문제2-1 (Index 이용방법)

lt = list(range(1,6))

tf=[True,True,True,True,True]

lt2=[]

cnt=0

def my\_number(n):

if n>1:

for i in range(2,n):

if n%i==0:

return False

else:

return False

return True

for i,j in enumerate(lt):

tf[i]=my\_number(j)

if tf[i]:

cnt+=1

lt2=list(range(cnt))

k=0

for i,j in enumerate(lt):

if tf[i]:

lt2[k]=(k+1,j)

k+=1

print (lt2)

정수(n)를 매개변수로 받고 소수면 True 아니면 False가 리턴되는 함수입니다.

n이 1보다 작으면 소수가 아니므로 False를 리턴합니다

n이 1보다 크면 2부터 n-1까지 증가하는 i로 n을 나누어서 나누어지면 소수가 아니므로 False를 리턴합니다.

위 조건에 맞지 않으면 소수이므로 True를 리턴합니다.

my\_number 함수를 이용해 tf에 소수인지 아닌지를 index에 맞춰 넣어줍니다

lt에 소수가 있는지를 tf의 값을 통해 알아내고 소수면 몇 번째 소수인지와 값을 lt2에 넣어줍니다

lt2를 출력합니다

문제2-2 (내장함수 이용방법)

lt = list(range(1,6))

lt2=[]

def my\_number(n): #소수판별 함수

if n>1:

for i in range(2,n):

if n%i==0:

return False

else:

return False

return True

print (lt2)

my\_number함수를 이용해 lt에 있는 값들이 소수인지 판별한뒤 소수면 몇번째 소수인지와 값을 lt2에 넣습니다.

Lt2를 출력합니다.

문제2-3 (함수화)

g\_lt = list(range(1,6))

def my\_number(n):

if n>1:

for i in range(2,n):

if n%i==0:

return False

else:

return False

return True

def li\_mul(x\_lt):

lt2=[]

cnt=1

for i in x\_lt:

if my\_number(i):

lt2.append((cnt,i))

cnt+=1

return lt2

print(li\_mul(g\_lt))

리스트를 매개변수로 받아서, 몇 번째 소수인지와 소수인 값을 튜플형태로 저장하는 리스트를 리턴하는 함수입니다.

문제2-4 (List Comprehension)

<코드>

lt = list(range(1,6))

def my\_number(n): #소수판별 함수

if n>1:

for i in range(2,n):

if n%i==0:

return False

else:

return False

return True

def prime(lt): #소수만 들어있는 리스트를 리턴하는 함수

lt2=[]

for i in lt:

if my\_number(i):

lt2.append(i)

return lt2

lt2=[(lambda xx:((kk+1,jj)))(kk) for kk,jj in enumerate(prime(lt))]

print (lt2)

소수만 들어있는 리스트를 만들어서 (인덱스+1, 값)들을 lt2에 저장하고 출력

문제2-6 (Map & filter)

lt = list(range(1,6))

def my\_number(n): #소수판별 함수

if n>1:

for i in range(2,n):

if n%i==0:

return False

else:

return False

return True

lt2 = list(filter(my\_number,lt))

lt3=[(lambda xx:((kk+1,jj)))(kk) for kk,jj in enumerate(lt2)]

print(lt3)

Filter로 lt의 소수인숫자만 lt2에 저장하고 정수(n)을 매개변수로 하고 n이 lt에 있으면 인덱스와 값을 리턴하는 함수(de)를 만들었습니다. map으로 de를 이용해서 lt3에 소수의 순서와 소수를 저장하고 출력합니다.