

HRN 기계위험성 평가 계산기

학번: 2318078

이름: 유원호

Github address: <https://github.com/dbdnjsgh12/HRN->

1. 계산기의 목적

- a. 안전관리자가 기계 위험성을 평가하기 위해 사용하는 HRN(Hazard Rating Number) 방법은 사고 발생 가능성, 위험 또는 재해로의 노출 빈도, 발생할 가능성이 있는 상해의 정도, 그리고 재해에 노출된 사람 수를 곱하여 수치로 위험성을 추정합니다. 이 방법을 통해 해결조치의 우선순위를 정할 수 있습니다. 안전관리자는 이러한 위험성 평가를 위해 기계의 위험성을 계산해야 할 때가 있습니다. 그러나 수치를 직접 곱하고 수치범위를 비교하는 것은 복잡하고 번거로울 수 있습니다. 이를 해결하기 위해 위험성 평가 계산기를 사용할 수 있습니다. 이 도구를 사용하면 사용자가 입력한 수치를 기반으로 HRN을 계산하여 기계의 위험성을 추정하고, 이를 바탕으로 해결조치의 우선순위를 정할 수 있습니다.

- b. 계산기 활용 대상: 기계위험성 평가를 하는 안전관리자

2. 계산기의 네이밍의 의미

- a. HRN 기계위험성 평가 계산기는 HRN방법을 사용해 위험성추정을 계산해주는 계산기이다.

3. 계산기 개발 계획

a. 변수

likelihood: 발생 가능성을 나타내는 변수입니다. 사용자로부터 입력받은 값으로, 사건이 발생할 가능성의 정도를 나타냅니다.

frequency: 노출 빈도를 나타내는 변수입니다. 사용자로부터 입력받은 값으로, 사건에 노출된 빈도를 나타냅니다.

severity: 상해의 정도를 나타내는 변수입니다. 사용자로부터 입력받은 값으로, 사건이 발생했을 때 상해의 정도를 나타냅니다.

exposure: 노출된 사람 수의 정도를 나타내는 변수입니다. 사용자로부터 입력받은 값으로, 사건에 노출된 사람 수를 나타냅니다.

HRN: HRN(Hazard Rating Number)을 나타내는 변수입니다. likelihood, frequency, severity, exposure를 곱하여 계산된 위험성을 나타냅니다.

grade: 등급을 나타내는 변수입니다. HRN에 따라 할당된 위험 등급을 저장합니다.

b. 위험성 평가를 하는 위험등급을 계산하는 계산기

c. 연산과정은 사용자가 입력한 발생 가능성(likelihood), 노출 빈도(frequency), 상해의 정도(severity), 노출된 사람 수(exposure)를 받아와 곱하여 HRN을 계산합니다. HRN이 계산된 후, 해당하는 위험 등급을 할당합니다. 조건문을 사용하여 HRN의 범위에 따라 적절한 등급을 선택합니다.

위 연산 과정을 통해 사용자가 입력한 값들을 기반으로 HRN을 계산하고, 해당하는 위험 등급을 할당하여 출력합니다. 또한 main 함수는 반복문을 통해 사용자가 계속해서 위험 평가를 수행할 수 있도록 합니다. 사용자가 더 이상 평가를 원하지 않을 경우 프로그램을 종료합니다. 이 프로그램에서 조건문이 필요한 이유는 HRN 값에 따라 해당하는 위험 등급을 할당하기 위해서입니다. HRN은 사용자가 입력한 발생 가능성, 노출 빈도, 상해의 정도, 노출된 사람 수를 곱한 값으로, 이 값이 서로 다르면 위험의 정도도 다를 수 있습니다. 따라서 조건문은 HRN 값의 범위를 검사하여 해당하는 위험 등급을 결정합니다. 예를 들어, HRN 값이 1 이하이면 "무시 할 수 있는 위험"으로 판단하고, HRN 값이 2에서 5 사이이면 "매우 낮은 위험"으로 판단합니다. 이렇게 HRN 값의 범위에 따라 다른 등급을 할당함으로써, 사용자는 입력한 값들을 바탕으로 HRN과 해당하는 위험 등급을 쉽게 확인할 수 있습니다.

4. 계산기 개발 과정

a. 사용자가 프로그램을 실행하면, main 함수가 호출됩니다.

main 함수에서는 사용자로부터 사고발생 가능성(likelihood), 노출 빈도(frequency), 상해의 정도(severity), 노출된 사람 수(exposure)를 입력 받습니다.

input 함수를 사용하여 사용자로부터 각각의 입력 값을 받습니다.

사용자는 사고 발생 가능성, 노출 빈도, 상해의 정도, 노출된 사람 수를 숫자로 입력합니다.

사용자가 입력한 값을 기반으로 calculate_HRN 함수가 호출됩니다.

calculate_HRN 함수는 받은 입력 값들을 곱하여 HRN을 계산합니다.

이 때, likelihood(발생 가능성), frequency(노출 빈도), severity(상해의 정도), exposure(노출된 사람 수)를 곱하여 HRN을 계산합니다.

HRN이 계산된 후에는 조건문은 if, elif, else를 사용하여 HRN이 속하는 등급을 판별하고 해당 등급을 할당한다.

계산된 HRN 값과 할당된 등급이 사용자에게 출력됩니다.

이를 통해 사용자는 입력한 값들을 바탕으로 계산된 위험 등급을 확인할 수 있습니다.

프로그램은 사용자에게 추가적인 입력을 원하는지 묻습니다.

while True 문은 무한 루프를 생성하여 프로그램이 사용자가 원할 때까지 계속 실행될 수 있도록 합니다. 사용자가 프로그램을 종료하고자 할 때까지 루프를 유지하고, 사용자가 추

가 입력을 원할 경우에도 다시 루프를 실행합니다.

사용자가 "Y" 또는 "y"를 입력하면 프로그램이 다시 시작되어 사용자는 새로운 입력을 할 수 있고 사용자가 "N" 또는 "n"을 입력하면 프로그램이 종료됩니다.

b. 이 프로그램은 HRN(Hazard Rating Number)을 계산하여 위험 등급을 할당하는 데 사용됩니다. 이 프로그램은 'calculate_HRN'과 'main'두 개의 함수로 구성되어 있습니다.

'calculate_HRN' 함수

이 함수는 네 개의 매개변수(likelihood, frequency, severity, exposure)를 받아 HRN을 계산하고 해당하는 위험 등급을 할당합니다.

HRN은 입력된 네 개의 매개변수를 곱하여 계산됩니다.

계산된 HRN에 따라 적절한 등급이 할당됩니다.

'main' 함수

이 함수는 사용자로부터 발생 가능성, 노출 빈도, 상해의 정도, 노출된 사람 수를 입력 받고, 이를 calculate_HRN 함수에 전달하여 HRN을 계산하고 등급을 출력합니다.

사용자가 추가 입력을 원할 경우 프로그램을 계속 실행합니다.

c. calculate_HRN 함수에서 HRN 변수를 사용하기 전에 정의되지 않아 NameError가 발생하였습니다.

d. calculate_HRN 함수 내에서 HRN 변수를 정의하기 전에 값을 할당해야 합니다.

HRN 변수를 likelihood * frequency * severity * exposure로 초기화하여 정의하면 됩니다.

e. 코드의 구조가 변경되어 HRN 변수가 calculate_HRN() 함수 내에서 계산 및 정의됩니다.

함수가 호출될 때마다 HRN 값이 계산되며, 이 값이 반환됩니다.

f. 동작 결과

```
사고발생 가능성을 0부터 15까지의 숫자로 입력하세요: 10
노출 빈도를 0부터 5까지의 숫자로 입력하세요: 3
상해의 정도를 0부터 15까지의 숫자로 입력하세요: 7
사고에 노출된 사람 수의 정도를 1부터 12까지의 입력하세요: 5
```

```
===== HRN 및 등급=====
```

```
HRN: 1050
```

```
등급: 극한 위험
```

```
더 평가하시겠습니까? (Y/N): N
```

▲ 마지막에 N을 입력한 결과

```
사고발생 가능성을0부터15까지의 숫자로 입력하세요: 10
노출 빈도를0부터5까지의 숫자로 입력하세요: 3
상해의 정도를0부터15까지의 숫자로 입력하세요: 8
사고에 노출된 사람 수의 정도를1부터12까지의 입력하세요: 5
```

```
===== HRN 및 등급=====
```

```
HRN: 1200
```

```
등급: 극한 위험
```

```
더 평가하시겠습니까? (Y/N): Y
```

```
사고발생 가능성을0부터15까지의 숫자로 입력하세요: 5
```

```
노출 빈도를0부터5까지의 숫자로 입력하세요: 2
```

```
상해의 정도를0부터15까지의 숫자로 입력하세요: 3
```

```
사고에 노출된 사람 수의 정도를1부터12까지의 입력하세요: 8
```

```
===== HRN 및 등급=====
```

```
HRN: 240
```

```
등급: 상당한 위험
```

```
더 평가하시겠습니까? (Y/N): N
```

▲ 마지막에 Y를 입력한 결과

계산기 개발 후기

a. 다른 전공수업과제로 기계 위험성평가를 할때에는 여러 가지 위험원을 찾고 수치를 직접 곱하여 계산해야 해서 번거롭고 계산의 오류가 있어 다시 계산하고 고치는데에 불편함을 느꼈지만 위 프로그램을 통해 일정한 규칙에 따라 위험을 평가하고 등급을 부여하는 작업을 자동화함으로써 시간과 인적 자원을 절약할 수 있겠다고 느꼈고 이는 안전 관리 업무를 보다 효율적으로 수행할 수 있도록 도와줄 수 있을 거라고 생각되었습니다. 또 제가 만든 프로그램은 간단한 계산 프로그램이지만 더 확장시켜 다른 위험성 평가 방법들도 프로그램으로 만들어 현장에서 사용할 수 있다고 생각했습니다.

또, 처음 전공 선택과목으로 프로그래밍 수업이 있었을 때 안전 공학을 배우면서 프로그래밍이 어디에 쓰이고 현장에서 쓸 수 있을까에 대한 의문과 프로그래밍을 하는것에 대해 자신이 없어 걱정이 있었는데 수업을 하면서 어디에나 쓸수 있다는 것을 느꼈고 과제로 직접 계산 프로그램을 만들면서 내가 원하는 작업이 실행되는 것을 보고 뿌듯함을 느꼈습니다.

