

재해율 및 비용 분석기

학번: 2118314

이름: 이준석

Github address: <https://github.com/Leejoonsuk01>

1. 계산기의 목적

- a. 본 계산기는 안전관리자 또는 근로자에게 자신에 재해율과 그에 따른 비용을 분석해주는 데에 있다.
- b. 특히 안전관리자는 이 계산기를 통해 근로자들의 재해율이 얼마인지 파악하여 관리를 보다 수월하게 하기 위한 목적으로 하였다.

2. 계산기의 네이밍의 의미

- a. 재해율 및 기용 분석기의 의미는 말 그대로 재해율과 재해 관련 비용을 분석한다는 기능을 직간접적으로 나타낼 수 있게 하였다.

3. 계산기 개발 계획

- a. 기본적으로 조건문과 반복문 위주 그리고 함수로 만들었다.

3-1 입력변수

- b. → 노출 시간 계산 : 근로자 수 x 주당 근무 시간 x 연간 근무 주 수
→ 재해율 계산 : 노출시간을 사용하여 재해 건수를 나눈 후, 표준화를 위해 200,000 을 곱한다.

```
def calculate_incident_rate(number_of_incidents, number_of_employees, hours_per_week, weeks_per_year):  
    # 노출 시간 계산  
    exposure_hours = number_of_employees * hours_per_week * weeks_per_year  
    # 재해율 계산  
    incident_rate = (number_of_incidents / exposure_hours) * 200000  
    return incident_rate, exposure_hours
```

→ 총 재해 비용 계산: 직접 비용에 간접 비용 배율을 곱한 후, 직접 비용과 합산합니다. 결과를 '만원' 단위로 조정한다.

```

def calculate_total_cost(direct_costs, indirect_multiplier=4):
    # 총 재해 비용 계산
    indirect_costs = direct_costs * indirect_multiplier
    total_costs = direct_costs + indirect_costs
    return total_costs / 10000 # 원에서 만원 단위로 변경

```

→ 안전 개선 조치 제안: 재해율을 분석하여 현재 상태에 따른 안전 조치를 제안한다.

```

def suggest_safety_measures(incident_rate):
    # 안전 개선 조치 제안
    if incident_rate < 0.5:
        return "현재 안전 관리가 잘 이루어지고 있습니다. 지속적인 모니터링과 기존 안전 프로그램 유지가 필요합니다."
    elif 0.5 <= incident_rate < 2.0:
        return "경미한 안전 문제가 있습니다. 위험 요소를 식별하고 개선하기 위한 안전 교육 및 작업장 환경 개선이 필요합니다."
    else:
        return "심각한 안전 문제가 있습니다. 즉각적인 개선 조치와 안전 프로그램 강화가 필요합니다."

```

→ 기대 효과 계산: 안전 개선 후의 재해율과 경제적 이익을 계산한다.

```

def calculate_expected_savings(incident_rate, total_costs, expected_reduction_rate):
    # 기대 효과 계산
    expected_incident_rate = incident_rate * (1 - expected_reduction_rate)
    expected_savings = total_costs * expected_reduction_rate
    return expected_incident_rate, expected_savings

```

3-2 연산과정

- c. ① 초기 입력 받기
- ② 연산 수행
- ③ 반복문과 조건문 실행
- ④ 결과 도출

4. 계산기 개발 과정

- a. 에러 발생. 사용자 입력에서 문자열을 쓰면 에러가 발생

```
재해 건수를 입력하세요 (건): 1건
Traceback (most recent call last):
  File "C:\Users\이준석\PycharmProjects\pythonProject\.venv\과제.py", line 67, in <module>
    main()
  File "C:\Users\이준석\PycharmProjects\pythonProject\.venv\과제.py", line 33, in main
    number_of_incidents = float(input("재해 건수를 입력하세요 (건): "))
                                                ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
ValueError: could not convert string to float: '1건'

Process finished with exit code 1
```

해결 방법: 숫자로 사용하라는 함수를 사용으로 에러 해결

```
def get_float_input(prompt):
    while True:
        user_input = input(prompt)
        try:
            return float(user_input)
        except ValueError:
            print("입력이 잘못되었습니다. 숫자를 입력해주세요.")

def get_int_input(prompt):
    while True:
        user_input = input(prompt)
        try:
            return int(user_input)
        except ValueError:
            print("입력이 잘못되었습니다. 정수를 입력해주세요.")
```

```
재해 건수를 입력하세요 (건): 1건
입력이 잘못되었습니다. 숫자를 입력해주세요.
재해 건수를 입력하세요 (건):
```

동작 결과 캡쳐

```
재해 건수를 입력하세요 (건): 1
근로자 수를 입력하세요 (명): 200
주당 근무 시간을 입력하세요: 42
연간 근무 주 수를 입력하세요: 310
직접 비용을 입력하세요 (원): 10000000
간접 비용 배율을 입력하세요 (예: 1, 2, 3, 4): 1.2
기대하는 재해율 감소율을 입력하세요 (예: 0.1, 0.2, 0.3): 0.1

노출 시간: 2604000 시간
계산된 재해율: 0.08 (백만 시간당 재해 건수)
계산된 총 재해 비용: 2200.00만원
제안된 안전 개선 조치: 현재 안전 관리가 잘 이루어지고 있습니다. 지속적인 모니터링과 기존 안전 프로그램 유지가 필요합니다.
기대되는 새로운 재해율: 0.07 (백만 시간당 재해 건수)
기대되는 경제적 이익: 220.00만원

더 많은 데이터로 계산을 계속하시겠습니까? (yes/no):
```

5. 계산기 개발 후기

우선 이 계산기를 만든 이유는 안전관리자 또는 근로자들이 자신이 일을 하고 있는 현장에서의 발생 할 수 있는 재해율을 알면 재해의 빈도를 줄일수 있지 않을까 하는 고민에서 만들었으며 이러한 프로그래밍을 통해서 간단하게 숫자만을 입력해서 위험성을 빠르게 파악할수 있다는 것에서 뿌듯함을 느꼈다.