

In [2]:

```
# 물리학과 2014103332 김민기 HW2_20191001
# 03강 변수, 자료형, 변환
```

In [3]:

```
#####
값을 변수에 저장하기
#####

''' 대입문/할당문'''
x=1
x
print(x)
```

1

In [6]:

```
'''변수에 값 대입하기'''
x=1
print(x)
y=2
print(y)
x=x+3
print(x)
y=y+3
print(y)
z=x+y
print(z)
```

1
2
4
5
9

In [7]:

```
'''두 변수의 값 교체'''
x=3
y=4
print(x,y)
```

3 4

In [8]:

```
t=x
x=y
y=t
print(x,y)
```

4 3

In [9]:

```
'''다중 대입문'''
x,y=3,4
print(x,y)
x,y=y,x
print(x,y)
```

3 4
4 3

In [10]:

```
'''변수 활용'''

# 소,대문자 구분
varname=1
VarName=3
print(varname,VarName)

#문자열 포함 변수의 계산
s1='파이썬'
s2=' '
s3='프로그래밍'
s4=s1+s2+s3
print(s4)

#변수포함 계산식의 직접 적용
a=3
b=5
c=a*a+b*b
print(a,b,c)
```

1 3
파이썬 프로그래밍
3 5 34

In [13]:

```
"""
데이터 입력 받기
"""
```

Out[13]:

'\n 데이터 입력 받기\n\n'

In [21]:

```
'''사용자로부터 문자열 받기'''

# 학번, 이름 입력 받고 출력
univ_name=input('학교: ')
dept_name=input('학과: ')
stud_num=input('학번: ')
name=input('이름: ')
print('학교: ',univ_name, '학과: ',dept_name, '학번: ',stud_num, '이름: ',name)
print('학교:%s 학과:%s 학번:%s 이름:%s ' %(univ_name, dept_name, stud_num, name))

type(stud_num)
type(name)
```

학교: kmk
학과: physics
학번: 2014103332
이름: kmk
학교: kmk 학과: physics 학번: 2014103332 이름: kmk
학교:kmk 학과:physics 학번:2014103332 이름:kmk

Out[21]:

str

In [22]:

```
'''사용자로부터 정수 입력 받기'''
x=int(input('정수: '))
y=x+1
```

```
print(y)
```

정수: 9
10

In [23]:

```
#사용자로부터 2개의 정수를 입력받아 각각 변수 x와 y에 대입하고
#print() 함수를 이용하여 덧셈한 결과를 출력
x=int(input('정수1: '))
y=int(input('정수2: '))
print(x+y)

x,y=map(int, input('숫자 두 개를 입력하세요; ').split(','))
print('합은 ',x+y)
```

정수1: 98
정수2: 65
163
숫자 두 개를 입력하세요; 78,65
합은 143

In [25]:

```
#이름을 문자열로 대입하고 출생년도를 정수로 대입
name=input("이름:")
year=int(input("출생년도:"))
print("이름:",name,"출생년도:",year)
print("이름: %s 출생년도: %s" %(name, year))
```

이름: kmk
출생년도: 9876
이름: kmk 출생년도: 9876
이름: kmk 출생년도: 9876

In [26]:

```
#####
다양한 자료형으로 저장
#####
```

Out [26]:

```
'\n 다양한 자료형으로 저장\n\n'
```

In [27]:

```
'''대입에 의한 자료형 지정'''
#정수
x=1
print(type(x))

#실수
x=3.14
print(type(x))

#문자열
x="Text string"
print(type(x))
```

```
<class 'int'>
<class 'float'>
<class 'str'>
```

In [28]:

```
'''자료형 지정과 확인'''
```

#변수 x에 정수 값 1, 실수 값 3.14, 문자열 값 "Text string"을 순서대로 대입 저장 후,
#변수 x의 값과 자료형을 출력하시오.

```
x=1
print(x,type(x))
x=3.14
print(x,type(x))
x='Text String'
print(x,type(x))

#다른 자료형
bool_data=True
print(bool_data,type(bool_data))

list_data=[1,2,3]
print(list_data,type(list_data))

tuple_data=(1,2,3)
print(tuple_data,type(tuple_data))

set_data={2,3,1}
print(set_data,type(set_data))

dict_data={0:False, 1:True}
print(dict_data,type(dict_data))
```

```
1 <class 'int'>
3.14 <class 'float'>
Text String <class 'str'>
True <class 'bool'>
[1, 2, 3] <class 'list'>
(1, 2, 3) <class 'tuple'>
{1, 2, 3} <class 'set'>
{0: False, 1: True} <class 'dict'>
```

In [29]:

'''다른 자료형으로 변환하기'''

```
#형 변환
x1="3.14"
x2=float(x1)
x3=int(x2)
x4=str(x3)
x_list=[x1,x2,x3,x4]

for i in range(4):
    print(x_list[i],type(x_list[i]))

#문자열 입력 후 실수, 정수, 문자열 변환
text=input("실수 형태의 문자열: ")
fnum=float(text)
inum=int(fnum)
tstr=str(inum)
print(text,type(text), fnum,type(fnum), inum,type(inum), tstr,type(tstr))

x=3.14
y=int(x)
z=float(y)
print(x,y,z)

inum=3
fnum=float(inum)
print(inum,fnum)

fnum=3.14
inum=int(fnum)
print(fnum, inum)
```

```
3.14 <class 'str'>
3.14 <class 'float'>
3 <class 'int'>
3 <class 'str'>
실수 형태의 문자열: 7.8665
7.8665 <class 'str'> 7.8665 <class 'float'> 7 <class 'int'> 7 <class 'str'>
3.14 3 3.0
```

3 3.0
3.14 3

In [30]:

```
# 물리학과 2014103332 김민기 HW3_20191002  
# 04강 데이터 자료형 (시퀀스)
```

In [31]:

```
'''리스트'''  
  
a = [38, 21, 53, 62, 19]  
person = ['james', 17, 175.3, True]  
print('type(a)=', type(a))  
print('type(person)', type(person))  
  
a = []  
print(a)  
  
b = list()  
print(b)  
  
a = list(range(10))  
print(a)  
  
b = list(range(5, 12))  
print(b)  
  
c = list(range(-4, 10, 2))  
print(c)  
  
d = list(range(10, 0, -1))  
print(d)
```

```
type(a)= <class 'list'>  
type(person) <class 'list'>  
[]  
[]  
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]  
[5, 6, 7, 8, 9, 10, 11]  
[-4, -2, 0, 2, 4, 6, 8]  
[10, 9, 8, 7, 6, 5, 4, 3, 2, 1]
```

In [32]:

```
'''튜플'''  
  
a = (38, 21, 53, 62, 19)  
print('type of ', a, '=', type(a))  
b = 38, 21, 53, 62, 19  
print('type of ', b, '=', type(a))  
  
person = ('james', 17, 175.3, True)  
(38,)  
38,  
  
a = [1, 2, 3]  
tuple(a)  
b = (4, 5, 6)  
print(b)  
  
print(list(b))  
  
print(list('Hello'))  
print(tuple('Hello'))
```

```
type of (38, 21, 53, 62, 19) = <class 'tuple'>  
type of (38, 21, 53, 62, 19) = <class 'tuple'>  
(4, 5, 6)  
(4, 5, 6)
```

```
[4, 3, 0]
['H', 'e', 'l', 'l', 'o']
('H', 'e', 'l', 'l', 'o')
```

In [33]:

```
'''딕셔너리'''
```

```
lux = [490, 334, 550, 18.72]
lux = {'health': 490, 'mana': 334, 'melee': 550, 'armor': 18.72}
print(lux)
lux = {'health': 490, 'health': 800, 'mana': 334, 'melee': 550, 'armor': 18.72}
print(lux) # 중복되는 키는 가장 뒤에 있는 값을 불러옴
x = {100: 'hundred', False: 0, 3.5: [3.5, 3.5]}
print(x)
```

```
{'health': 490, 'mana': 334, 'melee': 550, 'armor': 18.72}
{'health': 800, 'mana': 334, 'melee': 550, 'armor': 18.72}
{100: 'hundred', False: 0, 3.5: [3.5, 3.5]}
```

In [34]:

```
'''딕셔너리 만들기'''
```

```
lux1 = dict(health=490, mana=334, melee=550, armor=18.72)
lux2 = dict(zip(['health', 'mana', 'melee', 'armor'], [490, 334, 550, 18.72]))
lux3 = dict([('health', 490), ('mana', 334), ('melee', 550), ('armor', 18.72)])
lux4 = dict({'health': 490, 'mana': 334, 'melee': 550, 'armor': 18.72})
print(lux1, '\n', lux2, '\n', lux3, '\n', lux4)
```

```
{'health': 490, 'mana': 334, 'melee': 550, 'armor': 18.72}
{'health': 490, 'mana': 334, 'melee': 550, 'armor': 18.72}
{'health': 490, 'mana': 334, 'melee': 550, 'armor': 18.72}
{'health': 490, 'mana': 334, 'melee': 550, 'armor': 18.72}
```

In [35]:

```
'''딕셔너리 참조'''
```

```
lux = {'health': 490, 'mana': 334, 'melee': 550, 'armor': 18.72}
print(lux['health'])
lux['health'] = 2037 #키값 수정
lux['mana'] = 1184
print(lux)
lux['mana_regen'] = 3.28
print('health' in lux)
print('attack_speed' not in lux)
print(len(lux))
```

```
490
{'health': 2037, 'mana': 1184, 'melee': 550, 'armor': 18.72}
True
True
5
```

In [36]:

```
'''시퀀스'''
```

```
a = [0, 10, 20, 30, 40, 50, 60, 70, 80, 90]
print(30 in a)

a = [0, 10, 20, 30]
b = [9, 8, 7, 6]
print(a + b) # union

#range(0, 10) + range(10, 20) range는 연산자로 연결 불가!
list(range(0, 10)) + list(range(10, 20))
tuple(range(0, 10)) + tuple(range(10, 20))

'Hello, ' + 'world!'
10 + 10 + 20 + 30 + 40
```

```
[0, 10, 20, 30] * 3
#range(0, 5, 2) * 3 range는 연산자로 연결 불가!
print(list(range(0, 5, 2)) * 3)
print(tuple(range(0, 5, 2)) * 3)
'Hello, ' * 3
```

```
True
[0, 10, 20, 30, 9, 8, 7, 6]
[0, 2, 4, 0, 2, 4, 0, 2, 4]
(0, 2, 4, 0, 2, 4, 0, 2, 4)
```

```
Out[36]:
'Hello, Hello, Hello, '
```

In [37]:

```
'''시퀀스 활용'''

a = [0, 10, 20, 30, 40, 50, 60, 70, 80, 90]
print(len(a))

b = (38, 76, 43, 62, 19)
print(len(b))

print(len(range(0, 10, 2)))

hello = 'Hello, world!'
print(len(hello))

a = [38, 21, 53, 62, 19]
print(a[0])

b = (38, 21, 53, 62, 19)
print(b[0])

r = range(0, 10, 2)
r[2]

print(a[-1])
#a[5] 시퀀스 길이보다 크면 안나옴!
print(a[len(a) - 1])

print(a)
del a[2] #2+(1) 번째 원소 삭제
print(a)
```

```
10
5
5
13
38
38
19
19
[38, 21, 53, 62, 19]
[38, 21, 62, 19]
```

In [38]:

```
'''시퀀스 슬라이스'''

a=[0,10,20,30,40,50,60,70,80,90]
print(a)

print(a[0:4])
print(a[1:1])
print(a[1:2])
print(a[4:-1])
print(a[2:8:3])
print(a[:7])
print(a[7:])
print(a[:])
print(a[:7:2])
```

```

print(a[7::2])
print(a[:,2])
print(a[:,])
print(a[5:1:-1])
print(a[0:len(a)])

a[2:5] = ['a', 'b', 'c'] #element 수정
print(a)
a[2:5] = ['a']
print(a)
a[2:5] = ['a', 'b', 'c', 'd', 'e']
print(a)
a[2:8:2] = ['a', 'b', 'c']
print(a)
del a[2:5]
print(a)
del a[2:8:2]

[0, 10, 20, 30, 40, 50, 60, 70, 80, 90]
[0, 10, 20, 30]
[]
[10]
[40, 50, 60, 70, 80]
[20, 50]
[0, 10, 20, 30, 40, 50, 60]
[70, 80, 90]
[0, 10, 20, 30, 40, 50, 60, 70, 80, 90]
[0, 20, 40, 60]
[70, 90]
[0, 20, 40, 60, 80]
[0, 10, 20, 30, 40, 50, 60, 70, 80, 90]
[50, 40, 30, 20]
[0, 10, 20, 30, 40, 50, 60, 70, 80, 90]
[0, 10, 'a', 'b', 'c', 50, 60, 70, 80, 90]
[0, 10, 'a', 50, 60, 70, 80, 90]
[0, 10, 'a', 'b', 'c', 'd', 'e', 70, 80, 90]
[0, 10, 'a', 'b', 'b', 'd', 'c', 70, 80, 90]
[0, 10, 'd', 'c', 70, 80, 90]

```

In [39]:

```
# 05강 데이터 계산하기
```

In [40]:

```

'''수식과 연산자'''

'''
사용자로부터 2개의 정수를 입력받아 각각 변수 x와 y에 대입하고 print()
함수를 이용해서 사칙연산의 결과를 출력
'''
x,y=map(int,input("정수1, 정수2: ").split(','))

print(x+y)
print(x-y)
print(x*y)
print(x/y)

'''
나이 계산
'''
import datetime
current_year=datetime.datetime.today().year

name = input('이름 : ')
birth_year = int(input('출생년도 : '))
age = current_year - birth_year + 1
print('이름 : ', name)
print('출생년도 : ', birth_year)
print('나이 : ', age)

'''
10원 이상의 거스름돈에 해당하는 정수를 입력받아 줘야하는
500원, 100원, 50원, 10원 동전의 개수를 구하기
'''

```



```
'''
x = int(input('거스름돈 금액 : '))
x500 = x // 500
temp = x % 500
x100 = temp // 100
temp = temp % 100
x50 = temp // 50
temp = temp % 50
x10 = temp // 10
print("거스름돈:", x, '원', "500원:", x500, "100원:", x100, "50원:", x50, "10원:", x10)
```

정수1, 정수2: 89, 76

165

13

6764

1.1710526315789473

이름 : kmk

출생년도 : 8769

이름 : kmk

출생년도 : 8769

나이 : -6749

거스름돈 금액 : 5000

거스름돈: 5000 원 500원: 10 100원: 0 50원: 0 10원: 0

In [41]:

```
'''복합 대입 연산자'''

'''
정수를 입력받아 변수 x에 대입 후 사칙연산을 위한 복합 대입 연산자를 활용하여
변수 x에 각각 2를 계산하고 출력
'''
x = int(input('정수 : '))
print(x)
x += 2
print(x)
x -= 2
print(x)
x *= 2
print(x)
x /= 2
print(x)

'''
복합 대입 연산자 비교
'''
a = 1
b = 2
c = 3
a = a + b
b = b + a - c
c = c * (a - b)
print(a, b, c)

a = 1
b = 2
c = 3
a += b
b += (a - c)
c *= (a - b)
print(a, b, c)
```

정수 : 89

89

91

89

178

89.0

3 2 3

3 2 3

In [42]:

```
'''연산자 우선순위'''
```

```
'''
두 정수를 입력 받아 평균값을 구해서 출력
'''
```

```
x = int(input('값1 : '))
y = int(input('값2 : '))
z = (x + y) / 2
print("평균 :", z)
z1 = x + y / 2
print("평균(사실 아님) :", z1)
```

```
값1 : 78
값2 : 65
평균 : 71.5
평균(사실 아님) : 110.5
```

In [43]:

```
# 물리학과 2014103332 김민기 HW4_20191010

# 06강 알고리즘 사고와 선택
```

In [44]:

```
'''관계연산자'''
```

```
x = 6
y = 2
print(x, '>', y, x > y)
print(x, '>=', y, x >= y)
print(x, '<', y, x < y)
print(x, '<=', y, x <= y)
print(x, '==', y, x == y)
print(x, '!=', y, x != y)

x = y
print('x = ', x, 'y = ', y)
```

```
6 > 2 True
6 >= 2 True
6 < 2 False
6 <= 2 False
6 == 2 False
6 != 2 True
x = 2 y = 2
```

In [46]:

```
'''논리연산자 실습'''
```

```
k = 90
e = 85
print(k >= 80 and e >= 80)
print(k >= 80 or e >= 80)
k, e = map(int, input("두개의 숫자를 입력하세요: ").split())
print(k >= 80 and e >= 80)
k, e = map(int, input("두개의 숫자를 입력하세요: ").split())
print(k >= 80 or e >= 80)
```

```
True
True
두개의 숫자를 입력하세요: 67 56
False
두개의 숫자를 입력하세요: 98 65
True
```

In [47]:

```
'''if 문 연습'''
```

Out[47]:

'if 문 연습'

In [48]:

```
'''
score의 값이 65이고, score가 60이상이면 'pass'출력
'''

score = 65
if score >= 60:
    print("pass")
```

pass

In [49]:

```
'''
두 수를 입력받아 a,b에 대입한다. If 문의 a<b가 참이면
'b가 크다'출력. 아니면 '어떤 수가 큰지 판단함'
'''

a, b = map(int,input("두개의 숫자를 입력하세요: ").split())
if a < b:
    print(b, '가 크다')
print('어떤 수가 큰지 판단함')
```

두개의 숫자를 입력하세요: 56 76
76 가 크다
어떤 수가 큰지 판단함

In [50]:

```
'''
정수를 입력 받아 num에 대입한 수 양수, 0, 음수를 판단하기 위해
세개의 if문을 사용
'''

num = int(input("정수를 입력하세요: "))
if num > 0:
    print('양수')
if num == 0:
    print('0')
if num < 0:
    print('음수')
```

정수를 입력하세요: 65
양수

In [51]:

```
'''
if-else문 연습
'''
```

Out[51]:

'\nif-else문 연습\n'

In [52]:

```
'''
두 수를 입력받아 kor_score와 eng_score의 변수에 저장하고 값이 모두
80 이상인 경우 '합격입니다.' 출력, 아니면 '불합격입니다.'
'''

kor_score, eng_score = map(int,input("국어, 영어 성적을 입력하세요: ").split())
if kor_score >= 80 and eng_score >= 80:
    print('합격입니다.')
```

```

        print('합격입니다.')
    else:
        print('불합격입니다.')

kor_score, eng_score = map(int, input("국어, 영어 성적을 입력하세요: ").split())
if kor_score < 80 or eng_score < 80:
    print('불합격입니다.')
else:
    print('합격입니다.')

```

국어, 영어 성적을 입력하세요: 80 86
합격입니다.
국어, 영어 성적을 입력하세요: 76 98
불합격입니다.

In [53]:

```

'''
정수를 입력 받아 num 저장 후 짝수, 홀수 출력
'''

num = int(input("정수 :"))
if num%2 == 0:    #2로 나눈 나머지
    print("짝수")
else:
    print("홀수")

```

정수 :78
짝수

In [54]:

```

'''
if-elif-else문 실습
'''

```

Out[54]:

'\nif-elif-else문 실습\n'

In [55]:

```

'''
점수를 입력받아 점수가 90점 이상이면 A, 80점 이상이면 B, 70점 이상이면 C
60점 이상이면 D 60점 미만이면 F 출력
'''

score = int(input('점수 : '))
if score >= 90:
    print("A")
elif score >= 80:
    print("B")
elif score >= 70:
    print("C")
elif score >= 60:
    print("D")
else:
    print("F")

```

점수 : 78
C

In [56]:

```

'''
양수, 음수, 0 구분출력
'''

num = int(input("정수 :"))
if num == 0:
    print("0")

```

```
elif num > 0:
    print("양수")
else:
    print("음수")
```

정수 :98
양수

In [57]:

```
# 물리학과 2014103332 김민기 HW5_20191016
# 07강 알고리즘 사고와 반복
```

In [58]:

```
'''
횟수만큼 반복
'''
for i in [1,2,3,4,5]:
    print('파이썬')

for i in [1,2,3,4,5]:
    print('파이썬',i)

for i in range(5):
    print(i, end= ' ')
```

파이썬
파이썬
파이썬
파이썬
파이썬
파이썬 1
파이썬 2
파이썬 3
파이썬 4
파이썬 5
0 1 2 3 4

In [59]:

```
'''
for 문 실습
list/range()를 이용해서 1부터 5까지 정수의 합계를 구하시오
'''
#1
s = 0
for i in [1,2,3,4,5]:
    s += i
print('s: ',s)

#2
s = 0
print(s)
for i in [1,2,3,4,5]:
    s += i
    print('i: ', i, 's: ',s)
print('s :', s)

#3
s=0
print(s)
for i in [5, 4, 3, 2, 1]:
    s = s + i
print('s :', s)

#4
s=0
print(s)
for i in range(1, 6):
    s = s + i
    print('i :', i, ' s :', s)
print('s :', s)
```

```
s: 15
0
i: 1 s: 1
i: 2 s: 3
i: 3 s: 6
i: 4 s: 10
i: 5 s: 15
s : 15
0
s : 15
0
i : 1 , s : 1
i : 2 , s : 3
i : 3 , s : 6
i : 4 , s : 10
i : 5 , s : 15
s : 15
```

In [60]:

```
'''
1 부터 10까지 홀수의 합
'''
s=0
print(s)
for i in range(1, 11, 2):
    s = s + i
    print('i :', i, ', s :', s)
print('s :', s)
```

```
0
i : 1 , s : 1
i : 3 , s : 4
i : 5 , s : 9
i : 7 , s : 16
i : 9 , s : 25
s : 25
```

In [61]:

```
'''
while문 실습
'''
```

Out[61]:

```
'\nwhile문 실습\n'
```

In [62]:

```
'''
1부터 5까지 출력
'''
i = 1
while i <= 5:
    print(i, end=' ')
    i += 1
print(' ')
print(i)
# while 문 전에 초기조건 설정
# 조건식이 참이면 수행/ 아니면 x
```

```
1 2 3 4 5
6
```

In [63]:

```
'''
1부터 시작하여 루프 제어변수 n값이 10보다 작거나 같을 때까지 반복하면서
숫자가 짝수인 경우 2를 곱하고 홀수인 경우 1을 더함
'''
```

```
'''
숫자 n과 n의 합 s를 1씩 (n, s)을 세기 반복하여 구함
'''
```

```
n = 1
s = 0
while n <= 10:
    if n%2 == 0:
        s += n
        print('n: ', n, 's: ', s)
    n += 1
print('n: ', n)
print('s: ', s)
```

```
n: 2 s: 2
n: 4 s: 6
n: 6 s: 12
n: 8 s: 20
n: 10 s: 30
n: 11
s: 30
```

In [64]:

```
'''
while문을 이용하여 9부터 1까지 반복하면서 홀수의 합을 구함
(단, 루프 제어 변수 n은 -2씩 변화함)
'''
```

```
n = 9
s = 0
while n >= 1:
    if n%2 == 1:
        s += n
        print('n: ', n, 's: ', s)
    n += -2
print('n: ', n)
print('s: ', s)

print("_____")
```

```
n: 9 s: 9
n: 7 s: 16
n: 5 s: 21
n: 3 s: 24
n: 1 s: 25
n: -1
s: 25
```

In [65]:

```
'''
루프 제어(루프 탈출과 계속 반복)
'''

#loop 제어
for i in [1,2,3,4,5]:
    if i == 4:
        break
    if i%2 == 0:
        continue #짝수를 안보여줌
    print(i, end=' ')

print(' ')
print(i)
```

```
1 3
4
```

In [66]:

```
'''
for 문을 이용하여 1부터 20까지 반복하면서 홀수인 경우
continue를 사용하여 반복을 계속 진행하고, 짝수인 경우만 합계를 구함.
다 하계가 30을 넘어 가는 바보의 조그한 바보의 지혜치면서
```

```
'''
단, 합계가 30을 넘을 경우, 반복을 종료함. 반복을 진행하면서
반복 진행 횟수와 합계를 계속 표시하고, 반복이 종료된 후에도 표시함.
'''
```

```
s = 0
for i in range(1,21):
    if i%2 == 1:
        continue
    s += i
    print('i: ',i, 's: ',s)
    if s > 30:
        break
print('i: ',i, 's: ',s)
```

```
i:  2 s:  2
i:  4 s:  6
i:  6 s: 12
i:  8 s: 20
i: 10 s: 30
i: 12 s: 42
i: 12 s: 42
```

In [67]:

```
'''
for 문을 이용하여 1부터 20까지 반복하면서 짝수인 경우
continue를 사용하여 반복을 계속 진행하고, 홀수인 경우만 합계를 구함.
단, 합계가 30을 넘을 경우, 반복을 종료함. 반복을 진행하면서
반복 진행 횟수와 합계를 계속 표시하고, 반복이 종료된 후에도 표시함.
'''
```

```
s = 0
for i in range(1,21):
    if i%2 == 0:
        continue
    s += i
    print('i: ',i, 's: ',s)
    if s > 30:
        break
print('i: ',i, 's: ',s)
```

```
i:  1 s:  1
i:  3 s:  4
i:  5 s:  9
i:  7 s: 16
i:  9 s: 25
i: 11 s: 36
i: 11 s: 36
```

In [68]:

```
# 함수
```

In [69]:

```
'''
다음 동작을 순서대로 작성하시오.
1 print() 함수를 이용하여 순서대로 학번과 성명을 2회 출력하시오.
2 for 문을 이용하여 학번과 성명을 2회 출력하시오.
3 학번과 성명을 출력하는 sn() 함수를 만든 후 sn() 함수를 2회 호출하시오.
4 for 문을 이용하여 sn() 함수를 2회 호출하시오.
'''
```

```
#1
print("12345678")
print("홍길동")
print("12345678")
print("홍길동")
print("")

#2
for i in range(2):
    print("12345678")
    print("홍길동")
print("")
```



```
#3
def sn():
    print("12345678")
    print("홍길동")

sn()
sn()
print('')

#4
for i in range(2):
    sn()
print(' ')
```

```
12345678
홍길동
12345678
홍길동
```

```
12345678
홍길동
12345678
홍길동
```

```
12345678
홍길동
12345678
홍길동
```

```
12345678
홍길동
12345678
홍길동
```

In [71]:

```
'''
1부터 9까지 출력하는 함수 print19()를 for문을 이용하여 작성하여 2회 호출하시오.
'''
def print19():
    for i in range(1,10):
        print(i, end=' ')
    print(' ')

for i in range(2):
    print19()
```

```
1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9
```

In [72]:

```
def fadd(n,m):
    s=n+m
    return print(n,"+",m,"=",s)

fadd(4,5)
```

```
4 + 5 = 9
```

In [74]:

```
def fsubt(n,m):
    s=n-m
    print(n,"-",m,"=",s)

fsubt(6,10)
```

```
6 - 10 = -4
```

In [75]:

```
def calc_gugudan(dan):  
    for i in range(1,10):  
        print("%d * %d = %d" %(dan,i,dan*i))  
  
d = int(input("단: "))  
  
if d > 9:  
    print("단은 1~9까지만 입력하세요")  
else:  
    calc_gugudan(d)
```

```
단: 8  
8 * 1 = 8  
8 * 2 = 16  
8 * 3 = 24  
8 * 4 = 32  
8 * 5 = 40  
8 * 6 = 48  
8 * 7 = 56  
8 * 8 = 64  
8 * 9 = 72
```

In [76]:

```
def calc_gugudan_ver2(dan):  
    if float(dan) != int(dan):  
        print("단은 자연수만 입력하세요")  
    elif dan > 9 or dan <= 0:  
        print("단은 1~9까지만 입력하세요")  
    else:  
        for i in range(1,10):  
            print("%d * %d = %d" %(dan,i,dan*i))
```

In [77]:

```
calc_gugudan_ver2(45)
```

단은 1~9까지만 입력하세요

In [80]:

```
calc_gugudan_ver2(-5)
```

단은 1~9까지만 입력하세요

In [78]:

```
calc_gugudan_ver2(4.4)
```

단은 자연수만 입력하세요

In [79]:

```
calc_gugudan_ver2(4)
```

```
4 * 1 = 4  
4 * 2 = 8  
4 * 3 = 12  
4 * 4 = 16  
4 * 5 = 20  
4 * 6 = 24  
4 * 7 = 28  
4 * 8 = 32  
4 * 9 = 36
```

In [81]:

```
'''
시작에 해당하는 정수를 입력 받아 변수 s에 대입하고, 끝에 해당하는 정수를
입력받아 변수 e에 대입한 후 for문을 이용하여 start부터 end까지
출력하는 함수print19(start, end)를 작성하고 호출해 보자.
(단, 입력한 변수 s의 값이 변수 e의 값 보다 작을 때만 함수를 호출함)
'''

def print19(st, ed):
    if st >= ed:
        print("끝값이 더 커야함")
    elif int(st) != float(st) or int(ed) != float(ed):
        print("정수만 써야함")
    else:
        for i in range(st,ed+1):
            print(i, end=' ')
        print(" ")
```

In [82]:

```
print19(3,5)
```

3 4 5

In [83]:

```
print19(3,5.5)
```

정수만 써야함

In [84]:

```
print19(34,5.5)
```

끝값이 더 커야함

In [85]:

```
def add(n,m):
    s=n+m
    return s,n

a=3
b=4
r=add(a,b)[0]
print("반환값 = %f" %(r))
```

반환값 = 7.000000

In [86]:

```
add(3,5)
```

Out[86]:

(8, 3)

In [87]:

```
'''
사용자로부터 입력 받은 두 수의 평균값을 구하여 반환하는
avg() 함수를 만들고 호출. 또한 avg() 함수의 결과를 반환 받아 출력
'''

def avg(num1, num2):
    mean=(num1+num2)/2
    return mean
```

```
int1=int(input('값1: '))
int2=int(input('값2: '))
r=avg(int1, int2)
print("평균=%f" %(r))
```

값1: 45
값2: 234
평균=139.500000

In [88]:

```
'''
입력 값이 3개일 때의 평균을 구하는 avg() 함수로 변경하여 평균 계산을 수행
'''
def avg3(num1, num2, num3):
    mean=(num1+num2+num3)/3
    return mean

int1=int(input('값1: '))
int2=int(input('값2: '))
int3=int(input('값3: '))
r=avg3(int1, int2, int3)
print("평균=%f" %(r))
```

값1: 45
값2: 34
값3: 65
평균=48.000000