

프로그래밍2 튜터링

(2회차)

조세연

목표

- 1차시에 배웠던 이론들을 실습에서 사용해보기
- 9/26에 진행했던 파이썬 시험 문제 풀이

문제풀이

코딩 테스트(5차시)

아래와 같은 하나의 string을 **리스트로 변환**해서 출력하세요. 문자열에는 정수, 콤마, 빈칸만 사용할 수 있습니다. string 내부에는 **동일한 숫자**를 사용할 수 없습니다.

아래의 문제에서는 2가지 문자열을 테스트하세요. 이 문제는 아래를 입력받아서 **2번째 가장 큰 값**을 출력하는 프로그램을 작성하는 문제입니다. 이를 순서대로 구현하시오.

- "3, 5, 10, 20, 6, 100, 50, 1"
- "34, 21, 3, 100, 12, 82, 77, 91, 26"

문제풀이

코딩 테스트(5차시)

- 위의 내용을 함수로 구현하세요. 위의 문제의 입력과 같이 **하나의 string**을 파라미터로 받아서, **정수 리스트를 반환**하는 함수를 작성하시오.

- 이 함수의 이름은 **string_2_list()**로 하세요.

- 이 함수는 **list**를 반환합니다.

```
def string_2_list(strNum):  
  
    arrNum = list(map(int, strNum.split(",")))  
  
    return arrNum
```

문제풀이

코딩 테스트(5차시)

- 위에서 만든 **list**를 **파라미터**로 받아서 **최대값을 반환**하는 **함수**를 작성하시오.
 - 함수의 이름은 **max_of_list()**로 하세요.
 - 파이썬의 built-in 함수인 **max()** 함수를 사용하지 마세요.
 - 이 함수는 **int**를 **반환**합니다.
 - 이 함수를 이용하여, 위의 2개의 string 샘플을 테스트, 출력하시오.

```
def max_of_list(arrNum):  
    nMax = arrNum[0]  
  
    for i in arrNum:  
        if i > nMax:  
            nMax = i  
  
    return nMax
```

문제풀이

코딩 테스트(5차시)

- 위의 `max_of_list()` 함수를 변경해서 `second_max_of_list()` 함수를 작성하시오.

- `max()` 함수를 사용하지 마세요.

- 이 함수는 `int`를 반환합니다.

- 이 함수를 이용하여, 위의 2개의 string 샘플을 테스트, 출력하시오.

```
def second_max_of_list(arrNum):
```

```
    nSecondMax = arrNum[0]
```

```
    for i in arrNum:
```

```
        if i > nSecondMax and i < max_of_list(arrNum):
```

```
            nSecondMax = i
```

```
    return nSecondMax
```

문제풀이

코딩 테스트(5차시)

- 지금부터는, 아래의 2가지 문자열을 테스트하세요. 이 문제에는 숫자가 중복이 있을 수 있습니다.
 - "3, 5, 10, 100, 3, 50, 1, 20, 6, 100, 50, 1, 40, 100"
 - "34, 91, 3, 12, 5, 91, 3, 100, 12, 82, 77, 91, 26, 12, 3"
- 위에서 작성한 함수들을 변경해서 `second_max_of_list_with_same_numbers()` 함수를 작성하시오.
 - 중복된 숫자들은 1개로 정리한 후.
 - `second_max`를 출력해야 합니다.
 - 이 함수를 이용하여, 위의 2개의 string 샘플을 테스트, 출력하시오.

문제풀이

코딩 테스트(5차시)

```
def second_max_of_list_with_same_numbers(arrNum):
```

```
    arrNew = []
```

```
    for i in arrNum:
```

```
        if i not in arrNew:
```

```
            arrNew.append(i)
```

```
    return second_max_of_list(arrNew)
```


문제풀이

코딩 테스트(5차시)

```
strNum = input("문자열을 입력하세요: ")
```

```
arrNum = string_2_list(strNum)
```

```
print("first max number: ", max_of_list(arrNum))
```

```
print("second max number: ", second_max_of_list(arrNum))
```

```
print("second max number(중복제거): ", second_max_of_list_with_same_numbers(arrNum))
```

문제풀이

예제 1 - 문제

- 정수 n , m 을 각각 입력받는다.
- 별(*) 문자를 이용해 가로 길이가 n , 세로의 길이가 m 인 직사각형 형태를 출력하시오.

!! 제한조건 !!

- n 과 m 은 각각 1000 이하인 자연수여야 합니다.

* 입력: 5 3

* 출력: *****

문제풀이

예제 1 - 풀이

```
n, m = map(int, input().strip().split(' '))
```

```
if n >= 1000 and m >= 1000:
```

```
    pass
```

```
for i in range(m):
```

```
    for j in range(n):
```

```
        print("*", end="")
```

```
    print("\n", end="")
```

* pass란?

- 클래스, 함수, 반복문, if 문 등에서 사용 가능
- 클래스: 내부 동작은 필요 없고, 의미적으로 껍데기만 필요한 경우
- 함수: 실제 내부 구현은 필요 없지만, 추상 클래스에서 사용하는 경우에 이름을 전달해줄 필요가 있는 경우
- 반복문/if문: 어떠한 조건에서 코드를 수행하고 싶지 않을 경우

문제풀이

예제 2 - 문제

- 문자열 s에는 공백으로 구분된 숫자들이 저장되어 있다.
- str에 나타나는 숫자 중 최소값과 최대값을 찾아 이를 "(최소값) (최대값)" 형태의 문자열을 반환하는 함수, solution을 완성하시오.
- 예를들어 s가 "1 2 3 4"라면 "1 4"를 리턴하고, "-1 -2 -3 -4"라면 "-4 -1"을 리턴하면 됩니다.

!!! 제한조건 !!!

- s에는 둘 이상의 정수가 공백으로 구분되어 있다.

* 입력: "-1 -1"

* 출력: "-1 -1"

문제풀이

예제 2 - 풀이

```
def solution(s):
```

```
    arr = list(map(int, s.strip().split(' ')))
```

```
    answer = str(min(arr)) + " " + str(max(arr))
```

```
    return answer
```

감사합니다:)