|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **연도** | **일시** | **범위** | **연도** | **일시** | **범위** |
| **2022** | **06.30** | **1,2,3** |  |  |  |
|  | **07.01** | **4,5,6** |  |  |  |
|  | **07.03** | **7,8,9** |  |  |  |
|  | **07.05** | **10,11,12** |  |  |  |
|  | **07.07** | **13,14** |  |  |  |
|  | **07.16** | **15,16,17** |  |  |  |
|  | **07.17** | **18,19** |  |  |  |
|  | **07.18** | **20,21** |  |  |  |
|  | **07.19** | **22,23** |  |  |  |
|  | **07.22** | **24** |  |  |  |
|  | **07.24** | **25.26** |  |  |  |
|  | **07.25** | **27,28,29,30** |  |  |  |
|  | **07.26** | **31,32** |  |  |  |
|  | **07.27** | **33** |  |  |  |
|  | **07.29** | **34** |  |  |  |
|  | **07.31** | **35** |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**파이썬 문제 풀이**

1. 표준 입력으로 숫자 또는 문자열 여러 개가 입력되어 리스트x에 저장됩니다(입력되는 숫자 또는 문자열의 개수는 정해져 있지 않음). 다음 소스 코드를 완성하여 리스트x의 마지막 요소 5개를 삭제한 뒤 튜플로 출력되게 만드세요.

|  |
| --- |
| x = input().split()  print(tuple(x[:len(x)-5])) |

2. 표준 입력으로 문자열 두 개가 각 줄에 입력됩니다(문자열의 길이는 정해져 있지 않음). **첫 번째 문자열에서 인덱스가 홀수인 문자와 두 번째 문자열에서 인덱스가 짝수인 문자를 연결하여 출력하는 프로그램**을 만드세요(input에서 안내 문자열은 출력하지 않아야 합니다). 연결 순서는 첫 번째 문자열, 두 번째 문자열 순입니다. 그리고 0은 짝수로 처리합니다.

|  |
| --- |
| x=input()  y=input()  a=x[1::2]  b=y[::2]  print(a+b) |

3. 표준 입력으로 가격(정수)과 쿠폰 이름이 각 줄에 입력됩니다.Cash3000쿠폰은 3,000원,Cash5000쿠폰은 5,000원을 할인합니다. **쿠폰에 따라 할인된 가격을 출력하는 프로그램**을 만드세요(input에서 안내 문자열은 출력하지 않아야 합니다).

|  |
| --- |
| price=int(input()) #가격입력  coupon=input() #쿠폰이름 입력  if coupon=='Cash3000':  price-=3000    if coupon=='Cash5000':  price-=5000    print(price) |

4. 표준 입력으로 국어, 영어, 수학, 과학 점수가 입력됩니다. 여기서 네 과목의 평균 점수가 80점 이상일 때 합격이라고 정했습니다. 평균 점수에 따라'**합격','불합격'을 출력하는 프로그램**을 만드세요(input에서 안내 문자열은 출력하지 않아야 합니다). 단, 점수는 0점부터 100점까지만 입력받을 수 있으며 범위를 벗어났다면'잘못된 점수'를 출력하고 합격, 불합격 여부는 출력하지 않아야 합니다.

|  |
| --- |
| a,b,c,d=map(int,input().split())  avg=(a+b+c+d)//4  if 0<=a<=100 and 0<=b<=100 and 0<=c<=100 and 0<=d<=100:  if avg>=80:  print('합격')  else:  print('불합격')  else:  print('잘못된 점수') |

5. 표준 입력으로 나이(만 나이)가 입력됩니다(입력 값은 7 이상 입력됨). **교통카드 시스템**에서 시내버스 요금은 다음과 같으며 각 나이에 맞게 요금을 차감한 뒤 잔액이 출력되게 만드세요(if,elif사용). 현재 교통카드에는 9,000원이 들어있습니다.

어린이(초등학생, 만 7세 이상 12세 이하): 650원

청소년(중∙고등학생, 만 13세 이상 18세 이하): 1,050원

어른(일반, 만 19세 이상): 1,250원

|  |
| --- |
| age = int(input())  balance = 9000 # 교통카드 잔액  if 7<=age<=12:  balance-=650  elif 13<=age<=18:  balance-=1050  else:  balance-=1250  print(balance) |

6.표준 입력으로 정수가 입력됩니다. 입력된 정수의 **구구단**을 출력하는 프로그램을 만드세요(input에서 안내 문자열은 출력하지 않아야 합니다). 출력 형식은숫자 \* 숫자 = 숫자처럼 만들고 숫자와\*,=사이는 공백을 한 칸 띄웁니다.

|  |
| --- |
| x=int(input())  for i in range(1,10):  print(x,'\*',i,'=',x\*i) |

7.정수 2 5, 4 4, 8 3, 16 2, 32 1이 각 줄에 출력되게 만드세요.while에 조건식은 두 개 지정하고, 두 변수를 모두 변화시켜야 합니다

|  |
| --- |
| i = 2  j = 5  while i<=32 or j>=1:  print(i,j)  i\*=2  j-=1 |

8.표준 입력으로 금액(정수)이 입력됩니다. **1회당 요금은 1,350원이고, 교통카드를 사용했을 때마다의 잔액을 각 줄에 출력하는 프로그램**을 만드세요(input에서 안내 문자열은 출력하지 않아야 합니다). 단, 최초 금액은 출력하지 않아야 합니다. 그리고 잔액은 음수가 될 수 없으며 잔액이 부족하면 출력을 끝냅니다.

|  |
| --- |
| charge=int(input())  price\_once=1350  while charge-price\_once>=0:  charge-=price\_once  print(charge) |

9. 다음 소스 코드를 완성하여 **0과 73사이의 숫자 중 3으로 끝나는 숫자만 출력**되게 만드시오.

|  |
| --- |
| i = 0  while True:  if i%10!=3:  i+=1  continue  if i>73:  break  print(i, end=' ')  i += 1 |

10. 표준 입력으로 정수 두 개가 입력됩니다(첫 번째 입력 값의 범위는 1~200, 두 번째 입력 값의 범위는 10~200이며 첫 번째 입력 값은 두 번째 입력 값보다 항상 작습니다). 다음 소스 코드를 완성하여 **첫 번째 정수와 두 번째 정수 사이의 숫자 중 3으로 끝나지 않는 숫자가 출력**되게 만드세요. 정답에 코드를 작성할 때는while True:에 맞춰서 들여쓰기를 해주세요.

|  |
| --- |
| start, stop = map(int, input().split())    i = start    while True:  if i>stop:  break  if i%10==3:  i+=1  continue  if i==stop+1:  break  print(i, end=' ')  i += 1 |

11. 다음과 같은 모양의 **별모양이 출력**되도록 하시오.

|  |  |
| --- | --- |
| for i in range(5):  for j in range(5):  if j<i:  print(' ',end=' ')  else:  print('\*',end=' ')  print() | \*\*\*\*\*  \*\*\*\*  \*\*\*  \*\*  \* |

12. 표준 입력으로 삼각형의 높이가 입력됩니다. **입력된 높이만큼 산 모양으로 별을 출력하는 프로그램**을 만드세요(input에서 안내 문자열은 출력하지 않아야 합니다). 이때 출력 결과는 예제와 정확히 일치해야 합니다. 모양이 같더라도 공백이나 빈 줄이 더 들어가면 틀린 것으로 처리됩니다.

|  |  |
| --- | --- |
| n = int(input())  for i in range(1,n+1):  print(" "\*(n-i)+"\*"\*(i\*2-1)) | \*  \*\*\*  \*\*\*\*\*  \*\*\*\*\*\*\*  \*\*\*\*\*\*\*\*\* |

13.표준 입력으로 정수 두 개가 입력됩니다(첫 번째 입력 값의 범위는 1~1000, 두 번째 입력 값의 범위는 10~1000이며 첫 번째 입력 값은 두 번째 입력 값보다 항상 작습니다). 첫 번째 정수부터 두 번째 정수까지 숫자를 출력하면서 **5의 배수일 때는'Fizz', 7의 배수일 때는'Buzz', 5와 7의 공배수일 때는'FizzBuzz'를 출력하는 프로그램**을 만드세요(input에서 안내 문자열은 출력하지 않아야 합니다).

|  |  |
| --- | --- |
| x,y=map(int,input().split())  for i in range(x,y+1):  if i%35==0:  print('FizzBuzz')  elif i%7==0:  print('Buzz')  elif i%5==0:  print('Fizz')  else:  print(i) | 입력35 40  결과  FizzBuzz 36 37 38 39 Fizz |

14. 다음 소스 코드를 완성하여 오각별이 그려지게 만드세요.

각 변의 길이는 100 / 별의 꼭지점은 72도를 두 번 회전해서 144도 회전 / 별의 다음 꼭지점을 그릴 때는 72도 회전

|  |  |
| --- | --- |
| import turtle as t  n=5  t.shape('turtle')  for i in range(n):  t.forward(100)  t.right(360/n\*2)  t.forward(100)  t.left(360/n) |  |

15. 표준 입력으로 꼭지점 개수(정수)와 선의 길이(정수)가 입력됩니다(꼭지점 개수의 입력 범위는 5~10, 선의 길이 입력 범위는 50~150입니다). 다음 소스 코드를 완성하여 꼭지점 개수와 선의 길이에 맞는 별이 그려지게 만드세요. 별을 그릴 때는 현재 위치부터 오른쪽으로 이동해서 시작해야 하며 시계 방향으로 그려야 합니다.

|  |  |
| --- | --- |
| import turtle as t    n, line = map(int, input().split())  t.shape('turtle')  t.speed('fastest')  for i in range(n):  t.forward(line)  t.right(360/n\*2)  t.forward(line)  t.left(360/n) |  |

16. 리스트에서 가장 작은 수와 큰 수 구하기

|  |
| --- |
| a=[10,1,20,17,100]  smallest=a[0]  largeest=a[0]  for i in a:  if i < smallest:  smallest=i  if i > largeest:  largeest=i    print(smallest)  print(largeest) |

17.

|  |
| --- |
| a = ['alpha', 'bravo', 'charlie', 'delta', 'echo', 'foxtrot', 'golf', 'hotel', 'india']  b = [i for i in a if len(i)==5]    print(b) |

18.표준 입력으로 정수 두 개가 입력됩니다(첫 번째 입력 값의 범위는 1~20, 두 번째 입력 값의 범위는 10~30이며 첫 번째 입력 값은 두 번째 입력 값보다 항상 작습니다). **첫 번째 정수부터 두 번째 정수까지를 지수로 하는 2의 거듭제곱 리스트를 출력하는 프로그램**을 만드세요(input에서 안내 문자열은 출력하지 않아야 합니다). 단, 리스트의 두 번째 요소와 뒤에서 두 번째 요소는 삭제한 뒤 출력하세요. 출력 결과는 리스트 형태라야 합니다.

|  |
| --- |
| a,b=map(int,input().split())  #a는 1~20  #b는 10~30  #a는 b보다 항상 작다.  c=[2\*\*i for i in range(a,b+1)]  del c[1]  del c[-2]  print(c) |

19. 0**을 입력될 때까지 입력된 모든 값의 합을 출력**하시오

|  |
| --- |
| sum=0  while True:  x=int(input())  sum += x  if x==0:  break  print(sum) |

20.다음 소스 코드를 완성하여 높이 2, 세로 크기 4, 가로 크기 3인 3차원 리스트를 만드세요(리스트 표현식 사용).

|  |
| --- |
| from pprint import pprint  a=[[[0 for col in range(3)]  for row in range(4)]  for depth in range(2)]  pprint(a) |
| [[[0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0]],  [[0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0]]] |

21. 표준 입력으로 2차원 리스트의 가로(col)와 세로(row)가 입력되고 그 다음 줄부터 리스트의 요소로 들어갈 문자가 입력됩니다. 이때 2차원 리스트 안에서\*는 지뢰이고.은 지뢰가 아닙니다. **지뢰가 아닌 요소에는 인접한 지뢰의 개수를 출력하는 프로그램**을 만드세요(input에서 안내 문자열은 출력하지 않아야 합니다).

여러 줄을 입력 받으려면 다음과 같이for반복문에서input을 호출한 뒤append로 각 줄을 추가하면 됩니다(list안에 문자열을 넣으면 문자열이 문자 리스트로 변환됩니다).

matrix =[]

for i in range(row):

matrix.append(list(input()))

|  |
| --- |
| col, row = map(int, input().split())  matrix = []  for i in range(row):  matrix.append(list(input()))  for i in range(row):  for j in range(col):  if matrix[i][j] == '\*':  continue  else:  matrix[i][j] = 0 # 0을 넣은 뒤에는 요소 주변 8개를 탐색하면서 \*이면 요소를 1개씩 증가  # range에서 생성한 숫자의 마지막 값은 끝나는 값보다 1 작은데  # 탐색을 위해서 한 칸 아래까지 더 반복해야 하므로 i + 1이 아닌 i + 2가 되어야 함  for y in range(i - 1, i + 2): # 한 칸 위부터 한 칸 아래까지 반복  for x in range(j -1, j + 2): # 한 칸 앞(왼쪽)부터 한 칸 뒤(오른쪽)까지 반복  if y < 0 or x < 0 or y >= row or x >= col: # 리스트의 범위를 벗어나면 건너뛰기  continue  elif matrix[y][x] == '\*':  matrix[i][j] += 1  for i in range(row):  for j in range(col):  print(matrix[i][j], end='')  print() |

21.표준 입력으로 양의 정수 두 개가 입력됩니다. 다음 소스 코드를 완성하여 두 숫자의 공약수를 세트 형태로 구하도록 만드세요. 단, 최종 결과는 공약수의 합으로 판단합니다.

|  |  |
| --- | --- |
| 노가다로 내가 푼 풀이  a,b=map(int,input().split())  count=0  for i in range(1,a+1):  if a%i==0:  print(i)  count+=i  print(count)  divisor = a & b    result = 0  if type(divisor) == set:  result = sum(divisor)    print(result) | 힌트를 보며 푼 풀이  a,b=map(int,input().split())  a={i for i in range(1,a+1) if a%i==0}  b={i for i in range(1,b+1) if b%i==0}  divisor = a & b    result = 0  if type(divisor) == set:  result = sum(divisor)    print(result) |

22.파일에서 10자 이하인 단어 개수 세기

단어가 줄 단위로 저장된words.txt파일이 주어집니다. 다음 소스 코드를 완성하여 10자 이하인 단어의 개수가 출력되게 만드세요.

|  |  |
| --- | --- |
| anonymously  compatibility  dashboard  experience  photography  spotlight  warehouse  with open('words.txt','w') as file:  file.write('anonymously\n')  file.write('compatibility\n')  file.write('dashboard\n')  file.write('experience\n')  file.write('photography\n')  file.write('spotlight\n')  file.write('warehouse\n') | with open('words.txt','r') as file:  count=0  words=file.readlines()  for word in words:  if len(word.strip('\n'))<=10:  count+=1  print(count) |

23. 문자열이 저장된words.txt파일이 주어집니다(문자열은 한 줄로 저장되어 있습니다).words.txt파일에서 문자c가 포함된 단어를 각 줄에 출력하는 프로그램을 만드세요. 단어를 출력할 때는 등장한 순서대로 출력해야 하며,(콤마)와.(점)은 출력하지 않아야 합니다.

|  |  |
| --- | --- |
| Fortunately, however, for the reputation of Asteroid B-612, a Turkish dictator made a law that his subjects, under pain of death, should change to European costume. So in 1920 the astronomer gave his demonstration all over again, dressed with impressive style and elegance. And this time everybody accepted his report. | with open('words.txt','w') as file:  file.write('Fortunately, however, for the reputation of Asteroid B-612, a Turkish dictator made a law that his subjects, under pain of death, should change to European costume. So in 1920 the astronomer gave his demonstration all over again, dressed with impressive style and elegance. And this time everybody accepted his report.')  import string    with open('words.txt','r') as file:  c\_words=file.readline()  c\_words\_split=c\_words.split()  for c\_word in c\_words\_split:  if 'c' in c\_word:  print(c\_word.strip(string.punctuation)) |

24.단어 단위 N-gram 만들기

표준 입력으로 정수와 문자열이 각 줄에 입력됩니다. 다음 소스 코드를 완성하여 입력된 숫자에 해당하는 단어 단위 N-gram을 튜플로 출력하세요(리스트 표현식 사용). 만약 입력된 문자열의 단어 개수가 입력된 정수 미만이라면'wrong'을 출력하세요.

|  |
| --- |
| n = int(input())  text = input()  words=text.split()    if len(words)<n:  print('wrong')  else:  n\_gram =zip(\*[words[i:]for i in range(n)])  for i in n\_gram:  print(i) |

25.단어가 줄 단위로 저장된words.txt파일이 주어집니다.words.txt파일에서 회문인 단어를 각 줄에 출력하는 프로그램을 만드세요. 단어를 출력할 때는 등장한 순서대로 출력해야 합니다. 그리고 파일에서 읽은 단어는\n이 붙어있으므로\n을 제외한 뒤 회문인지 판단해야 하며 단어를 출력할 때도\n이 출력되면 안 됩니다(단어 사이에 줄바꿈이 두 번 일어나면 안 됨).

|  |  |
| --- | --- |
| '''내가 푼 풀이'''  import pickle  with open('wordsss.txt','r') as file:  s=file.read()    is\_palindrome = True  test\_words=s.split()  for i in test\_words:  for j in range(len(i)//2):  if s[j]!=s[-1-j]:  is\_palindrome = False  break    print(is\_palindrome)  '''내가 푼 풀이2'''  with open('wordsss.txt','r') as file:  s=file.read()  t=s.split()    for i in t:  for j in i.split():  if list(j)==list(reversed(j)):  print(j.strip()) | with open('wordsss.txt','r') as file:  for i in file:  text=i.split()  for j in text:  if list(j)==list(reversed(j)):  print(j.strip()) |

26.표준 입력으로 숫자 두 개가 입력됩니다. 다음 소스 코드를 완성하여 두 숫자의 덧셈, 뺄셈, 곱셈, 나눗셈의 결과가 출력되게 만드세요. 이때 나눗셈의 결과는 실수라야 합니다.

|  |
| --- |
| x, y = map(int, input().split())  def calc(a,b):  return a+b,a-b,a\*b,a/b  a, s, m, d = calc(x, y)  print('덧셈: {0}, 뺄셈: {1}, 곱셈: {2}, 나눗셈: {3}'.format(a, s, m, d)) |

27.가장 높은 점수를 구하는 함수 만들기

|  |
| --- |
| korean, english, mathematics,science = 100, 86, 81, 91  def get\_max\_score(\*args):  return max(args)  max\_score = get\_max\_score(korean, english, mathematics, science)  print('높은 점수:', max\_score)    max\_score = get\_max\_score(english, science)  print('높은 점수:', max\_score) |

28.표준 입력으로 국어, 영어, 수학, 과학 점수가 입력됩니다. 다음 소스 코드를 완성하여 가장 높은 점수, 가장 낮은 점수, 평균 점수가 출력되게 만드세요. 평균 점수는 실수로 출력되어야 합니다.

|  |  |
| --- | --- |
| 내가 푼 풀이  korean, english, mathematics, science = map(int, input().split())  def get\_min\_max\_score(\*args):  return min(args),max(args)    def get\_average(\*\*kwargs):  return sum(kwargs)/kwargs    min\_score,max\_score=get\_min\_max\_score(korean, english, mathematics, science)  average\_score = get\_average(korean=korean, english=english,  mathematics=mathematics, science=science)  print('낮은 점수: {0:.2f}, 높은 점수: {1:.2f}, 평균 점수: {2:.2f}'  .format(min\_score,max\_score,average\_score))  min\_score,max\_score=get\_min\_max\_score(english, science)  average\_score = get\_average(english=english, science=science)  print('낮은 점수: {0:.2f}, 높은 점수: {1:.2f}, 평균 점수: {2:.2f}'  .format(min\_score,max\_score,average\_score)) | 힌트 참고해서 푼 풀이  korean, english, mathematics, science = map(int, input().split())  def get\_min\_max\_score(\*args):  return min(args),max(args)    def get\_average(\*\*kwargs):  return sum(kwargs.values())/len(kwargs)    min\_score,max\_score=get\_min\_max\_score(korean, english, mathematics, science)  average\_score = get\_average(korean=korean, english=english,  mathematics=mathematics, science=science)  print('낮은 점수: {0:.2f}, 높은 점수: {1:.2f}, 평균 점수: {2:.2f}'  .format(min\_score,max\_score,average\_score))  min\_score,max\_score=get\_min\_max\_score(english, science)  average\_score = get\_average(english=english, science=science)  print('낮은 점수: {0:.2f}, 높은 점수: {1:.2f}, 평균 점수: {2:.2f}'  .format(min\_score,max\_score,average\_score)) |

29.재귀 호출로 회문 판별하기

|  |
| --- |
| #재귀호출로 회문판별하기  def is\_palindrome(word):  if len(word)<2:  return True  if word[0]!=word[-1]:  return False  return is\_palindrome(word[1:-1])      print(is\_palindrome('hello'))  print(is\_palindrome('level')) |

30.재귀호출로 피보나치 수 구하기

표준 입력으로 정수 한 개가 입력됩니다(입력 값의 범위는 10~30). 다음 소스 코드를 완성하여 입력된 정수에 해당하는 피보나치 수가 출력되게 만드세요.

피보나치 수는 0과 1로 시작하며, 다음 번 피보나치 수는 바로 앞의 두 피보나치 수의 합입니다.

|  |
| --- |
| #재귀호출로 피보나치 수 구하기  def fib(n):  if n==0:  return 0  elif n==1:  return 1  else:  return fib(n-1)+fib(n-2)      n=int(input())  print(fib(n)) |

２０２２년　７월　２６일　화요일

31. 확장자가 .jpg, .png인 이미지 파일만 출력되게 만드세요. 여기서는 람다 표현식을 사용해야 하며 출력 결과는 리스트 형태라야 합니다. 람다 표현식에서 확장자를 검사할 때는 문자열 메서드를 활용하세요.

|  |
| --- |
| #32.4연습문제:이미지 파일만 가져오기  files = ['font', '1.png', '10.jpg',  '11.gif', '2.jpg', '3.png',  'table.xslx', 'spec.docx']  a=list(filter(lambda x:x.find('.jpg')!=-1 #filer 함수는 True여야 값을 반환한다.  or x.find('.png')!=-1,files))  print(a) |

32. 표준 입력으로**숫자.확장자**형식으로 된 파일 이름 여러 개가 입력됩니다. 다음 소스 코드를 완성하여 파일 이름이 숫자 3개이면서 앞에 0이 들어가는 형식으로 출력되게 만드세요. 예를 들어1.png는001.png,99.docx는099.docx,100.xlsx는100.xlsx처럼 출력되어야 합니다. 그리고 람다 표현식을 사용해야 하며 출력 결과는 리스트 형태라야 합니다. 람다 표현식에서 파일명을 처리할 때는 문자열 포매팅과 문자열 메서드를 활용하세요.**(1시간 30분 소요)**

|  |
| --- |
| files = input().split()  print(list(map(lambda x:x.split('.')[0].zfill(3)+'.'+x.split('.')[1],files)))  또는  list(map(**lambda**x: '{0:03d}.{1}'.format(int(x.split('.')[0]),x.split('.')[1]) ,files)) |

33. 클래스를 작성하여 게임 캐릭터의 능력치와'베기'가 출력되게 만드세요.

|  |
| --- |
| class Knight:  def \_\_init\_\_(self,health,mana,armor):  self.health=health  self.mana=mana  self.armor=armor  def slash(self):  print('베기')      x = Knight(health=542.4, mana=210.3, armor=38)  print(x.health, x.mana, x.armor)  x.slash() |

34.표준 입력으로 게임 캐릭터 능력치(체력, 마나, AP)가 입력됩니다. 다음 소스 코드에서 애니(Annie) 클래스를 작성하여 티버(tibbers) 스킬의 피해량이 출력되게 만드세요. 티버의 피해량은AP \* 0.65 + 400이며 AP(Ability Power, 주문력)는 마법 능력치를 뜻합니다.

|  |
| --- |
| class Annie:  def \_\_init\_\_(self,health,mana,ability\_power):  self.health=health  self.mana=mana  self.ability\_power=ability\_power    def tibbers(self):  print('티버:피해량',self.ability\_power\*0.65+400)    health, mana, ability\_power = map(float, input().split())  #능력치(체력, 마나, AP) 입력받는다.  x = Annie(health=health, mana=mana, ability\_power=ability\_power)  #클래스 인스턴스 생성/ 티버스킬의 피해량이 출력되게 만들기  #티버의 피해량은 AP\*0.65+400이고, AP는 마법 능력치를 의미  x.tibbers() |

20220731

35.is\_date\_valid는 문자열이 올바른 날짜인지 검사하는 메서드입니다. 날짜에서 월은 12월까지 일은 31일까지 있어야 합니다.

|  |
| --- |
| class Date:  @staticmethod  def is\_date\_valid(dates):  y,m,d=map(int,dates.split('-'))  return m<=12 and d<=31    if Date.is\_date\_valid('2000-10-31'):  print('올바른 날짜 형식입니다.')  else:  print('잘못된 날짜 형식입니다.') |