

Python 기초

1. 문제

다음 프로그램의 결과를 예상하시오

```
print("%d/%d=%d...%d (%.2f)" %(10,3,10/3,10/3))
print("%d/%d=%d...%d (%.2f)" %(10,3,10//3,10/3))
print("%d/%d=%f...%d (%.2f)" %(10,3,10/3,10/3))
```

```
10/3=3...1 (3.33)
10/3=3...1 (3.33)
10/3=3.333333...1 (3.33)
```

2. 문제

나머지와 몫을 한번에 구하는 내장함수를 이용해서 프로그램을 완성하세요

```
print("%d/%d=%d...%d" %(10,3,_____))
```

```
10/3=3...1
```

3. 문제

sss = '10*3+2' 과 같이 연산식이 주어진 경우 연산결과를 아래와 같이 출력하고자 한다. 프로그램을 완성하시오.

```
sss = '10*3+2'
```

```
print(_____)
```

```
10*3+2=32
```

4. 문제

다음과 같은 에러를 만났다. 무슨 의미인지 프로그램의 잘못된 곳을 수정해 보시오.

```
str = '10*3+2'
```

```
print(str + '=' + str(eval(str)))
```

```
----- Error Info -----
```

```
Traceback (most recent call last):
  File "binarytest12.py", line 3, in <module>
    print(str + '=' + str(eval(str)))
TypeError: 'str' object is not callable
```

5. 문제

다음 프로그램의 결과를 예상하시오

```
print("%05d" % 65)

print("%10s" % 'pythonWn'*3)
```

6. 문제

다음 프로그램의 결과를 예상하시오

```
print(hex(65),oct(65),bin(65))

print(bin(int("%d" % 11,10)))

print(bin(int("%d" % 11,2)))

print(bin(int("%d" % 11,8)))
```

7. 문제

다음 프로그램의 결과를 예상하시오

```
print(format(int("%d" % 65),'b'))

print('123'.zfill(8))

print(format(int("%d" % 65),'b').zfill(8))

print(format(int("%d" % 65),'x').zfill(8))
```

8. 문제

다음 프로그램의 결과를 예상하시오

```
aaa = 'Fe'

kkk = _____

if _____
```

```
else:
```

```
print('입력값이 16진수가 아닙니다.')
```

254

9. 문제

파이썬을 구동시키기 위해서는 Python.exe 파일이 필요하다. Path 설정방법은 어떻게 하는가?

10. 문제

원의 반지름을 입력받아서 넓이를 구하는 프로그램을 작성해보자.(input 함수이용)

```
import math
```

```
rr = _____
```

```
area = _____
```

```
print(area)
```

```
c:\temp\W7YyM7J207I2s6riv66eQ6rOg7IKs>python binarytest12.py
radius :2
25.132741228718345
```

```
c:\temp\W7YyM7J207I2s6riv66eQ6rOg7IKs>
```

11. 문제

MyMain.py 을 동작시키기 위한 콘솔 명령을 작성하시오 (Dos 명령)

```
c:\temp\W기초프로그래밍>_____
```

Python 연산자

12. 문제

다음 프로그램의 결과를 예상하시오

```
print(3+2**2*3)
```

13. 문제

다음 프로그램의 결과를 예상하시오

```
aaa = 3
```

```
print(~aaa)
```

```
print(3<<2)
```

```
print(not aaa)
```

14. 문제

2.1²을 구하는 방법 2가지 모두 사용한 프로그램을 완성하시오. (외부 모듈을 사용하는 코드를 함께 작성하시오)

15. 문제

다음의 두 변수의 덧셈 값을 구하는 프로그램의 결과를 완성하시오.

```
aaa ='111.11'
```

```
bbb ='12.11'
```

```
print(_____)
```

16. 문제

다음 프로그램의 결과를 예상하시오

```
aaa = 0xFF00
```

```
bbb = 0x00FF
```

```
print(aaa | bbb) #비트연산
```

17.문제

다음 프로그램의 결과를 예상하시오

```
aaa = 'FF00'
```

```
bbb = '00FF'
```

```
print(int(aaa,16) | int(bbb,16)) #비트연산
```

18.문제

다음 프로그램의 결과를 예상하시오

```
aaa = '0b11111111'
```

```
bbb = '0x0F'
```

```
print(int(aaa,2) & int(bbb,16)) #비트연산 *****
```

```
print(int(aaa,2) ^ int(bbb,16))
```

제어문

19.문제

다음 프로그램의 결과를 예상하시오

```
arr = ""
```

```
for ii in range(1,3,2):
```

```
    for kk in range(5):
```

```
        arr += str(kk)
```

```
print(ii, kk, arr)
```

20.문제

구구단 9단을 거꾸로 출력하는 프로그램을 작성하시오.

```
9 x 9 = 81
```

```
9 x 8 = 72
```

```
9 x 7 = 63
```

```
9 x 6 = 54
```

```
9 x 5 = 45
```

```
9 x 4 = 36
```

```
9 x 3 = 27
```

```
9 x 2 = 18
```

21.문제

1부터 100까지 자연수 중에서 2의 배수의 합을 구하시오.

22.문제

1부터 100까지 자연수 중에서 2의 배수의 합을 구하시오. (if 문 사용)

23.문제

1부터 100 사이의 홀수의 누적 합계가 최초로 50이 넘는 위치와 합을 출력하는 프로그램을 작성하시오.

```
15 번째까지의 누적 값 64
```

배열 자료형

24.문제

복수의 자료를 담을 수 있는 데이터타입 중 리스트와 튜플 그리고 array (numpy) 총 3가지 자료형의 차이점을 예를 들어 설명하시오.

25.문제

다음과 같은 튜플 자료의 최대값 최소값을 구하는 프로그램을 작성하시오.(반복문 사용할 것)

```
var = (30,20,10,15,90,95,40)
```

```
print(max,min)
```

26.문제

다음과 같은 튜플 자료의 최대값 최소값을 구하는 프로그램을 작성하시오.(내장함수 사용)

```
var = (30,20,10,15,90,95,40)
```

```
print(max,min)
```

27.문제

다음과 같은 튜플 자료의 평균과 합을 구하는 프로그램을 작성하시오.

```
var = (30,20,10,15,90,95,40)
```

```
hap=0; avg=0
```

```
for kk in var:
```

```
    hap += kk
```

```
avg = hap/len(var)
```

```
print(hap,avg)
```

28.문제

다음과 같은 튜플 자료의 중간값을 구하는 프로그램을 작성하시오.

```
var = (30,20,10,15,90,95,40)
```

```
30
```

29.문제

random 모듈이 가지고 있는 함수 정보를 아래와 같이 출력해 보고 싶다. 프로그램을 완성하시오.

```
import random
```

```
print(_____)
```

```
[ 'BPF', 'LOG4', 'NV_MAGICCONST', 'RECIP_BPF', 'Random', 'SG_MAGICCONST',
'SystemRandom', 'TWOPI', '_BuiltinMethodType', '_MethodType', '_Sequence', '_Set', '__all__',
'__builtins__', '__cached__', '__doc__', '__file__', '__loader__', '__name__', '__package__',
'__spec__', '_acos', '_ceil', '_cos', '_e', '_exp', '_inst', '_log', '_pi', '_random', '_sha512', '_sin',
'_sqrt', '_test', '_test_generator', '_urandom', '_warn', 'betavariate', 'choice', 'expovariate',
'gammavariate', 'gauss', 'getrandbits', 'getstate', 'lognormvariate', 'normalvariate',
'paretovariate', 'randint', 'random', 'randrange', 'sample', 'seed', 'setstate', 'shuffle', 'triangular',
'uniform', 'vonmisesvariate', 'weibullvariate']
```

30.문제

1~10까지중 5개의 랜덤값을 포함하는 리스트를 생성하고 평균, 표준편차를 구하는 프로그램을 작성하시오. (numpy array 사용 x)

31.문제

1~10까지중 5개의 랜덤값을 포함하는 리스트를 생성하고 평균, 표준편차를 구하는 프로그램을 작성하시오. (numpy array 사용 o)

32.문제

45°에 해당되는 sin 값을 출력하는 프로그램을 작성하시오.

33.문제

다음과 같은 극좌표계의 각 40°, 반지름 10의 값을 직각좌표계의 xy 값으로 출력하는 프로그램을 작성하시오. (함수명은 Cart2Polar()을 사용한다.)

단, 함수의 입력은 각도와 반지름값을 가지는 딕셔너리 자료형을 사용하도록 한다.

```
print(pt1,Cart2Polar(pt1))
```

```
{'theta': 40, 'r': 10} (7.66044443118978, 6.4278760968653925)
```

34.문제

다음과 같은 리스트 자료형의 출력결과를 아래와 같이 나타내기 위한 방법 2가지를 각각 작성하

시오.

```
lt = [(1,'a'),(2,'b'),(3,'c'),(4,'d'),(5,'e')];
```

```
a
b
c
d
e
```

35.문제

1부터 100까지 자연수 중에서 2의 배수의 합을 구하시오. (리스트 내부 제어문을 사용하시오.)

```
lt2 = [ _____ ]
```

```
print(lt2);
```

36.문제

다음과 같이 튜플 형태의 xy 좌표값 쌍을 3개 가지는 리스트가 존재한다. x값과 y값을 분리해서 각각 리스트를 생성하고 각각 화면에 출력해보자.

```
pt = [(3,5),(1,-1),(8,9)]
```

```
print(pt)
```

```
xx = []; yy = []
```

```
print(xx,yy)
```

```
[(3, 4), (1, 2), (8, 9)]
[3, 1, 8] [4, 2, 9]
```

함수 생성

37.문제

1부터 100까지 자연수 중에서 2의 배수의 합을 구하는 함수 mysum() 을 작성하시오 실행결과와 다음과 같다.

- 함수명 mysum()

```
print(mysum((1,101,2))) #시작,끝,배수값
```

```
2550
```

38.문제

동전의 총 합을 입력 받고 500,100,50원 동전의 개수를 출력하는 함수를 작성하라.

- 함수명 calCoin()

```
print(calCoin(7850));
```

```
(15, 3, 1)
```

39.문제

다음과 같은 극좌표계의 각 40°, 반지름 10 의 값을 직각좌표계의 zy 값으로 출력하는 프로그램을 작성하시오.

- 함수명은 Cart2Polar()

```
pt1 = (40,10)
```

```
print(pt1, Cart2Polar(pt1))
```

```
(40, 10) (7.66044443118978, 6.4278760968653925)
```

40.문제

다음과 같은 극좌표계의 각 40°, 반지름 10 의 값을 직각좌표계의 zy 값으로 출력하는 프로그램을 작성하시오. (함수명은 Cart2Polar() 을 사용한다.)

단, 함수의 입력은 각도와 반지름값을 가지는 딕셔너리 자료형을 사용하도록 한다.

```
print(pt1, Cart2Polar(pt1))
```

```
{'theta': 40, 'rrr': 10} (7.66044443118978, 6.4278760968653925)
```

41.문제

원기둥의 표면적의 넓이를 구하는 함수를 작성하시오

- 함수명 calCylinderArea()

```
import math
```

```
area = calCylinderArea(10,4)
```

```
print(area)
```

```
879.645943005142
```

42.문제

다음 요구사항을 만족하는 함수(Represent)를 작성하시오.

Represent() 함수는 16,8,2,10진수 부호 있는 값을 입력 받고 8비트 2진수 값을 문자열형태로 출력하도록 한다.

입력 값이 음수가 입력되는 경우 2의 보수 값 연산 결과를 보이도록 한다.

※ 입력 값이 실수인 경우는 포함되지 않습니다.

```
def adder(pre,carr):

    if pre == '1' and carr == '1':

        return ('0','1')

    elif pre == '1' and carr == '0':

        return ('1','0')

    elif pre == '0' and carr == '1':

        return ('1','0')

    else:

        return ('0','0')

def twoscomplement(var):

    carr = '1';re =''

    for ii in reversed(var):

        kk, carr = adder(ii,carr)

        re = kk + re

    return re

def represent(var):

    kkk = ""

    if var<0:

        vvv = format(var,'b').replace('-','').zfill(8)
```

```
        for ii in vvv:

            if ii == '1': kkk = kkk + '0'

            else: kkk = kkk + '1'

        vvv = twoscomplement(kkk)

    else:

        vvv = format(var,'b').zfill(8)

    return vvv

xxx = -65

print(xxx,'->',represent(xxx))

xxx = 65

print(xxx,'->',represent(xxx))

xxx = 0b10

print(xxx,'->',represent(xxx))

xxx = -0b10

print(xxx,'->',represent(xxx))

xxx = 0o11

print(xxx,'->',represent(xxx))

xxx = -0o11

print(xxx,'->',represent(xxx))

xxx = 0x65

print(xxx,'->',represent(xxx))

xxx = -0x65

print(xxx,'->',represent(xxx))
```

```
-65 -> 10111111
65 -> 01000001
2 -> 00000010
-2 -> 11111110
```

모듈화

43.문제

MyModule.py 라는 파일 이름으로 calCoin() 라는 함수를 작성했다.

새로운 파일 MyMain.py 에서 calCoin() 함수를 호출하는 방법 4가지를 기술하시오.

44.문제

CalCoin 함수를 MyModule.py 파일에 저장했다. 이후 MyTest.py 라는 파일을 만들고 앞서 만든 CalCoin 함수를 호출해서 1750원에 대한 동전의 개수를 구하는 프로그램을 작성하시오 (외부 모듈 호출 방법4가지 각각 작성)

데이터 처리

45.문제

아래 프로그램은 0에서 1사이의 5개의 랜덤값을 만들고, 각 요소에 10을 곱한 결과이다. 결과를 설명해보자.

```
import numpy

arr = numpy.random.rand(5)

print(arr*10)

print('-'*10)

print(list(arr)*10)
```

```
[ 0.94729697  8.43991443  2.80519038  0.97332763  5.3775889 ]
-----
[0.094729696940663688, 0.84399144341952181, 0.28051903774897502,
0.097332762583655863, 0.53775888955080731, 0.094729696940663688,
0.84399144341952181, 0.28051903774897502, 0.097332762583655863,
0.53775888955080731, 0.094729696940663688, 0.84399144341952181,
0.28051903774897502, 0.097332762583655863, 0.53775888955080731,
0.094729696940663688, 0.84399144341952181, 0.28051903774897502,
0.097332762583655863, 0.53775888955080731, 0.094729696940663688,
0.84399144341952181, 0.28051903774897502, 0.097332762583655863,
0.53775888955080731, 0.094729696940663688, 0.84399144341952181,
0.28051903774897502, 0.097332762583655863, 0.53775888955080731,
0.094729696940663688, 0.84399144341952181, 0.28051903774897502,
0.097332762583655863, 0.53775888955080731]
```

46.문제

0에서 1사이의 5개의 랜덤값을 만들고, 각 요소에 10을 곱한 결과값을 가지는 배열값을 출력하는 프로그램을 완성하시오.

```
import numpy

arr = numpy.random.rand(5)

print(arr)

print(_____)
```

```
[ 0.63723571  0.23269183  0.50968944  0.66869784  0.4874024 ]
[ 6.  2.  5.  7.  5.]
```

47.문제

0에서 1사이의 5개의 랜덤값을 만들고, 각 요소에 10을 곱한 결과값을 리스트로 출력하는 함수를 만드시오.

- 함수이름은 myNorm()

```
import numpy
```



```
arr = numpy.random.rand(5)

print(arr)

print(myNorm(arr))
```

```
[ 0.22447398  0.27289982  0.08869783  0.41707719  0.52125695]
[2, 2, 0, 4, 5]
```

48.문제

0에서 1사이의 5개의 랜덤값을 만들고, 각 요소에 10을 곱한 결과값을 리스트로 출력하는 함수 myNorm() 을 만들었다. 이 함수를 배열의 각 요소에 적용시키는 프로그램을 완성하시오.

```
import numpy

arr = numpy.random.rand(5)

def myNorm2(var): return int(var*10)

print(arr)

print(_____)
```

```
[ 0.08500594  0.00846604  0.42785998  0.888863   0.48793253]
[0, 0, 4, 8, 4]
```

49.문제

0에서 1사이의 5개의 랜덤값을 만들고, 각 요소에 10을 곱한 결과값을 리스트로 출력하는 함수 lambda 함수를 만들어서 배열의 각 요소에 적용시키는 프로그램을 완성하시오.

```
import numpy

arr = numpy.random.rand(5)

print(arr)
```

```
print(_____)
```

```
[ 0.8785484  0.96197539  0.17014506  0.05366219  0.27361521]
[8, 9, 1, 0, 2]
```

50.문제

0에서 1사이의 5개의 랜덤값을 만들고, 각 요소에 10을 곱한 결과값을 반올림 없이 절사한 결과를 출력한다.

또한 이중에서 짝수인 결과값만을 가지는 리스트를 새로 만들어 보자.

```
import numpy

arr = numpy.random.rand(5)

print(arr)

print(numpy.trunc(arr*10))

print(_____)
```

51.문제

10개의 랜덤값을 포함하는 리스트를 생성하고, 상위 50%에 해당되는 값의 평균값을 구하시오.

52.문제

다음과 같이 튜플 형태의 xy 좌표값 쌍을 3개 가지는 리스트가 존재한다. x값과 y값을 분리해서 각각 리스트를 생성하는 프로그램을 완성하시오.

```
pt = [(3,5),(1,-1),(8,-9)]

print(pt)

xx = []; yy = []
```

```
print(xx,yy)
```

```
[(3, 5), (1, -1), (8, -9)]
[3, 1, 8] [5, -1, -9]
```

53.문제

다음과 같이 튜플 형태의 x,y 좌표값 쌍을 3개 가지는 리스트가 존재한다. x값과 y값을 분리해서 각각 리스트를 생성하는 프로그램을 완성하시오.

```
pt = [(3,5),(1,-1),(8,-9)]
```

```
print(pt)
```

```
xx = _____
```

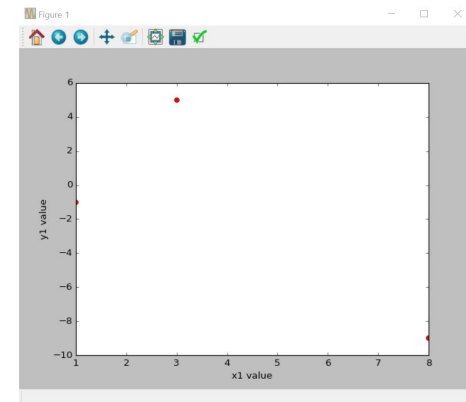
```
yy = _____
```

```
print(xx,yy)
```

```
[(3, 5), (1, -1), (8, -9)]
[3, 1, 8] [5, -1, -9]
```

54.문제

다음과 같이 튜플 형태의 x,y 좌표값 쌍을 3개 가지는 리스트가 존재한다. x값과 y값을 분리해서 각각 리스트를 생성했고, 이를 그림과 같이 출력하는 함수를 완성하시오.



```
import _____
```

```
pt = [(3,5),(1,-1),(8,-9)]
```

```
print(pt)
```

```
xx = list(map(lambda x:x[0],pt))
```

```
yy = list(map(lambda x:x[1],pt))
```

```
print(xx,yy)
```

```
plt.plot(xx,yy,'ro')
```

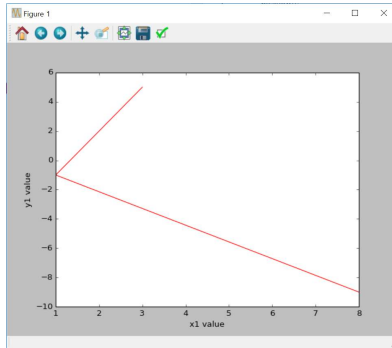
```
plt.xlabel('x1 value')
```

```
plt.ylabel('y1 value')
```

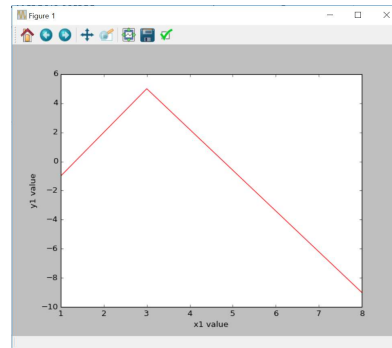
```
plt.show()
```

55.문제

다음과 같이 튜플 형태의 x,y 좌표값 쌍을 3개 가지는 리스트가 존재한다. x값과 y값을 분리해서 각각 리스트를 생성했고, 점을 연결한 경우 아래 (a) 그림과 같은 출력결과를 보인다. (b) 와 같이 출력되도록 프로그램을 수정하시오.



(a)



(b)

```
import matplotlib.pyplot as plt

pt = [(3,5),(1,-1),(8,-9)]

print(pt)

xx = list(map(lambda x:x[0],pt))

yy = list(map(lambda x:x[1],pt))

print(xx,yy)

plt.plot(xx,yy,'r-')

plt.xlabel('x1 value')

plt.ylabel('y1 value')

plt.show()
```

파일 입출력

56.문제

test2.txt 파일에 "Hello World" 를 기록하고 이것을 다시 불러서 출력하는 프로그램을 작성하시오.

```
f = open("test2.txt",'w')

f.write("Hello World !");

f.close();

f = _____

s = f.read()

f.close();

print(s)
```

57.문제

test2.txt 파일에 리스트 데이터를 기록하고 이것을 다시 불러서 출력하는 프로그램을 작성하시오

```
name = ['reachal ','kevin ','tom ']

f = _____

f.writelines(name);

f.close();
```

58.문제

다음문장을 포함하는 wikitest2.txt 파일을 읽고,

Python is a widely used high-level programming language for general-purpose programming, created by Guido van Rossum and first released in 1991. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java.[25][26] It provides constructs that enable clear programming on both small and large scales.[27]

Python 이 포함된 문장만을 출력하는 프로그램을 작성하시오.

```
f = open('wikitest2.txt','r')
s = f.read()
print(s)
rrr = []
tt = _____
for kk in tt:
    rrr += kk.split('\n')
print(tt)
keyword = "Python"
print('-----')
for ii, kk in enumerate(rrr): # enumerate
    if kk.find("Python") >= 0:
        print(ii, kk)
```

Python is a widely used high-level programming language for general-purpose programming, created by Guido van Rossum and first released in 1991. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java.[25][26] It provides constructs that enable clear programming on both small and large scales.[27]

['Python is a widely used high-level programming language for general-purpose programming, created by Guido van Rossum and first released in 1991', 'An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java', '[25][26] It provides constructs that enable clear programming on both small and large scales', '[27]\n']

0 Python is a widely used high-level programming language for general-purpose programming, created by Guido van Rossum and first released in 1991

1 An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code

than might be used in languages such as C++ or Java

59.문제

다음문장을 포함하는 wikitest2.txt 파일을 읽고,

Python is a widely used high-level programming language for general-purpose programming, created by Guido van Rossum and first released in 1991. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java.[25][26] It provides constructs that enable clear programming on both small and large scales.[27]

Python 키워드를 모두 Java 로 바꾼 후, Out.txt 파일에 저장하는 프로그램을 작성하시오.

```
f = open('wikitest2.txt','r')
s = f.read()
print(s)
f = open("out.txt",'w')
s = _____
f.writelines(s);
f.close();
print(s)
```

Python is a widely used high-level programming language for general-purpose programming, created by Guido van Rossum and first released in 1991. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java.[25][26] It provides constructs that enable clear programming on both small and large scales.[27]

Java is a widely used high-level programming language for general-purpose programming, created by Guido van Rossum and first released in 1991. An interpreted language, Java has a

design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java.[25][26] It provides constructs that enable clear programming on both small and large scales.[27]

시리얼 통신

60.문제

시리얼 통신을 하고자한다. 다음 코드를 분석해보자.

```
import serial #pyserial3.4 설치

from tkinter import *

serialPort = "COM3"

baudRate = 9600

ser = serial.Serial(serialPort , baudRate, timeout=0, writeTimeout=0) #ensure non-blocking

#make a TkInter Window

root = Tk()

root.wm_title("Reading Serial")

# make a scrollbar

scrollbar = Scrollbar(root)

scrollbar.pack(side=RIGHT, fill=Y)

# make a text box to put the serial output

log = Text ( root, width=30, height=30, takefocus=0)
```

```
log.pack()

# attach text box to scrollbar

log.config(yscrollcommand=scrollbar.set)

scrollbar.config(command=log.yview)

#make our own buffer

#useful for parsing commands

#Serial.readline seems unreliable at times too

serBuffer = ""

rate = 5

def readSerial():

    global serBuffer

    c = ser.read()

    c = c.decode("utf=8")

    serBuffer += c

    if c == '\n':

        log.insert('0.0', serBuffer)

        serBuffer = ""

    root.after(rate, readSerial) # check serial again soon

# after initializing serial, an arduino may need a bit of time to reset

root.after(rate, readSerial)
```

```
root.mainloop()
```

GUI 프로그래밍

61.문제

다음은 레이더 프로그램에이다. 프로그램을 분석해보자.

```
from tkinter import *
import math

window = Tk()

canvas = Canvas(window, width=1100,height=600,background="black")

canvas.pack()

radius = 500;

degree = 0

rate = 10

xx =550

xy=580

direction = 1

def draw_init():

    deg=30
```

```
global radius,degree,rate

r1 = 100

for ii in range(7):

    rrr = 500

    rad = (deg*ii)*3.14/180

    canvas.create_line(xx,xy,

                       xx+rrr*math.cos(rad),xy-rrr*math.sin(rad),

                       fill='white',width=2)

    for jj in range(1,6):

        r1 = 100*jj

        canvas.create_arc(xx-r1,xy-r1,xx+r1,xy+r1,extent=180,outline="white",width=2)

def draw_line():

    global radius,degree,rate,direction

    canvas.delete('all')

    if direction == 1:

        if degree >=0 and degree <180:

            degree += 1

        elif degree>=180:

            direction = 0

    elif direction == 0:

        if degree >0 and degree <=180:

            degree -= 1
```

```
        elif degree <=0:
            direction = 1

        rad = degree*3.14/180
        canvas.create_line(xx,xy,
                           xx+radius*math.cos(rad),xy-radius*math.sin(rad),
                           fill='green',width=10)

        draw_init()

        window.after(rate, draw_line)

draw_init()
window.after(rate, draw_line)

window.mainloop()
```