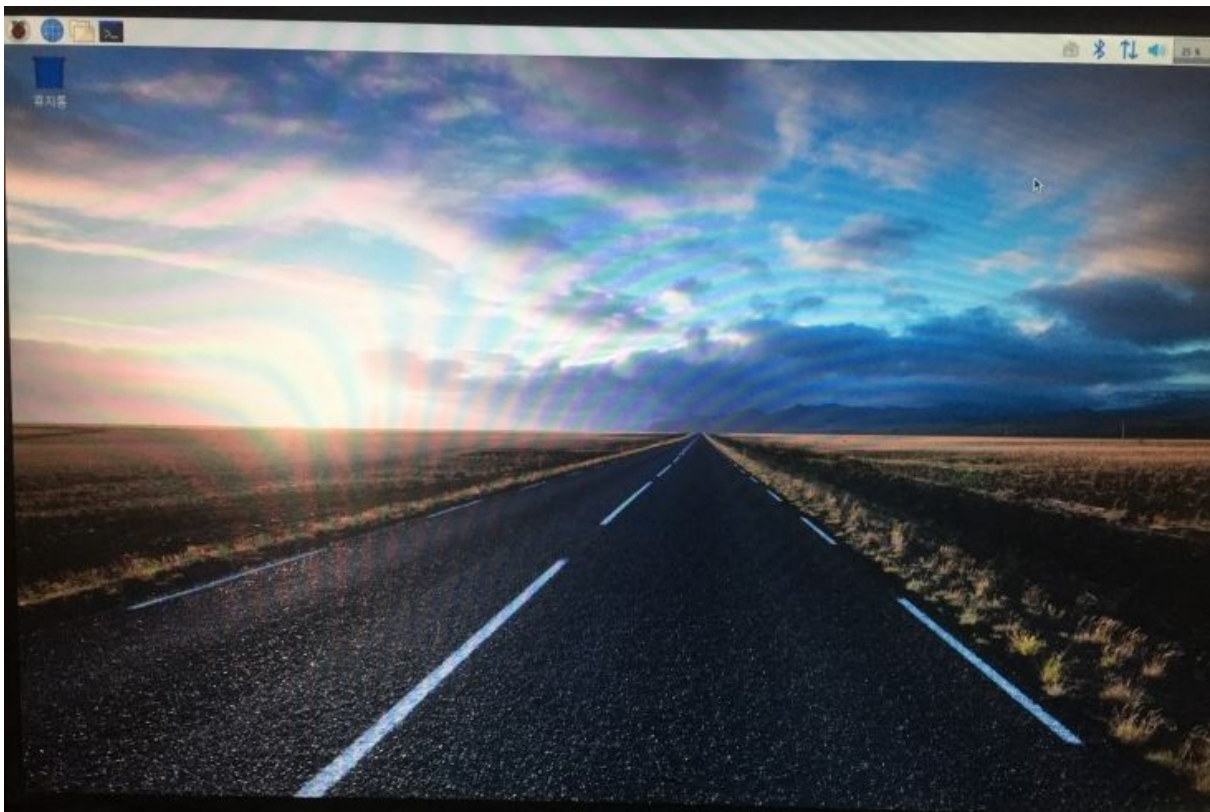
14. 설치가 완료되면 기본 설정들을 해줍니다.







15. 모든설정이 완료되면 재부팅 해줍니다.



16. 라즈베리파이를 사용하기 위한 모든 준비가 끝났습니다.

한글 글꼴과 이모지 설치

터미널 실행후

```
sudo apt-get --yes install fonts-nanum fonts-symbola
```

[엔터]

마이크 스피커 설정

메뉴>보조 프로그램>텍스트 에디터 실행

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

입력 한 후에 저장

위치는 /home/pi/

파일명은 “.asoundrc”

재부팅

화면 상단 오른쪽의 스피커 아이콘을 우클릭하여 나온 메뉴에서 Analog 선택

MagicMirror설치

MagicMirror설치

터미널 실행후

bash -c "\$(curl -sL

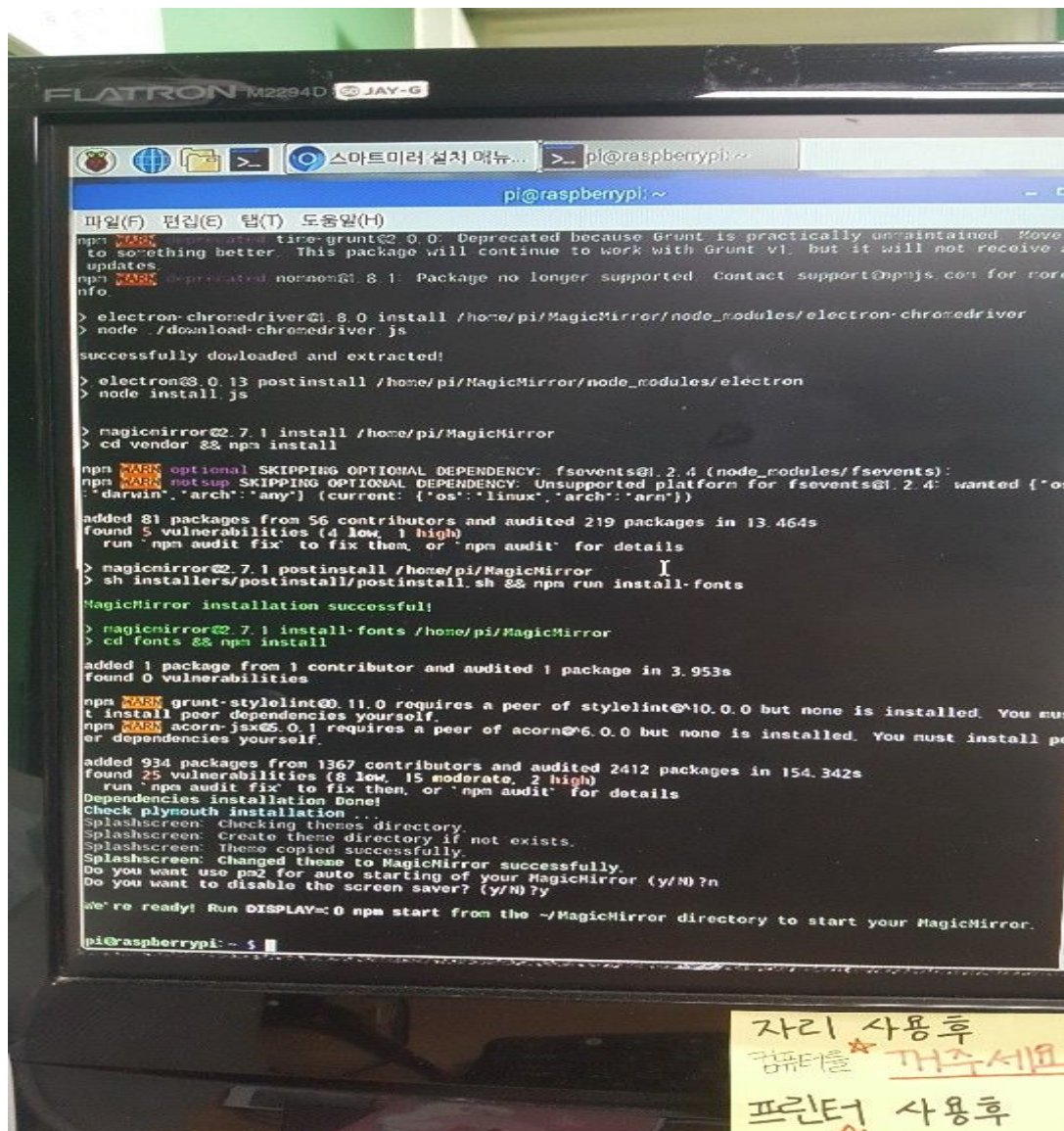
[https://raw.githubusercontent.com/MichMich/MagicMirror/master/installers/raspberry.sh\)](https://raw.githubusercontent.com/MichMich/MagicMirror/master/installers/raspberry.sh))"

[엔터]

마지막에 물어보는

Do you want use pm2 for auto starting of your MagicMirror (y/N)

에서 [N] [엔터]



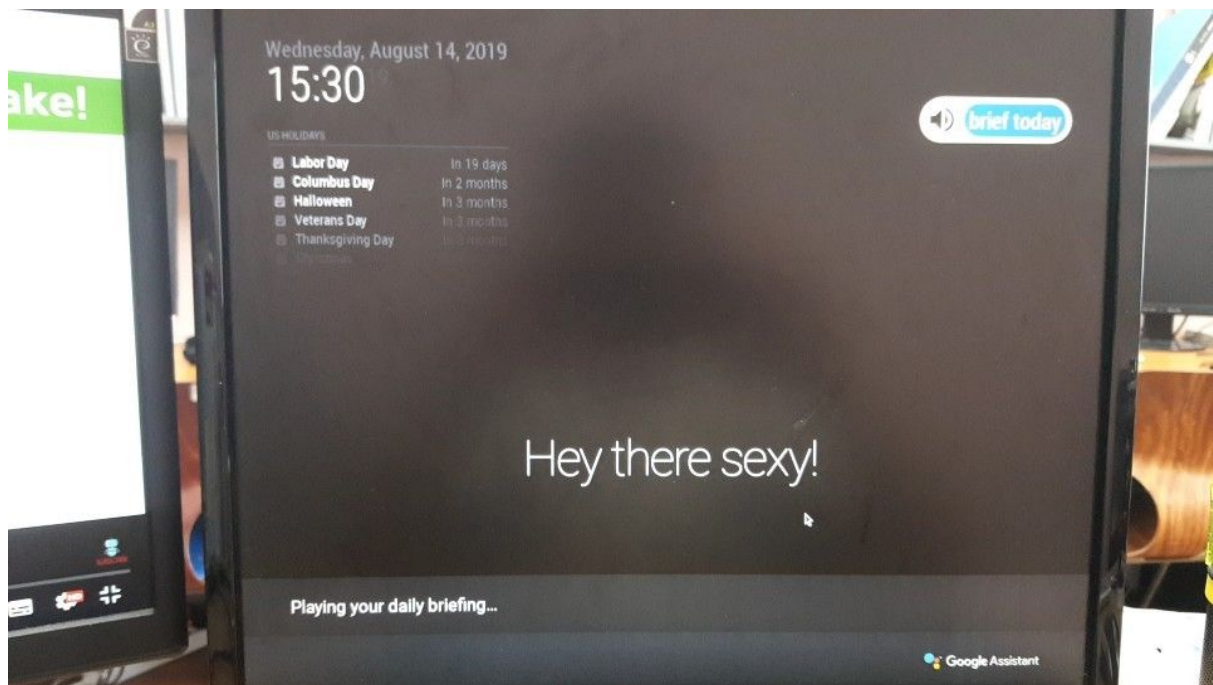
MaigcMirror 테스트

터미널에서
cd ~/MagicMirror/
[엔터]
npm start
[엔터]

MagicMirror가 잘 실행되는지 확인후
[Ctrl+Q]로 종료

실행이 안될경우

rm -rf ~/MagicMirror/
[엔터]
설치한 MagicMirror를 삭제한후
설치과정을 처음부터 다시



구글어시스턴트 모듈 설치

사용모듈: MMM-Hotword, MMM-NotificationTrigger, MMM-AssistantMk2

(제작자: Seongnoh Sean Yi) <https://github.com/eouia>

관련 프로그램 설치

터미널에서

```
sudo apt-get --yes install libmagic-dev libatlas-base-dev sox libsox-fmt-all libasound2-dev
```

[엔터]

모듈 다운로드

터미널에서

```
cd ~/MagicMirror/modules/
```

[엔터]

```
git clone https://github.com/eouia/MMM-Hotword.git
```

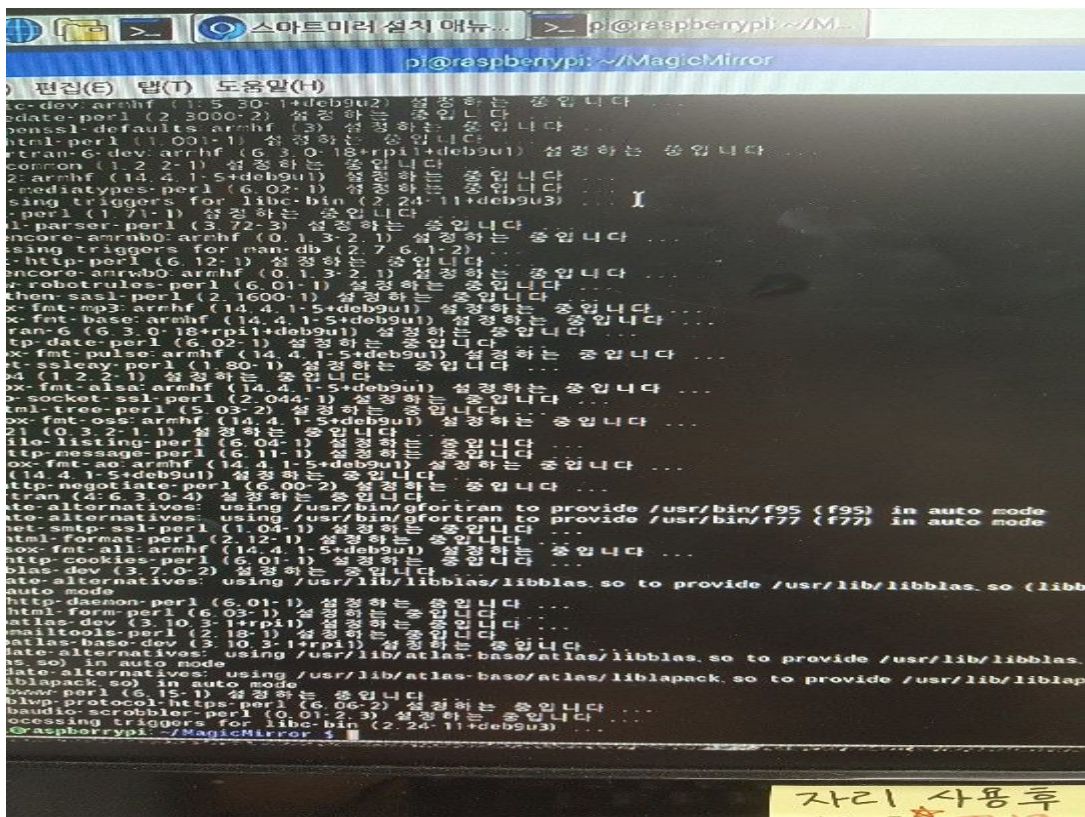
[엔터]

```
git clone https://github.com/eouia/MMM-NotificationTrigger.git
```

[엔터]

```
git clone https://github.com/eouia/MMM-AssistantMk2.git
```

[엔터]



모듈 설치

MMM-Hotword 설치

터미널에서

```
cd ~/MagicMirror/modules/MMM-Hotword/
```

[엔터]

```
npm install
```

[엔터]

설치가 끝나고

```
cd ~/MagicMirror/modules/MMM-Hotword/node_modules/snowboy/
```

[엔터]

```
npm install --save-dev electron-rebuild
```

[엔터]

```
npm install nan
```

[엔터]

```
./node_modules/.bin/electron-rebuild
```

[엔터]

MMM-AssistantMk2 설치

터미널에서

```
cd ~/MagicMirror/modules/MMM-AssistantMk2/
```

[엔터]

```
npm install
```

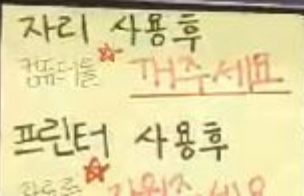
[엔터]

구글 어시스턴트 모듈 설정

다음 단계는 설치된 [구글 어시스턴트](#) 를 동작시키기 위해서 [구글 백엔드](#) 서버의 설정을 해보도록 하겠습니다. 이작업을 하기 이전에 아래 접속되는 [구글클라우드플랫폼](#)에 사전에 회원가입이 되어 있어야 합니다.

[구글액션콘솔](#) Google Action Console 에 접속하여 새로운 프로젝트를 생성합니다.

검색창에서 [Google](#) Assistant API 를 검색하여 활성화(Enable)를 실행해줍니다. 그러면 생성된 프로젝트에서 [구글어시스턴트](#) 기능을 사용가능한 상태로 됩니다.



config파일 설정

탐색기를 열어서 config.json을 /home/pi/MagicMirror/config/ 에 덮어쓰기

currentweather 설정

날씨 지역설정

그리고 config.json을 편집기로 실행해서 [Ctrl+F]로 찾기 도구를 써서 currentweather검색

그 아래의 location항목의 따옴표 안을 자신의 지역으로 편집

날씨 API설정

<https://openweathermap.org/> 에 접속

상단의 Sign up으로 가입

로그인 후 My Home에서 API keys으로 이동

페이지에 있는 키를 복사

날씨 지역설정에서 편집중이던 편집기로 와서

appid 항목의 따옴표안에 붙여넣기

weatherforecast 설정

config.json을 편집기로 실행해서 [Ctrl+F]로 찾기 도구를 써서 weatherforecast 검색

그 아래의 location항목의 따옴표 안을 자신의 지역으로 편집

appid 부분을 currentweather 에서 쓴거랑 동일하게 지정

편집파일저장

[Ctrl+S]로 저장

MagicMirror작동확인

터미널을 열고
cd ~/MagicMirror/
[엔터]
npm start
[엔터]

완성.

화면 계속 켜지게 하기

터미널을 열고
sudo apt-get --yes install xscreensaver

진행이 끝나고 나면 재부팅

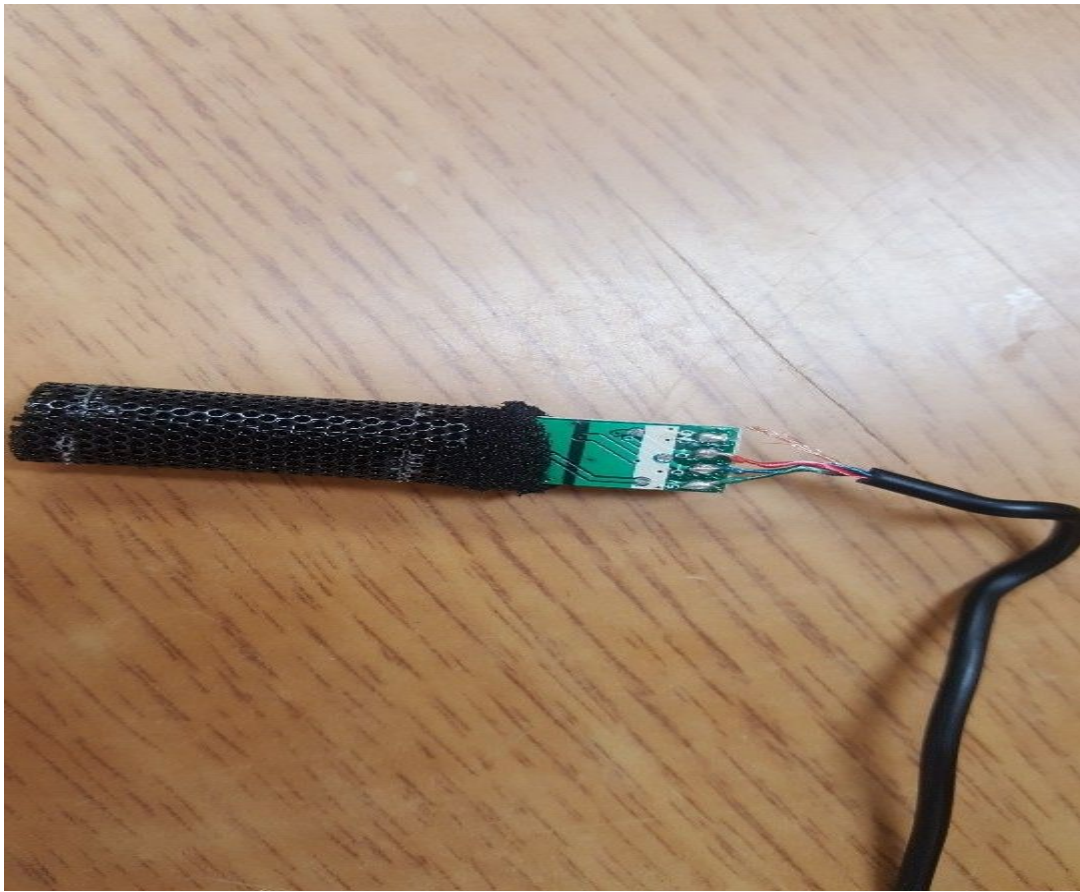
메뉴>기본 설정>화면 보호기

모드에서 Disable Screen Saver 선택

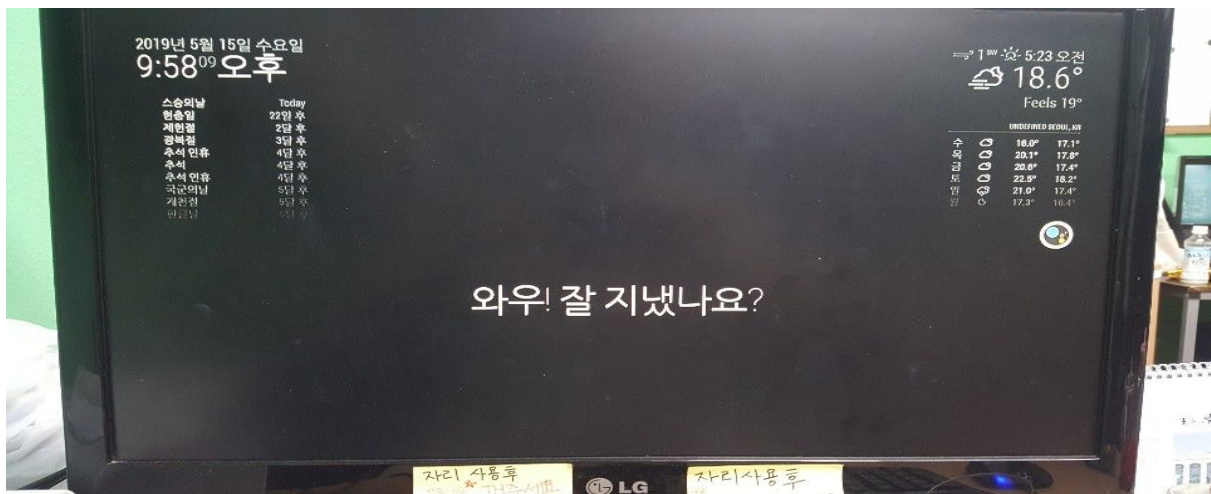
MagicMirror자동 실행 설정 (최종 테스트후 마지막)

터미널에서
sudo npm install -g pm2
[엔터]
pm2 startup
[엔터]
sudo env PATH=\$PATH:/usr/bin /usr/lib/node_modules/pm2/bin/pm2 startup systemd -u pi
--hp /home/pi
[엔터]
pm2 start ~/MagicMirror/installers/pm2_MagicMirror.json
[엔터]
pm2 save
[엔터]MagicMirror자동 실행 멈추는 방법
터미널에서 대소문자 구분하여서
pm2 stop MagicMirror
[엔터]

※스마트 미러 N:N 통신에 대한 피드백을 교수님께 받음 도서관에서 3권의 책(프로젝트로 배우는 라즈베리 파이, 리눅스, 라즈베리 파이 시작하기)를 참고하여 통신에 대해 공부



마이크 테스트중 마이크 고장.. 고장나기전까지는 테스트 성공





거울 본체 만드는 중

MagicMirror 버전 변경으로 인해 재 설치

1.라즈비안 패키지 시스템을 최신으로 업데이트 합니다.

```
sudo apt-get update
```

2.#MagicMirror2 설치

node.js #electron 으로 만들어진 MagicMirror2 프로그램을 설치합니다. MagicMirror2 는 #스마트미러 구현을 위한 기본 프레임워크 이며, 필요한 기능들을 모듈형태로 추가할 수 있는 구조 입니다.

스마트미러의 근간이 되는 프로그램 MagicMirror2 는 Node.js 기반의 Electron 으로 구현되어 있습니다. 라즈비안 운영체제에서 동작시키기 위해 Node.js 환경을 구성해줘야 합니다. node 명령어는 이미 설치가 되어 있어서, node 의 패키지 관리 시스템인 npm 을 최신버전으로 설치 합니다.

```
sudo apt-get install npm
```

```
sudo npm install -g npm@latest
```

오픈소스 MagicMirror2 에서 제공되는 스크립트로 설치 합니다. (최대 5분 소요)

```
bash -c "$(curl -sL
```

```
https://raw.githubusercontent.com/MichMich/MagicMirror/master/installers/raspberry
.sh)"
```

3.#MagicMirror2 에 핫워드와 구글어시스턴트 모듈 설치

node.js electron 으로 만들어진 MagicMirror2 프로그램을 설치하였고, 이번에는 MagicMirror2 에 음성인식과 말하기 기능을 넣기 위해 #핫워드 모듈과 구글어시스턴트 모듈을 설치합니다.

이전 단계에서 스마트미러의 근간이 되는 프로그램 MagicMirror2 설치완료 했습니다. 이번 단계에서는 음성인식 말하기 기능을 위한 추가 모듈을 설치 합니다. 우선, 모듈 설치전에 필요한 종속 프로그램과 라이브러리를 미리 설치 합니다.

```
sudo apt-get install libmagic-dev libatlas-base-dev sox libsox-fmt-all mpg321
libasound2-dev
```

modules 디렉토리 이동후, 핫워드 MMM-Hotword 설치를 수행합니다. 터미널 창을 열고, 다음의 명령어를 차근차근히 입력하시면 됩니다.

4. 기존 MMM-Hotword는 Submodule 들을 하나하나 직접 설치하고 빌드해야 했지만, 2.0.0버전 부터는 설치 스크립트에 통합되었습니다!

```
cd ~/MagicMirror/modules/  
git clone https://github.com/eouia/MMM-Hotword.git  
cd MMM-Hotword  
chmod +x ./installer/install.sh  
./installer/install.sh
```

기존 튜토리얼에서 가장 실패가 많았던 Hotword 가 설치스크립트 하나로 금방 설치 되었습니다. 다들 잘 되셨는지요? 그럼 다시 modules 디렉토리 이동후, [구글어시스턴트 MMM-AssistantMk2](#) 설치를 수행합니다. (20~30분 소요)

5.#MagicMirror2 에 핫워드와 구글어시스턴트 모듈 설치

```
cd ~/MagicMirror/modules/  
git clone https://github.com/eouia/MMM-AssistantMk2  
cd MMM-AssistantMk2  
npm install  
cd scripts  
chmod +x *.sh  
cd ..  
npm install --save-dev electron-rebuild  
./node_modules/.bin/electron-rebuild
```

6.#구글어시스턴트 설정

구글어시스턴트 사용을 위해 백엔드 서버의 설정을 합니다.

이전버전과 같은 방식으로 진행!

다. 지금 활성화하여 Google Cloud 제품에 쓸 수 있는 \$300의 크레딧을 받아 보세요. [자세히 알아보기](#)

form

scnujm

Q

사용자 인증 정보

사용자 인증 정보

OAuth 동의 화면

도메인 확인

사용자 인증 정보 만들기

삭제

OAuth 클라이언트

API 및 서비스의 사용자 인증 정보에서 언제든지 클라이언트 ID와 보안 비밀에 액세스할 수 있습니다.

- ① OAuth 동의 화면을 게시할 때까지 OAuth에서 민감한 범위 로그인에 100개로 제한됩니다. 확인 절차가 필요할 수 있으며 며칠 정도 걸립니다.

클라이언트 ID입니다

747684463053-p1bhgkhj71kpm3u3ecocghucct0ish12.apps.googleusercontent.com

클라이언트 보안 비밀입니다.

TKVH41_EbW6DPGGx3VgY2xEI

확인

<input type="checkbox"/> 기타 클라이언트 3	2019. 5. 15.	기타	747684463053-p1bhgkhj71kpm3u3ecocghucct0ish12.apps.googleusercontent.com
<input type="checkbox"/> 기타 클라이언트 2	2019. 5. 15.	기타	747684463053-p2nkfq196v7v9jrlksmjnq39b1d8kgj0.apps.googleusercontent.com
<input type="checkbox"/> New Actions on Google App	2019. 5. 15.	웹 애플리케이션	747684463053-aca6910kkfo48qu43nl3irfe135j2sac7.apps.googleusercontent.com
<input type="checkbox"/> Web client (auto created by Google)	2019. 5. 15.	웹 애플리케이션	747684463053-q1kjo4r8pu8olaf01ruff19q39c9h0n.apps.googleusercontent.com

자리사용 후
★ 컴퓨터를 ★
꺼주세요



거울까지 부착완료

6.#구글어시스턴트 모듈 설정값 수정

[스마트미러](#)의 근간이 되는 프로그램 [MagicMirror2](#) 설치완료 하고, [음성인식 말하기](#) 기능을 위한 추가 모듈을 설치하고 [구글클라우드플랫폼](#) 서비스에 연결을 했습니다. 설치된 모듈을 구성하기 위해서 설정파일인 config/config.js 에 내용을 추가해야합니다.

아래 추가될 코드를 넣었습니다. 추가되는 내용은 [하위드 구글어시스턴트](#) 동작에 대한 내용입니다.

```
var config = {  
  address: "localhost",  
  port: 8080,  
  ipWhitelist: ["127.0.0.1", "::ffff:127.0.0.1", ":::1"],  
  language: "ko",  
  timeFormat: 12,  
  units: "metric",  
  
  modules: [  
    {  
      module: "alert",  
    },  
    {  
      module: "updatenotification",  
      position: "top_bar"  
    },  
    {  
      module: "clock",  
      position: "top_left",  
      config: {  
        dateFormat: "LL dddd",  
      }  
    }  
  ]  
}
```

```

    },
    {
      module: "calendar",
      position: "top_left",
      config: {
        urgency: 21,
        fetchInterval: 3600000,
        calendars: [{
          url:
'https://calendar.google.com/calendar/ical/ko.south_korea%23holiday%40g
roup.v.calendar.google.com/public/basic.ics',
          symbol: '대한민국 공휴일',
        }, ]
      }
    },
    {
      module: "currentweather",
      position: "top_right",
      config: {
        location: "Seoul, KR", //https://openweathermap.org/ 에서 지역을
찾으세요.
        locationID: "",
        appid: "" //openweathermap.org의 API key를 입력하세요.
      }
    },
    {
      module: "weatherforecast",
      position: "top_right",
      config: {
        location: "Seoul, KR", //https://openweathermap.org/ 에서 지역을
찾으세요.
        locationID: "",

```

```

    appid: "" //openweathermap.org의 API key를 입력하세요.
  }
},
{
  module: "compliments",
  position: "lower_third",
  config: {
    compliments: {
      anytime: [ //아무때나
        "오늘도 좋은 하루!",
      ],
      morning: [ //아침
        "좋은 아침!",
        "힘찬 아침!",
        "잘 잤나요?",
      ],
      afternoon: [ //오후
        "안녕하세요!",
        "멋져요!",
        "잘 지내고 있나요?",
      ],
      evening: [ //저녁
        "와우! 잘 지냈나요?",
        "멋져 보이네요!",
        "반가워요!",
      ],
      day_sunny: [//맑은 낮
      ],
      day_cloudy: [//흐린 낮

```



```
    ],
    cloudy: [//흐림

],
    cloudy_windy: [//흐리고 바람

],
    showers: [//소나기

],
    rain: [//비

],
    thunderstorm: [//천둥번개

],
    snow: [//눈

],
    fog: [//안개

],
    night_clear: [//맑은 밤

],
    night_cloudy: [//흐린 밤

],
    night_showers: [//소나기 밤

],
```

```
        night_rain: [//비오는 밤
```

```
        ],
```

```
        night_thunderstorm: [//천둥번개 밤
```

```
        ],
```

```
        night_snow: [//눈오는 밤
```

```
        ],
```

```
        night_alt_cloudy_windy: [//흐리고 바람부는 밤
```

```
        ],
```

```
    }  
  }  
}
```

```
  }
```

```
},
```

```
        coordinates: {{
module: "MMM-AssistantMk2",
position: "top_right",
config: {
  record: {
    recordProgram : "arecord",
    device          : "plughw:1",
  },
  notifications: {
    ASSISTANT_ACTIVATED: "HOTWORD_PAUSE",
    ASSISTANT_DEACTIVATED: "HOTWORD_RESUME",
  },
  deviceLocation: {

    latitude: 37.57,
    longitude: 126.98

  }
},
profiles: {
  "default": {
    lang: "ko-KR"
  }
}
},
{
module: "MMM-Hotword",
config: {
  snowboy: [
    {
```



```

        hotwords: "smartmirror",
        file: "resources/models/smart_mirror.umd1",
        sensitivity: '1.0',
    },
    {
        hotwords: "snowboy",
        file: "resources/models/snowboy.umd1",
        sensitivity: '1.0',
    },
    {
        file: 'resources/models/jarvis.umd1',
        sensitivity: '0.8,0.80',
        hotwords: ['jarvis', 'jarvis']
    }
],
record: {
    recordProgram : "arecord",
    device        : "plughw:1",
},
autostart:true,
onDetected: {
    notification: function (payload) {
        return "ASSISTANT_ACTIVATE"
    },
    payload: function (payload){
        return {
            profile: payload.hotword
        }
    }
},
},
},


```



```
    },  
  ]  
};
```

```
/****** DO NOT EDIT THE LINE BELOW *****/  
if (typeof module !== "undefined") {  
  module.exports = config;  
}
```



최종 테스트 완료



[Dashboard](#)
[Docs](#)
[Forum](#)


[지만장](#)



Hotword Library

Hide Abusive Words ☐

All Models ▾ Languages Filter ▾ Sort by ☒ Recently added ☐ Most popular

Hotword	Language	Public model	Personal model
Kompliment	German	<input type="text"/> 1/2000 samples ⓘ	Record and Download 

«
<
1
>
»



지만장

Activites

- "smart mirror" now has a general model

See more

MMM-Hotword 의 서브 모듈인 스노우보이는 MMM-AsistansMK2를 호출하기 위한 예약어를 만들거나 사용할 수 있는 사이트를 제공하고 있다.

New Hotword

Hotword details

Record my voice

Test the model

Hotword Name

Language

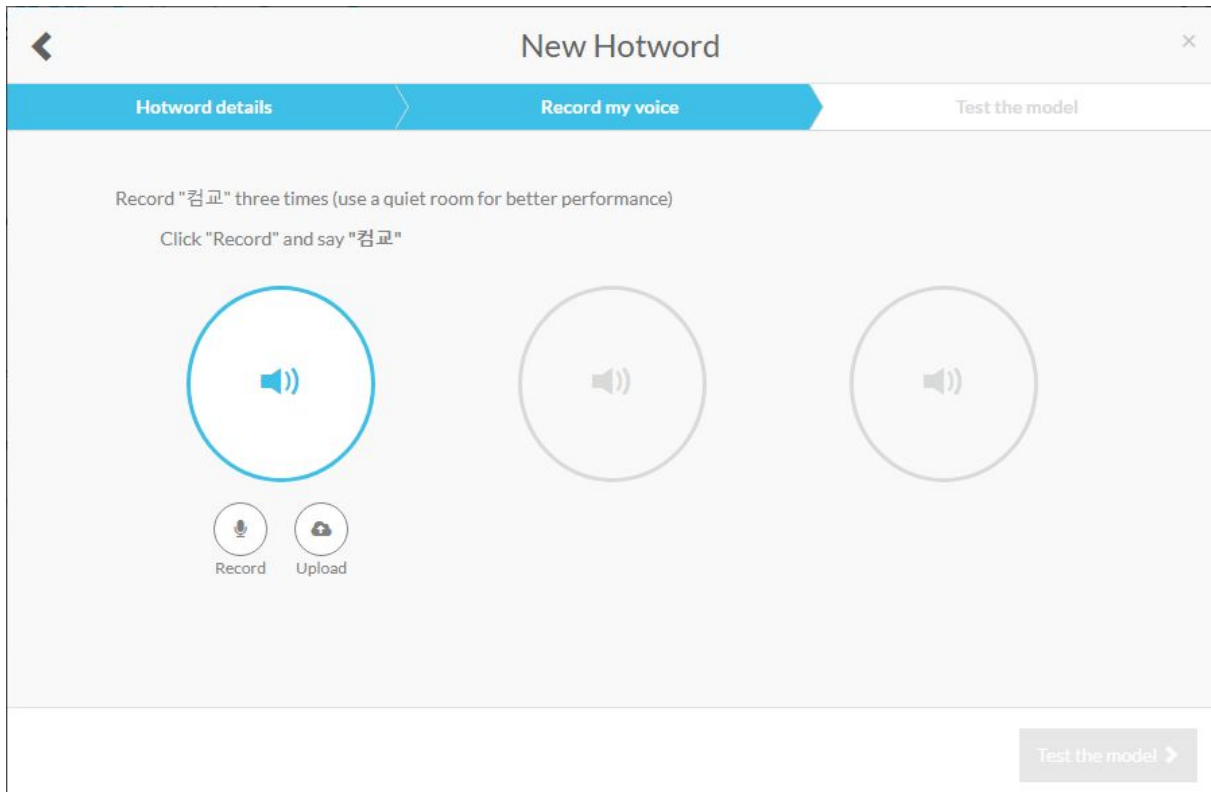
Korean ▾

Personal Comment (Optional)

2019 소프트웨어 전시회에서 사용하기 위한 핫키워드

This comment will be visible only to you and will make hotword easier to find

Record my voice



목소리를 녹음하여 호출어를 만든다.

녹음후 완성된 umdl 파일은 Config.js에 삽입한다.

```

/* Magic Mirror Config Sample
*
* By Michael Teeuw http://michaelteww.nl
* MIT Licensed.
*
* For more information how you can configure this file
* See https://github.com/MichMich/MagicMirror#configuration
*
*/

var config = {
  address: "localhost", // Address to listen on, can be:
                        // - "localhost", "127.0.0.1", "::1" to listen on loopback interface
                        // - another specific IPv4/6 to listen on a specific interface
                        // - "", "0.0.0.0", "::" to listen on any interface
                        // Default, when address config is left out, is "localhost"
  port: 8080,
  ipWhitelist: ["127.0.0.1", "::ffff:127.0.0.1", "::1"], // Set [] to allow all IP addresses
                                                         // or add a specific IPv4 of 192.168.1.5 :
                                                         // ["127.0.0.1", "::ffff:127.0.0.1", "::1",
::ffff:192.168.1.5"],
                                                         // or IPv4 range of 192.168.3.0 --> 192.168.3.15
use CIDR format :
                                                         // ["127.0.0.1", "::ffff:127.0.0.1", "::1",
::ffff:192.168.3.0/28"],

  language: "ko",
  timeFormat: 24,
  units: "metric",

  modules: [
    {
      module: "alert",
    },
    {
      module: "updatenotification",
      position: "top_bar"
    },
    {
      module: "clock",
      position: "top_left"
    },
    {
      module: "calendar",
      header: "대한민국 공휴일",
      position: "top_left",
    }
  ]
};

```

```

config: {
  calendars: [
    {
      symbol: "calendar-check",
      url:
"https://calendar.google.com/calendar/ical/ko.south_korea%23holiday%40group.v.calendar.
google.com/public/basic.ics"
    }
  ]
},
{
  module: "compliments",
  position: "lower_third",
  config: {
    compliments: {
      anytime: [ //아무때나
        "2019 소프트웨어 전시회"
      ],
      morning: [ //아침
        "좋은 아침!",
        "힘찬 아침!",
        "오신걸 환영합니다!"
      ],
      afternoon: [ //오후
        "안녕하세요!",
        "멋져요!",
        "오신걸 환영합니다!"
      ],
      evening: [ //저녁
        "와우! 잘 지냈나요?",
        "멋져보이네요!",
        "반가워요!"
      ],
      day_sunny: [//맑은 낮
      ],
      day_cloudy: [//흐린 낮
      ],
      cloudy: [//흐림
      ],
      cloudy_windy: [//흐리고 바람
      ],
      showers: [//소나기

```

```

    ],
    rain: [//비

    ],
    thunderstorm: [//천둥번개

    ],
    snow: [//눈

    ],
    fog: [//안개

    ],
    night_clear: [//맑은 밤

    ],
    night_cloudy: [//흐린 밤

    ],
    night_showers: [//소나기 밤

    ],
    night_rain: [//비오는 밤

    ],
    night_thunderstorm: [//천둥번개 밤

    ],
    night_snow: [//눈오는 밤

    ],
    night_alt_cloudy_windy: [//흐리고 바람부는 밤

    ],
    }
}
},
{

```

```

    module: "currentweather",
    position: "top_right",
    config: {
        location: "Gwangju, KR",
        locationID: "", //ID from

```

<http://bulk.openweathermap.org/sample/city.list.json.gz>; unzip the gz file and find your city
 appid: "6050bc695807684cf9594b35887a5464"


```

    }
  },
  {
    module: "weatherforecast",
    position: "top_right",
    header: "Weather Forecast",
    config: {
      location: "Gwangju, KR",
      locationID: "", //ID from
      http://bulk.openweathermap.org/sample/city.list.json.gz; unzip the gz file and find your city
      appid: "6050bc695807684cf9594b35887a5464"
    }
  },
  {
    module: "newsfeed",
    position: "bottom_bar",
    config: {
      feeds: [
        {
          title: "New York Times",
          url:
            "http://www.nytimes.com/services/xml/rss/nyt/HomePage.xml"
        }
      ],
      showSourceTitle: true,
      showPublishDate: true,
      broadcastNewsFeeds: true,
      broadcastNewsUpdates: true
    }
  },
  {
    module: "MMM-Hotword",
    position: "top_right",
    config: {
      chimeOnFinish: null,
      mic: {
        recordProgram: "arecord",
        device: "plughw:1"
      },
      models: [
        {
          hotwords : "smart_mirror",
          file      : "smart_mirror.umdl",
          sensitivity : "0.5",
        },
      ],
    }
  },

```

```

commands: {
  "smart_mirror": {
    notificationExec: {
      notification: "ASSISTANT_ACTIVATE",
      payload: (detected, afterRecord) => {
        return {profile:"default"}
      }
    },
    restart:false,
    afterRecordLimit:0
  }
}

},

{
  chimeOnFinish: null,
  mic: {
    recordProgram: "arecord",
    device: "plughw:1"
  },
  models: [
    {
      hotwords : "컴교",
      file      : "컴교.pmdl",
      sensitivity : "0.5",
    },
  ],
  commands: {
    "smart_mirror": {
      notificationExec: {
        notification: "ASSISTANT_ACTIVATE",
        payload: (detected, afterRecord) => {
          return {profile:"default"}
        }
      },
      restart:false,
      afterRecordLimit:0
    }
  }
}

},
{
  module: "MMM-AssistantMk2",
  position: "top_right",
  config: {
    deviceLocation: {

```

```

        coordinates: {
            latitude: 34.969548, // -90.0 - +90.0
            longitude: 127.478610, // -180.0 - +180.0
        },
    },
    record: {
        recordProgram : "arecord",
        device      : "plughw:1",
    },
    notifications: {
        ASSISTANT_ACTIVATED:
"HOTWORD_PAUSE",
        ASSISTANT_DEACTIVATED:
"HOTWORD_RESUME",
    },
    useWelcomeMessage: "Hellow, jiman",
    profiles: {
        "default" : {
            lang: "ko-KR"
        }
    },
    },
    },
    ],
};

```

```

/***** DO NOT EDIT THE LINE BELOW *****/
if (typeof module !== "undefined") {module.exports = config;}

```