

# 유농알

조장: 김민석

조원: 이승연, 임하영, 장영수

Github address:

## 1. 어플리케이션 목적

유해가스농도 측정을 자동으로 수집하여 경고하기 위한 목적을 가지고 있다.

## 2. 어플리케이션의 네이밍의 의미

<유농알>

유(해가스) 농(도측정) 알(리미)의 줄임말

## 3. 어플리케이션의 활용 가치

- 유농알은 밀폐공간 등 유해가스 사용 작업을 시행하는 기업 및 근로자를 대상으로 한다.
- 유농알 어플리케이션은 근로자가 작업 시 착용하는 가스 농도 측정 기기를 통해 산소, 일산화탄소, 탄산가스, 황화수소, 탄화수소를 측정한다.
- 1 차적으로 정해진 농도 이상 올라가면 주의, 2 차적으로 정해진 농도 이상 올라가면 경고 알림을 올린다.
- 밀폐공간작업 및 유해가스 사용 작업 시 근로자를 질식, 발작, 혼절 등의 위험에서 보호하고, 작업장 유해가스 농도가 측정 기준 이상 올라가지 않도록 할 수 있다.

## 4. 어플리케이션 개발 계획

- 입력 변수는 무엇 : 각각의 유해가스의 측정된 수치
- 개발한 함수는 무엇을 연산하는 함수인지 : 입력된 수치가 정해진 기준을 벗어났을 경우 주의, 경고, 위험 등을 알리는 함수.
- 연산 과정은 어떻게?
  - 조건문, 반복문, 함수, 리스트 등 지금까지 배운 내용을 토대로 연산

## 5. 어플리케이션개발 과정

계획 후 실제 개발 과정을 기록

- 각 함수는 어떻게 동작하는 지 구체적으로 설명

- def 함수로 가스 종류 및 농도를 고정
- 사용자에게 정보(가스 종류, 각 농도 값) 를 받아
- if 문으로 지정한 유해가스농도의 범위 안에 드는지, 벗어나는지 판별
- While 문으로 사용자가 입력을 멈출 때까지 지속적으로 값 수령
- For 문으로 조건에 맞는 결과값 도출
- b. 에러 발생 지점
  - 가스 종류 중 산소만 농도 범위가 두가지로 나뉘어 조건문 적용하는데 에러 발생
  - 농도 기준의 범위를 85%로 재분류 하는 과정에서 에러 발생
- c. 에러 발생에 대한 해결책
  - if 문, elif 문을 이용해 조건을 한 번 더 나눴음
  - 산소는 농도 범위가 두가지이고 85%로 나누면 정상 범위가 극단적으로 작아 산소를 제외하고 4 가지 유해가스만 측정
- d. 해결책 적용 시 어떻게 변화
  - 정상 작동
- e. 동작 결과 캡처

전체 결과 :

```

안전 : 입력한 O2 농도는 안전 기준에 부합합니다. 가스: O2, 농도: 20.0 %
주의 : 입력한 CO 농도는 안전 기준 85%를 초과했습니다. 가스: CO, 농도: 8.5 ppm
위험 : CO2의 농도는 안전 기준을 초과했습니다. 가스: CO2, 농도: 1.7 ppm
안전 : 입력한 H2S 농도는 안전 기준에 부합합니다. 가스: H2S, 농도: 8.0 ppm
위험 : HC의 농도는 안전 기준을 초과했습니다. 가스: HC, 농도: 28.0 ppm
Excel 파일로 저장했습니다.
프로그램을 종료했습니다.
  
```

## 6. 어플리케이션 개발 후기

- a. 이번 프로젝트를 통해 배운 내용을 기반으로 팀원들과 함께 고민하고  
회의하며 코드를 작성하게 되어 보다 뜻 깊은 프로젝트가 되었습니다. 또한  
문제를 해결하는데 있어 필요한 다양한 접근법과 커뮤니케이션이  
중요하다는 것을 다시금 깨닫게 되었습니다.  
이러한 경험이 이후에 있을 다양한 상황을 해결하는데 밑바탕이 될 것이라  
믿어 의심치 않습니다. 함께한 팀원들, 도움을 준 교육생분들께 감사합니다.