



Raspberry Pi Project

온습도 & 미세먼지 모니터링 시스템

손원용, 옥진해, 정지희

01



개발목표

02



부품 및 개발환경

03



결선도 & code

04

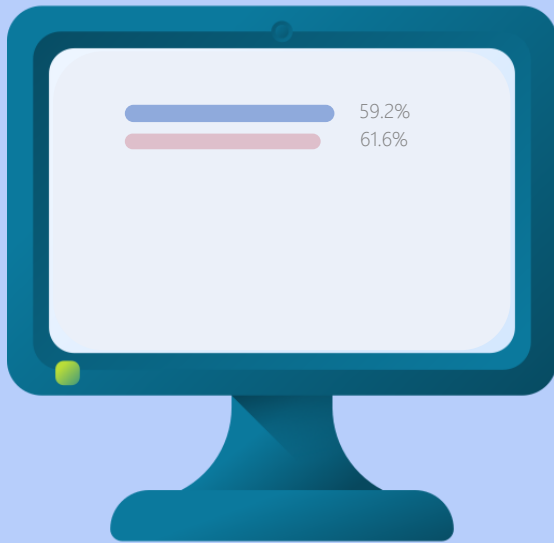


Demo

01. 개발목표



01 개발 목표



● 실시간 모니터링

온습도센서, 미세먼지센서, 유해가스 센서를 이용하여
실시간 모니터링 기능

02. 부품 및 개발환경



02 부품 및 개발환경

부품

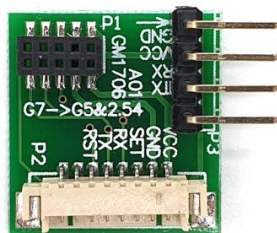
- 레이저 방식 미세먼지 센서 [PMS7003M]
- PMS7003 어댑터 보드 [SY-PMS7003ab]
- 유해가스 센서 [MQ-135]
- MCP3008
- 온습도센서[DHT11]
- 라즈베리파이 7인치 터치스크린

개발환경



02 부품 및 개발환경

레이저 방식 미세먼지 센서 [PMS7003M] / PMS7003 어댑터 보드 [SY-PMS7003ab]

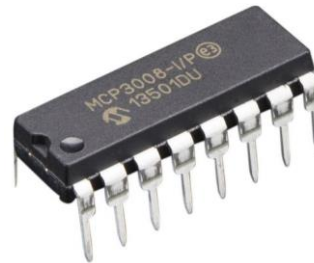
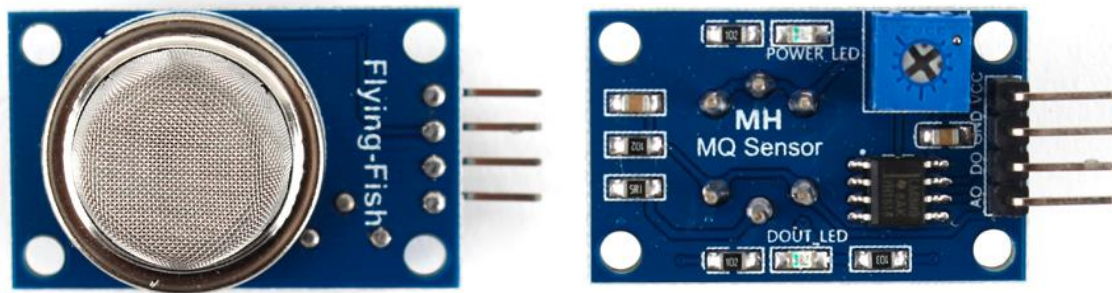


일반형 미세먼지 센서로 대기 중 입자 농도를 감지하고
디지털 인터페이스 형태로 출력

- 입자 검출 최소 크기 : $0.3 \mu\text{m}$
- 감지 대상 : PM2.5/PM10, $0 \sim 999 \mu\text{g}/\text{m}^3$

02 부품 및 개발환경

유해가스/공기질 센서 모듈 [SZH-SSBH-038] / MCP3008

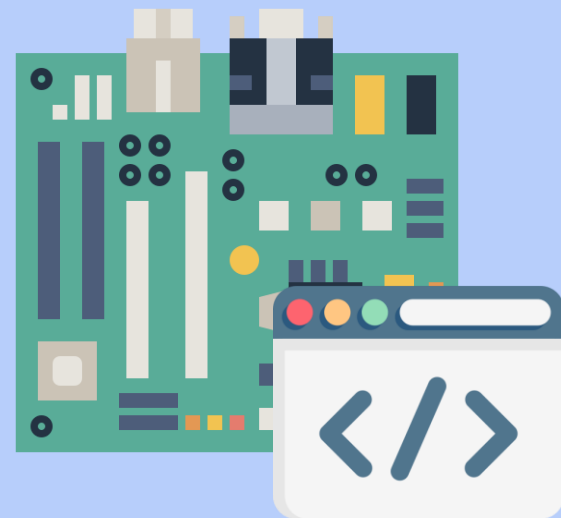


CH0	□	1	MCP3008	16	□	V _{DD}
CH1	□	2		15	□	V _{REF}
CH2	□	3		14	□	AGND
CH3	□	4		13	□	CLK
CH4	□	5		12	□	D _{OUT}
CH5	□	6		11	□	D _{IN}
CH6	□	7		10	□	CS/SHDN
CH7	□	8		9	□	DGND

- CO₂, NH₃, smoke, Benzene, alcohol 등을 측정
- 디지털 출력과 아날로그 출력 가능
- 가변저항으로 감도조절 가능

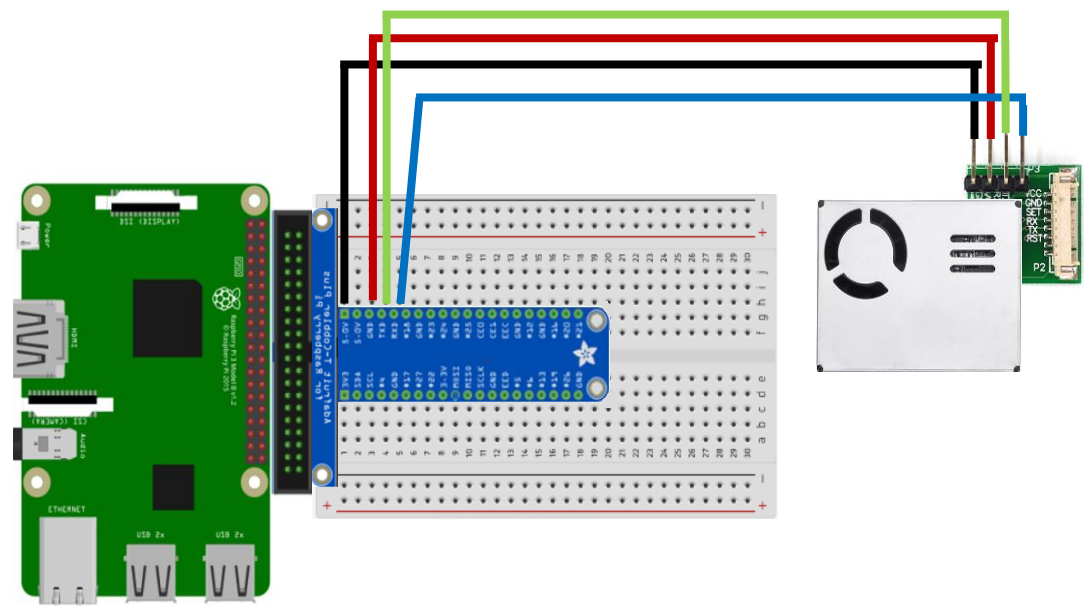
- ADC(Analog-to-digital converter)
- 연속적인 analog 신호를 digital 신호로 변환하는 칩(I/C)

03. 결선도 & code



03 결선도 & code

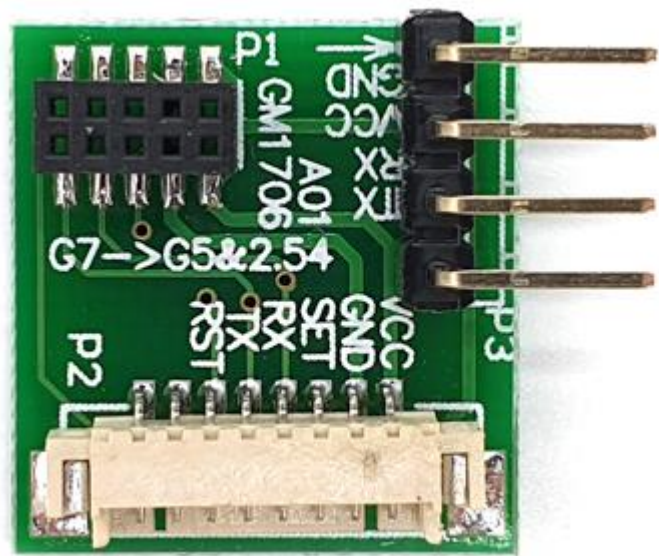
미세먼지 결선도



Rpi	PMS7003M
5V	VCC
GND	GND
Tx	Rx
Rx	Tx

03 결선도 & code

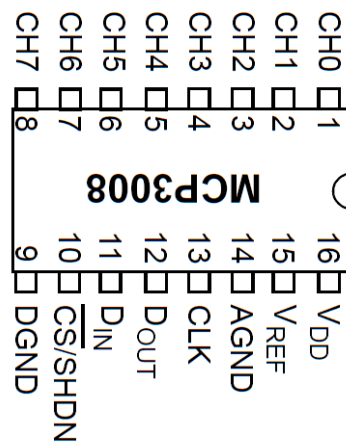
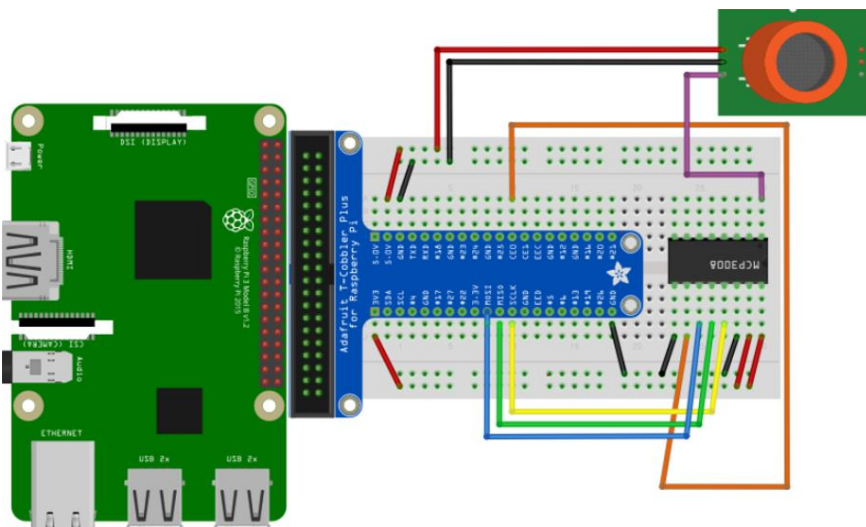
미세먼지 결선도



Rpi	PMS7003M
5V	VCC
GND	GND
Tx	Rx
Rx	Tx

03 결선도 & code

유해가스/공기질 센서 결선도



Rpi	MQ-135	MCP3008
5V	VCC	Vdd, Vref
GND	GND	AGND, DGND
	AO	CH0
SPISCLK		CLK
SPIMISO		Dout
SPIMOSI		Din
SPICEO		CS/SHDN

Rpi	MQ-135	MCP3008
5V	VCC	Vdd, Vref
GND	GND	AGND, DGND
	AO	CH0
SPISCLK		CLK
SPIMISO		Dout
SPIMOSI		Din
SPICEO		CS/SHDN

03 결선도 & code

```
import Adafruit_DHT
from time import *
import sys
from PyQt5.QtWidgets import *
from PyQt5 import uic
from PyQt5 import QtCore, QtGui, QtWidgets
from PyQt5.QtCore import *
from PyQt5.QtGui import *
from PyQt5.QtWidgets import *
from PyQt5.QtCore import QTimer, QTime
from threading import Timer
import datetime
import serial
from PMS7003 import PMS7003
from mq import *
```

```
form_class = uic.loadUiType("test_window.ui")[0]
pin = 4
sensor = Adafruit_DHT.DHT11
dust = PMS7003()
Speed = 9600
UART = '/dev/ttyAMA0'
SERIAL_PORT = UART
ser = serial.Serial(SERIAL_PORT, Speed, timeout = 1)
```

03 결선도 & code

```
def on_set_clicked(self):
    tem = self.t_on_set_spinbox_on.value()
    hum = self.h_on_set_spinbox_on.value()
    set_list1 = [tem, hum]
    return set_list1

def off_set_clicked(self):
    tem1 = self.t_off_set_spinbox_off.value()
    hum1 = self.h_off_set_spinbox_off.value()
    set_list2 = [tem1, hum1]
    return set_list2
```

```
def displayTime(self):
    h, t = Adafruit_DHT.read_retry(sensor, pin)
    buffer = ser.read(1024)
    data = dust.unpack_data(buffer)
    d = (data[dust.DUST_PM2_5_ATM])
    now = datetime.datetime.now()
    nowDate = now.strftime('%Y-%m-%d %H:%M:%S')
    lpg = round((perc["GAS_LPG"] / 10000, 7)
    mq = MQ()
    perc = mq.MQPercentage()

    self.time.setText(str(nowDate))
    self.h_label.setText(str(h))
    self.t_label.setText(str(t))
    self.d_label.setText(str(d))
    self.g_label.setText(str(perc["GAS_LPG"]))
```

03 결선도 & code

```
Python 3.7.3 Shell
File Edit Shell Debug Options Window Help
DATA read success
=====
Header : B M          | Frame length : 28
PM 1.0 (CF=1) : 24     | PM 1.0 : 23
PM 2.5 (CF=1) : 40     | PM 2.5 : 36
PM 10.0 (CF=1) : 42    | PM 10.0 : 42
0.3um in 0.1L of air : 4149
0.5um in 0.1L of air : 1211
1.0um in 0.1L of air : 237
2.5um in 0.1L of air : 34
5.0um in 0.1L of air : 4
10.0um in 0.1L of air : 0
Reserved F : 152 | Reserved B : 0
CHKSUM : 1065 | read CHKSUM : 1065 | CHKSUM result : True
=====
DATA read success
=====
Header : B M          | Frame length : 28
PM 1.0 (CF=1) : 25     | PM 1.0 : 23
PM 2.5 (CF=1) : 43     | PM 2.5 : 38
PM 10.0 (CF=1) : 46    | PM 10.0 : 46
0.3um in 0.1L of air : 4257
0.5um in 0.1L of air : 1247
1.0um in 0.1L of air : 250
2.5um in 0.1L of air : 40
5.0um in 0.1L of air : 6
10.0um in 0.1L of air : 0
Reserved F : 152 | Reserved B : 0
CHKSUM : 1244 | read CHKSUM : 1244 | CHKSUM result : True
=====
Header err
DATA read fail...
DATA read success
=====
Header : B M          | Frame length : 28
PM 1.0 (CF=1) : 23     | PM 1.0 : 22
PM 2.5 (CF=1) : 41     | PM 2.5 : 37
PM 10.0 (CF=1) : 44    | PM 10.0 : 44
0.3um in 0.1L of air : 3966
0.5um in 0.1L of air : 1182
=====
Ln: 262 Col: 4
```

```
Python 3.7.3 Shell
File Edit Shell Debug Options Window Help
===== RESTART: /home/pi/project/example.py =====
Press CTRL+C to abort.
Calibrating...
Calibration is done...

Ro=-0.027214 kohm
LPG: 0.0139967 ppm, CO: 0.011422 ppm, Smoke: 0.0389112 ppm
LPG: 0.0150702 ppm, CO: 0.0126505 ppm, Smoke: 0.0421071 ppm
LPG: 0.0139967 ppm, CO: 0.011422 ppm, Smoke: 0.0389112 ppm
LPG: 0.0150702 ppm, CO: 0.0126505 ppm, Smoke: 0.0421071 ppm
LPG: 0.013014 ppm, CO: 0.0103286 ppm, Smoke: 0.0360001 ppm
LPG: 0.0139967 ppm, CO: 0.011422 ppm, Smoke: 0.0389112 ppm
LPG: 0.0112864 ppm, CO: 0.00848268 ppm, Smoke: 0.0309193 ppm
LPG: 0.0139967 ppm, CO: 0.011422 ppm, Smoke: 0.0389112 ppm
LPG: 0.0139967 ppm, CO: 0.011422 ppm, Smoke: 0.0389112 ppm
LPG: 0.0121132 ppm, CO: 0.00935353 ppm, Smoke: 0.0333447 ppm
LPG: 0.0159467 ppm, CO: 0.0136787 ppm, Smoke: 0.0447281 ppm
LPG: 0.0139967 ppm, CO: 0.011422 ppm, Smoke: 0.0389112 ppm
LPG: 0.0121132 ppm, CO: 0.00935353 ppm, Smoke: 0.0333447 ppm
LPG: 0.0150702 ppm, CO: 0.0126505 ppm, Smoke: 0.0421071 ppm
LPG: 0.0150702 ppm, CO: 0.0126505 ppm, Smoke: 0.0421071 ppm
LPG: 0.0139967 ppm, CO: 0.011422 ppm, Smoke: 0.0389112 ppm
LPG: 0.013014 ppm, CO: 0.0103286 ppm, Smoke: 0.0360001 ppm
LPG: 0.00950801 ppm, CO: 0.0066926 ppm, Smoke: 0.0257446 ppm
LPG: 0.0101796 ppm, CO: 0.00735474 ppm, Smoke: 0.0276916 ppm
LPG: 0.0139967 ppm, CO: 0.011422 ppm, Smoke: 0.0389112 ppm
LPG: 0.0105266 ppm, CO: 0.00770361 ppm, Smoke: 0.0287012 ppm
LPG: 0.013014 ppm, CO: 0.0103286 ppm, Smoke: 0.0360001 ppm
LPG: 0.0121132 ppm, CO: 0.00935353 ppm, Smoke: 0.0333447 ppm
LPG: 0.0109091 ppm, CO: 0.00809317 ppm, Smoke: 0.0298164 ppm
LPG: 0.0112864 ppm, CO: 0.00848268 ppm, Smoke: 0.0309193 ppm
LPG: 0.00780465 ppm, CO: 0.00509421 ppm, Smoke: 0.0208499 ppm
LPG: 0.0121132 ppm, CO: 0.00935353 ppm, Smoke: 0.0333447 ppm
LPG: 0.0121132 ppm, CO: 0.00935353 ppm, Smoke: 0.0333447 ppm
LPG: 0.0121132 ppm, CO: 0.00935353 ppm, Smoke: 0.0333447 ppm
LPG: 0.0112864 ppm, CO: 0.00848268 ppm, Smoke: 0.0309193 ppm
LPG: 0.0101796 ppm, CO: 0.00735474 ppm, Smoke: 0.0276916 ppm
LPG: 0.0105266 ppm, CO: 0.00770361 ppm, Smoke: 0.0287012 ppm
LPG: 0.0101796 ppm, CO: 0.00735474 ppm, Smoke: 0.0276916 ppm
LPG: 0.0112864 ppm, CO: 0.00848268 ppm, Smoke: 0.0309193 ppm
Ln: 372 Col: 0
```


03 결선도 & code

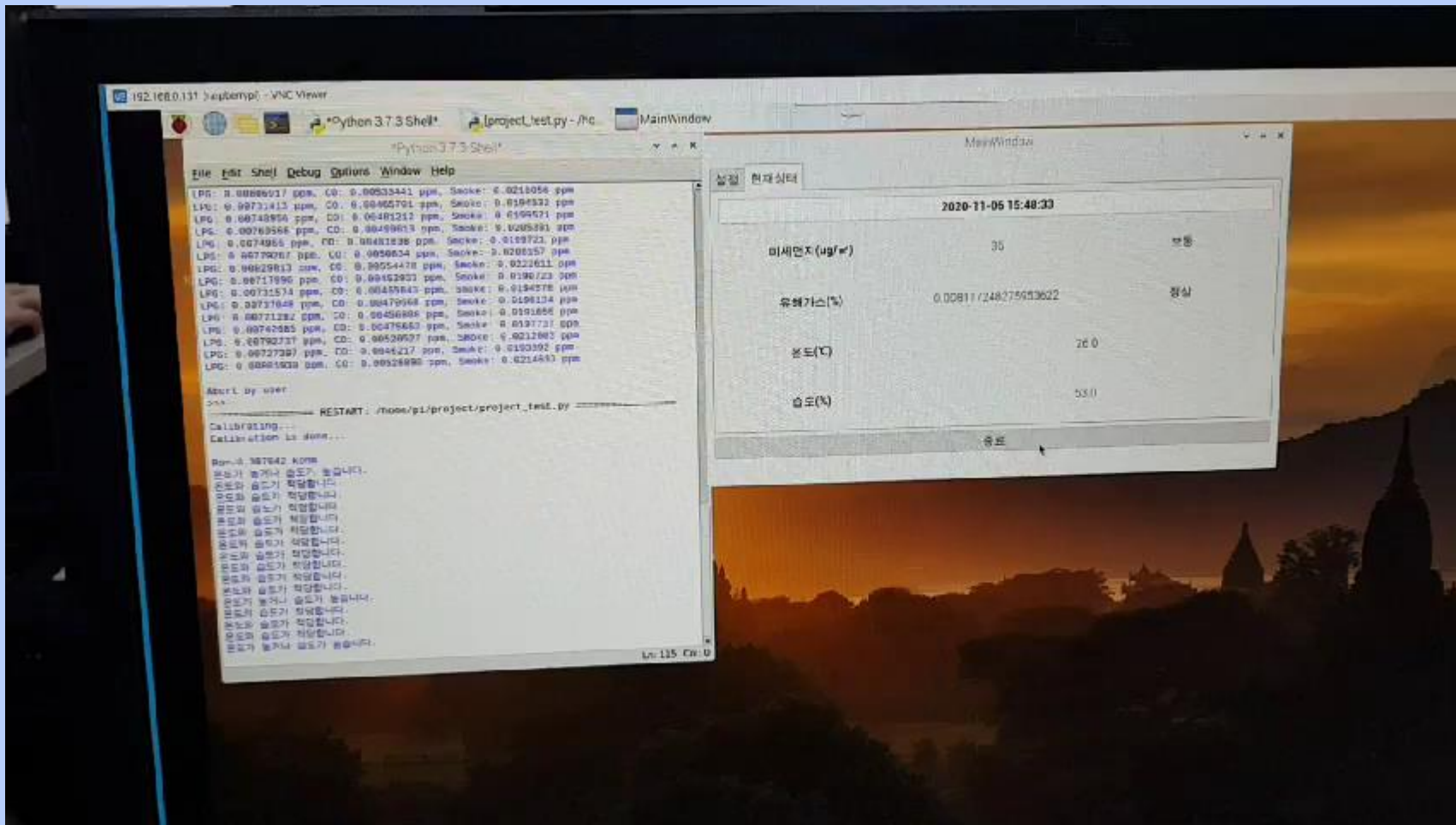
```
if(self.on_set_clicked == True):
    set_list1 = set_on_clicked()
    set_hum1 = set_list1[1]
    set_tem1 = set_list1[0]
    if(set_hum1 < h) | (set_tem1 < t):
        print("온도가 높거나 습도가 높습니다.")
        if(d < 15):
            self.d_image.setText(" 좋음 ")
            if(perc["GAS_LPG"] < 0.5):
                self.g_image.setText(" 정상 ")
            else:
                self.d_image.setText(" 경고 ")
        elif(d < 50):
            self.d_image.setText(" 보통 ")
            if(perc["GAS_LPG"] < 0.5):
                self.g_image.setText(" 정상 ")
            else:
                self.d_image.setText(" 경고 ")
```

```
elif(d < 100):
    self.d_image.setText(" 나쁨 ")
    if(perc["GAS_LPG"] < 0.5):
        self.g_image.setText(" 정상 ")
    else:
        self.d_image.setText(" 경고 ")
else:
    self.d_image.setText(" 매우나쁨 ")
    if(perc["GAS_LPG"] < 0.5):
        self.g_image.setText(" 정상 ")
    else:
        self.d_image.setText(" 경고 ")
```


04. Demo



04 Demo





감사합니다.