

Cmm(calculate mildo and molar) 계산기

학번: 2218079

이름: 이은아

Github address: <https://github.com/2218079LEEEUNA/CMM>

1. 계산기의 목적

- Cmm 는 화학 물질의 물리적 및 화학적 특성을 계산하고 평가하는데 사용된다. 안전공학과 관련 된 주요 목적으로는 화학 물질의 물리적 및 화학적 특성을 계산하여 화학 물질의 안전성을 평가하고 관련 위험을 이해하는 것이다. 이로 인해 작업자 및 관리자 등이 화학물질을 처리하고 저장 및 취급하기 위한 정보를 제공받을 수 있다.
- 계산기 활용 대상 : 화학 공정 엔지니어, 화학자, 화학 연구원, 화학 물질을 취급하는 모든 사업장, 안전 공학 분야 전문가들 및 교육자 화재 안전 전문가 등

2. 계산기의 네이밍의 의미

[Cmm]는 화학 물질의 특성을 계산해주기 위해 개발하였다. 밀도와 물을 계산해주고 이해하기 위한 도구입니다.

3. 계산기 개발 계획

- 입력 변수 : 사용자로부터 입력 받은 화학 물질의 질량, 부피, 몰 수 등
- 개발한 함수 : “calculate_mildo”함수는 밀도를 계산하고, “calculate_molar”함수는 몰 질량을 계산하는 함수이다.
- 연산 과정 : 사용자에게 어떤 화학 물질의 특성을 구할 것인지 입력 받은 후 사용자가 선택 한 화학 물질의 특성에 따라 계산해준다. 예를 들어 밀도를 선택했다면 질량과 부피의 값을 입력 받아 밀도의 값을 구해주는 형식으로 진행한다.
- 조건문 : 1. 밀도계산, 2. 몰 질량 계산, 3. 종료 이렇게 3 개의 선택을 할 수 있기 때문에 각각의 진행코드를 작성하기 위해 if 조건문을 사용할 계획이다. if 조건문을 통해 사용자의 선택에 따라 어떤 함수를 호출할지 결정하고, 사용자가 필요한 계산을 진행하도록 도와줄 수 있도록 한다

4. 계산기 개발 과정

```
def calculate_mildo(mass, volume):
    mildo = mass / volume
    return mildo
```

- Calculate_mildo : 이 함수는 밀도를 계산하는 역할을 한다.
 사용자로부터 질량과 부피를 입력 받을 수 있게 설정한 후, 밀도가
 질량을 부피로 나눈 값이므로 그 식을 적용하여 프로그램이 밀도를
 계산하고 반환할 수 있도록 코드를 짰다.

```
def calculate_molar(mass, moles):
    molar = mass / moles
    return molar
```

- Calculate_molar : 이 함수는 몰 질량을 계산하는 역할을 한다.
 사용자로부터 질량과 몰 수를 입력 받아 몰 질량을 구하고 반환할
 수 있도록 설정하였다.

```
def main():
    print("화학물질 특성 계산기")

    while True:
        print("1. 밀도 계산")
        print("2. 몰 질량 계산")
        print("3. 종료")

        choice = input("선택 (1/2/3): ")
```

- 이 부분은 사용자에게 어떤 계산이 필요한지 선택창을 제공하기
 위해 설정한 코드이다. 앞서 말한 것과 같이 밀도와 몰을 구할 수
 있도록 설정한 계산기이기 때문에 반복문을 사용하여 3 가지
 선택지를 넣었다.

```

if choice == '1':
    mass = float(input("질량(g): "))
    volume = float(input("부피(L): "))

    mildo = calculate_mildo(mass, volume)
    print(f"밀도(g/L): {mildo}")

elif choice == '2':
    mass = float(input("질량(g): "))
    moles = float(input("몰 개수(mol): "))

    molar = calculate_molar(mass, moles)
    print(f"몰 질량(g/mol): {molar}")

elif choice == '3':
    print("프로그램을 종료합니다.")
    break

```

- ➔ If choice == '1'은 사용자가 1 을 선택했을 경우 밀도를 계산해 주기 위해 설정한 부분이다. 사용자에게 질량과 부피를 입력할 수 있는 창을 input 함수를 통해 제공하고 float 를 사용하여 숫자형으로 변환하여 준다. 또한 제공 받은 정보를 통해 calculate_mildo 함수를 사용하여 계산한 후 사용자에게 밀도를 계산하여 출력해준다.
- ➔ Elif choice == '2' if 함수가 참이 아닐 경우 실행되는 항으로 사용자가 1 이 아닌 2 를 선택하였을 경우 몰 질량을 계산해 주기 위해 설정한 부분이다. 사용자에게 질량과 몰 수를 입력 받고, calculate_molar 함수를 통해 계산한 수 몰 질량을 사용자에게 제공해준다.
- ➔ Eilf choice == '3'은 사용자가 1 도 2 도 아닌 3 을 선택하였을 경우 실행되는 항이다. 이는 계산기를 종료할 때 사용하기 위해 만들었기 때문에 사용자가 3 을 선택한다면 프로그램 종료 문구를 프린트 한 다음 break 함수로 반복문을 종료 시켜준다.

```
if __name__ == "__main__":
    main()
```

- 이 부분은 Python 프로그램을 실행 할 때 사용되는 표준 관례로, 프로그램을 직접 실행할 때만 main()함수를 호출하도록 설정한 부분이다. 이는 코드를 모듈로 사용하거나 직접 실행 할 때의 동작을 구분하기 위해 사용하는 것이다.

오류 발생부분과 해결

```
화학물질 특성 계산기
1. 밀도 계산
2. 몰 질량 계산
3. 종료
선택 (1/2/3): 4
1. 밀도 계산
2. 몰 질량 계산
3. 종료
선택 (1/2/3): 5
1. 밀도 계산
2. 몰 질량 계산
```

- 위의 함수로 코드를 돌려 마무리 했을 때 1,2,3 번 외의 숫자를 사용자가 입력했을 경우 계속해서 반복문이 돌아 끝이 나지 않기 때문에 이 부분을 오류라고 생각했다.

```
else:
    print("잘못된 선택입니다. 1, 2, 3 중에서 선택하세요.")
```

- 위 같은 상황을 해결하기 위해 추가한 항이다. 이를 통해 1,2,3 이외의 함수를 넣었을 경우에도 안내 문구를 사용자에게 제공하여 올바르게 프로그램이 돌아갈 수 있도록 하였다.

```

화학물질 특성 계산기
1. 밀도 계산
2. 몰 질량 계산
3. 종료
선택 (1/2/3): 4
잘못된 선택입니다. 1, 2, 3 중에서 선택하세요.
1. 밀도 계산
2. 몰 질량 계산
3. 종료
선택 (1/2/3): |

```

→ 오류를 수정하기 위해 추가항을 넣은 이후의 결과이다. 계획한 대로 사용자에게 안내문구를 출력하고, 다시 선택을 할 수 있도록 선택지를 제공하는 항이 출력되게 하였다.

```

화학물질 특성 계산기
1. 밀도 계산
2. 몰 질량 계산
3. 종료
선택 (1/2/3): 2
질량(g): 12
몰 개수(mol): 1.4
몰 질량(g/mol): 8.571428571428571

```

→ 최종적으로 몰 질량을 구하는 항을 선택하여 예시로 결과값을 출력하였다.

5. 계산기 개발 후기

계산기를 직접 개발하면서 사용자의 요구사항이 무엇일지 생각하고 이를 코드로 구현하는 과정이 복잡하고 어려웠다. 물론, 지금까지 내가 직접 사용하던 계산기 보다는 간단한 계산기일 수 있지만 계산기를 만드는 과정을 직접 경험해보면서 수업 때 배웠던 코드가 더 정확하게 이해되었다. 또한 프로그래밍은 조금의 어긋남이 있으면 바로 오류가 뜨기 때문에 이를 해결하려고 노력하는 과정에서 문제 해결 능력도 향상된 것 같다. 나는 프로그램을 개발한다는 것은 마냥 어려운 일이라고 생각했는데 내가

수업 때 배웠던 코드들을 사용하여 개발을 해본 후 프로그래밍과 좀 더 가까워졌고
흥미가 생겼다.