

Stage 4 - 급여 계산기 입력받기

급여 계산기를 한단계 업그레이드 시켜봅니다. 사용자로부터 가변적으로 시급과 일일근무시간, 한달근무일수를 입력받아 급여 계산기를 작동시킵니다.

사용자로부터 입력받기

프로그램의 코드는 개발자가 보는 것입니다. 사용자는 코드를 보지 않고 실행된 결과만 볼 수 있습니다. 그러면 우리는 실행하는 과정에서 사용자로부터 시급, 일일근무시간, 한달근무일수에 대해 입력받고 싶은 것이죠.

```
pay_calculator (1) × /Users/duhui/venv/python_day1/bin/python /Users/d
시급계산기 프로그램입니다!
시급을 입력해주세요 : 8500
일일근무시간 : 10
한달근무일수 : 15
예상 월급은 : 1275000원입니다.

1275000원으로 무엇을 할 수 있을까요?
PC방 (1200원 기준) : 1062시간
점심 (7000원 기준) : 182끼
영화 (11000원 기준) : 115편
노래방 (20000원 기준) : 63시간

열심히 일하시는 당신에게...
비록 세상이 그대를 속일지라도
슬퍼하거나 노여워하지 말아라
슬픈 날은 참고 견디라
기쁜 날이 오고야 말리니.

Process finished with exit code 0
```

사용자에게 입력받기

input문 알아보기

input문

'input'을 한영사전에 검색하면 '입력하다'라는 뜻이 있습니다. input문이 실행되면 사용자가 Enter 키를 누르기 전까지 다음 동작이 실행되지 않습니다. 즉, 사용자가 어떤 값을 키보드로 입력을 해주고 '난 다 입력했다' 신호를 주어야 다음으로 프로그램이 실행됩니다.

input문을 통해 사용자로부터 받아온 값은 문자의 형태로 인식합니다. 사용자가 100000을 입력하더라도 프로그램은 일영영영영영으로 인식합니다.

```
pay = input("시급을 입력해주세요 : ")
```



input문

input문의 괄호에는 안내문이 들어갈 수 있습니다. 위와 같은 예시 문을 보면 "시급을 입력해주세요 : "라는 안내문이 출력된 뒤에 사용자가 입력하면 그 정보가 pay라는 변수에 담기게 됩니다. 문자의 형태로 말이죠.

급여계산기에서 input문 사용하기

pay, time_of_day, day_of_month 3개의 변수의 값을 사용자로부터 입력받는 코드입니다.

```
calculators/pay_calculator.py
```

```
1 pay = input("시급을 입력해주세요 : ")
2 time_of_day = input("일일근무시간 : ")
3 day_of_month = input("한달근무일수 : ")
```

자료형이란 무엇인가?

자료형이란?

우리는 지금까지 간접적으로 자료형에 대해 느끼고 있었습니다. 바로 숫자, 문자를 구분하면서 말입니다. 변수에 값을 담을 때, 변수에는 다양한 형태의 값이 들어갈 수 있습니다. 그래서 이를 구분하기 위해 자료형이 존재합니다.

자료형	설명	표기법	예
정수	정수 표현된 값	int	-123, -1, 0, 23, 10000, 98368
실수	실수 표현된 값	float	-0.12, 1/3, 123.23, 512.00
문자	따옴표 사이의 문자	str	'i love you', '111', "123.123"
불	참/거짓의 값	bool	True, False
리스트	순서가 있는 집합	list	[Val0, Val1, ...]
튜플	수정이 불가능한 리스트	tuple	(Val0, Val1, ...)
집합	순서가 없는 중복 불가 리스트	set	{Val0, Val1, ...}
딕셔너리	쌍으로 이뤄진 값들의 리스트	dic	{Key0 : Val0, Key1 : Val1 ...}

급여 계산기의 문제

우리가 당착한 문제가 무엇인지 짐작이 가시나요? 우리는 급여를 계산하기 위해서 곱하기 연산을 사용합니다. 그런데 input문을 통하여 사용자로부터 받은 값은 문자라고 배웠습니다. 과연 문자끼리의 곱하기가 가능할까요? 불가능합니다. 예시를 통해 살펴보겠습니다.

숫자끼리의 덧셈

example

```
1 print(123+123)
2 # 246
```

문자끼리의 덧셈

example

```
1 print('123'+ '123')
2 # 123123
```

문자와 숫자의 덧셈

example

```
1 print('123'+123)
2 # error
```

숫자끼리의 곱셈

example

```
1 print(123*3)
2 # 369
```

문자끼리의 곱셈

example

```
1 print('123'*'3')
2 # error
```

문자와 숫자의 곱셈

example

```
1 print('123'*3)
2 # 123123123
```

이해가 되시나요? 문자끼리의 곱셈은 컴퓨터가 인식하지 못합니다. 그래서 급여 계산기를 작성할 때 문자끼리의 곱셈이 아닌 숫자끼리의 곱셈이 될 수 있도록 문자를 숫자로 바꾸어주는 과정이 필

요합니다.

형변환하기

형변환이란?

자료형을 바꾸는 행위를 형변환이라고 합니다. 숫자에서 문자로, 문자에서 숫자로, 문자에서 리스트로, 리스트에서 집합으로 등등 이와 같은 과정을 형변환이라고 합니다.

pay = int(pay)

pay 변수의 형변환

pay라는 변수는 input문을 통해 값을 입력받았기 때문에 문자입니다. 그러나 우리는 숫자로 바꿔주어야 곱하기가 가능하기 때문에 정수로 바꾸어주는 연산을 하고 싶습니다.

형변환은 자료형의 표에서 표기법을 쓰고 괄호안에 바꾸고자 하는 변수나 값을 넣으면 표기법에 해당하는 자료형으로 변환이 됩니다.

급여 계산기 문제 해결하기

input문을 통해 값을 받은 pay, time_of_day, day_of_month 변수를 정수로 형변환하여 봅시다.

```
calculators/pay_calculator.py
```

```
1 pay = int(pay)
2 time_of_day = int(time_of_day)
3 day_of_month = int(day_of_month)
```

급여 계산기 완성하기

calculators/pay_calculator.py

```
1  print("급여계산기 프로그램입니다!")
2
3  pay = input("시급을 입력해주세요 : ")
4  time_of_day = input("일일근무시간 : ")
5  day_of_month = input("한달근무일수 : ")
6
7  pay = int(pay)
8  time_of_day = int(time_of_day)
9  day_of_month = int(day_of_month)
10
11 # 월급 계산
12 profit = pay * time_of_day * day_of_month
13
14 # 결과 출력
15 print("예상 월급은 : {0}원입니다.\n\n".format(profit))
16 print("{0}원으로 무엇을 할 수 있을까요?".format(profit))
17 print("PC방 (1200원 기준) : {0}시간".format(profit // 1200))
18 print("점심 (7000원 기준) : {0}끼".format(profit // 7000))
19 print("영화 (11000원 기준) : {0}편".format(profit // 11000))
20 print("노래방 (20000원 기준) : {0}시간".format(profit // 20000))
21
22 print('''
23 열심히 일하시는 당신에게...
24 비록 세상이 그대를 속일지라도
25 슬퍼하거나 노여워하지 말아라
26 슬픈 날은 참고 견디라
27 기쁜 날이 오고야 말리니.''')
```