

함수 응용

국민대학교
이정미교수

학습내용

함수 응용

문자열

튜플

1. 함수(function)응용

함수 사용의 장점

- 프로그램 안에서 중복된 코드를 제거한다.
- 복잡한 프로그래밍 작업을 더 간단한 작업들로 분해할 수 있다
- 함수는 한번 만들어지면 다른 프로그램에서도 재사용될 수 있다.
- 함수를 사용하면 가독성이 증대되고, 유지 관리도 쉬워진다.

실습예제 - 소수판별

- 여기서는 소수를 판별하는 함수 `is_prime()`을 작성하여 사용하여 보자.

```
# 소수 판별 함수 (2이상의 자연수에 대하여)
# 함수 is_prime()
def is_prime(n):      # 매개변수
    for i in range(2, n):
        if ( n%i == 0 ):
            return False
    return True

n = int(input("정수를 입력하시오: "))
print(is_prime(n))
```

정수를 입력하시오: 101
True

실습예제 - 패스워드 생성기

- 패스워드 생성기

```
import random
def genPass():
    a = "abcdefghijklmnopqrstuvwxyz0123456789"
    password = ""
    for i in range(6):
        index = random.randrange(len(a))
        password = password + a[index]
    #print(len(a)) == 36개
    return password

print(genPass())
print(genPass())
print(genPass())
```

q546zv
1kvkss
b3vrmi

디폴트 인수

- 파이썬에서는 함수의 매개변수가 기본값을 가질 수 있다. 이것을 **디폴트 인수(default argument)**라고 한다.

```
def greet(name, msg="별일없죠?"):
    print("안녕 ", name + ', ' + msg)
```

```
greet("소희")
```

안녕 소희,
별일없죠?

실습예제 – 계산기 사칙연산

- 이들 함수를 이용하여 $10+20*30$ 을 계산하여 보자.

```
def add(a, b):  
    return a + b
```

```
def sub(a, b):  
    return a - b
```

```
def mul(a, b):  
    return a * b
```

```
def div(a, b):  
    return a / b
```

```
r1 = mul(a=20, b=30)  
r2 = add(a=10, b=r1) #  $10+(20*30)$   
print(r2)
```

610

실행 과정

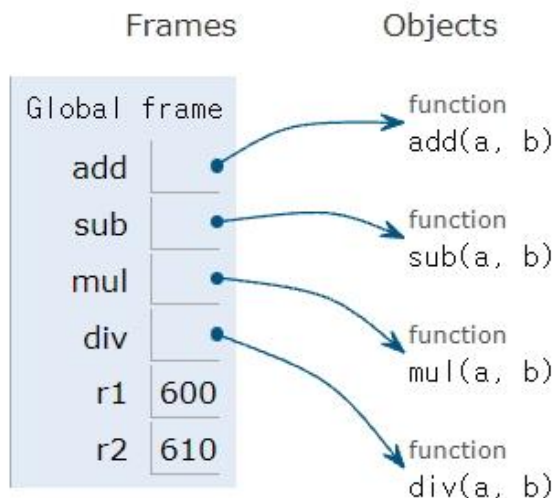
<https://pythontutor.com/visualize.html#mode=display>

Python 3.6
([known limitations](#))

```
1 def add(a, b):  
2     return a + b  
3  
4 def sub(a, b):  
5     return a - b  
6  
7 def mul(a, b):  
8     return a * b  
9  
10 def div(a, b):  
11     return a / b  
12  
13 r1 = mul(a=20, b=30)  
14 r2 = add(a=10, b=r1)  
→ 15 print(r2)
```

Print output (drag lower right corner to resize)

610



실습예제 – 리스트 예제(문자열)

- 강아지들의 이름을 저장하였다가 출력하는 프로그램을 작성해보자.

```
dogNames = []
while True:
    name = input('강아지의 이름을 입력하시오(종료시에는 엔터키) ')
    if name == '':
        break
    dogNames.append(name)

print('강아지들의 이름:')
for name in dogNames:
    print(name, end=" ", " ")
```

```
강아지의 이름을 입력하시오(종료시에는 엔터키) 미나
강아지의 이름을 입력하시오(종료시에는 엔터키) 초롱이
강아지의 이름을 입력하시오(종료시에는 엔터키)
강아지들의 이름:
미나, 초롱이
```

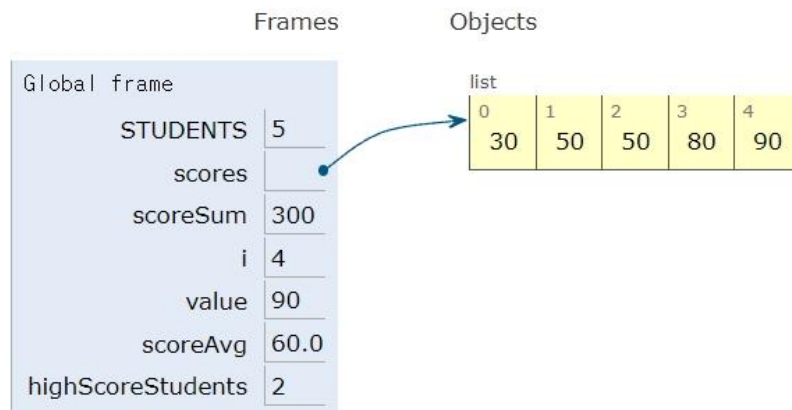
실습예제 – 리스트 예제 성적 처리 프로그램

- 성적의 평균을 구하고 80점 이상 성적을 받은 학생의 숫자를 계산하여 출력해보자.

```
STUDENTS = 5
scores = []
scoreSum = 0
for i in range(STUDENTS):
    value = int(input("성적을 입력하시요: "))
    scores.append(value)
    scoreSum += value
```

```
scoreAvg = scoreSum / len(scores)
```

```
highScoreStudents = 0
for i in range(len(scores)):
    if scores[i] >= 80:
        highScoreStudents += 1
```



실습예제 – 성적 처리 프로그램

- 성적의 평균을 구하고 80점 이상 성적을 받은 학생의 숫자를 계산하여 출력해보자.

```
print("성적 평균은", scoreAvg, "입니다.")  
print("80점 이상 성적을 받은 학생은", highScoreStudents, "명입니다.")
```

```
성적을 입력하시요: 10  
성적을 입력하시요: 20  
성적을 입력하시요: 60  
성적을 입력하시요: 70  
성적을 입력하시요: 80  
성적 평균은 48.0 입니다.  
80점 이상 성적을 받은 학생은 1 명입니다.
```

추가 문자열, 튜플 사용하기

리스트 정렬하기

1. 리스트 객체의 sort() 메소드를 사용하는 방법

```
>>> a = [ 3, 2, 1, 5, 4 ]  
>>> a.sort()  
>>> a  
[1, 2, 3, 4, 5]
```

문자열

1. 문자열은 문자들의 시퀀스로 정의된다. 글자들이 실(string)로 묶여 있는 것이 문자열이라고 생각하면 된다.
2. in 연산자와 not in 연산자, 단어 분리 split()

```
>>> word = 'abcdef'
>>> print(word[0])
'a'
>>> word[5]
'f'
>>> s="Love will find a way."
>>> "Love" in s      # print("Love" in s)
True
>>> "love" in s
False
s = ' I love you'
>>> s.split()
```

튜플 (tuple)은 변경될 수 없는 리스트

1. 튜플은 리스트보다 접근 속도가 빠르다
2. 한번 만들어지면 내용을 변경 할 수 없다
3. 생성하려면 소괄호(...) 안에 요소들을 넣으면 된다.

```
>>> numbers = ( 1, 2, 3, 4, 5 )  
>>> colors = ("red", "green", "blue")  
>>> t = numbers + colors  
>>> t  
(1, 2, 3, 4, 5, 'red', 'green', 'blue')
```


실습예제 – 함수의 튜플 반환 예제

- 원의 넓이와 둘레를 동시에 반환하는 함수를 작성

```
import math

def calCircle(r):
    # 반지름이 r인 원의 넓이와 둘레를 동시에 반환하는 함수 (area, circum)

    area = math.pi * r * r
    circum = 2 * math.pi * r
    return (area, circum)

radius = float(input("원의 반지름을 입력하시오: "))
(a, c) = calCircle(radius)
print("원의 넓이는 "+str(a)+"이고 원의 둘레는"+str(c)+"이다.")
```

실습예제 – 함수의 튜플 반환 예제

- 원의 넓이와 둘레를 동시에 반환하는 함수를 작성

원의 반지름을 입력하시오: 10

원의 넓이는 314.1592653589793이고 원의 둘레는 62.83185307179586이다.

수고 하였습니다.