

```
In [1]: #Perform Following on Python Shell Window

5 ** 9

Out[1]: 1953125

In [2]: 3 // 2

Out[2]: 1

In [3]: 7/3

Out[3]: 2.3333333333333335

In [4]: 6 == 6

Out[4]: True

In [5]: a =20
a+= 30
a%= 3
print(a)

2

In [6]: True * False

Out[6]: 0

In [7]: True & False

Out[7]: False

In [8]: True and False

Out[8]: False

In [9]: ((6>3) and (7<4) or (18 == 3)) and (9>3)

Out[9]: False

In [10]: True is False

Out[10]: False

In [ ]: False in 'False'

((True == False) or (False > True)) and (False <= True)

In [2]: ((True == False) or (False > True)) and (False <= True)

Out[2]: False

In [5]: #Q3. Try to get following output from two python strings
#s1 = "Nice to have it"
#s2 = "here"
#Expected output
#Nice to have it here

s1 = "Nice to have it"
s2 = "here"

print(s1+" "+s2)

Nice to have it here

In [8]: #4. Given this nested list, use indexing to grab the word "hello"
#a = [1,2,[3,4],[5,[100,200,['hello']],23,11],1,7]

a = [1,2,[3,4],[5,[100,200,['hello']],23,11],1,7]
b = a[3][1][2]
print(b)

['hello']

In [9]: #5. Try to insert above strings s1 and s2 in the list 'a' mentioned in que 4, in the
#beginning and end of it respectively

s1 = "Nice to have it"
s2 = "here"
a = [1,2,[3,4],[5,[100,200,['hello']],23,11],1,7]
print([s1] + a + [s2])

['Nice to have it', 1, 2, [3, 4], [5, [100, 200, ['hello']], 23, 11], 1, 7, 'here']

In [11]: #6. Write a Python program to print all even numbers from a given numbers list in
#the same order and stop the printing if any numbers that come after 237 in the
#sequence.
#Sample numbers list:
#numbers = [386, 462, 47, 418, 907, 344, 236, 375, 823, 566, 597, 978, 328, 615,
#953, 345, 399, 162, 758, 219, 918, 237, 412, 566, 826, 248, 866, 950, 626, 949,
#687, 217, 815, 67, 104, 58, 512, 24, 892, 894, 767, 553, 81, 379, 843, 831, 445,
#742, 717, 958, 743, 527]

numbers = [386, 462, 47, 418, 907, 344, 236, 375, 823, 566, 597, 978, 328, 615,
953, 345, 399, 162, 758, 219, 918, 237, 412, 566, 826, 248, 866, 950, 626, 949,
687, 217, 815, 67, 104, 58, 512, 24, 892, 894, 767, 553, 81, 379, 843, 831, 445,
742, 717, 958, 743, 527]

for x in numbers:
    if x == 237:
        print(x)
        break,
    elif x%2 == 0:
        print (x)

386
462
418
344
236
566
978
328
162
758
918
237

In [12]: #Q7Write a Python program to print out a set containing all the colours from
#color_list_1 which are not present in color_list_2.
#Test Data:
#color_list_1 = set(["White", "Black", "Red"])
#color_list_2 = set(["Red", "Green"])
#Expected Output:
#{'Black', 'White'}

color_list_1 = set(["White", "Black", "Red"])
color_list_2 = set(["Red", "Green"])
print(color_list_1.difference(color_list_2))

{'Black', 'White'}

In [22]: #Q8. WAP to find if the given input string is Pangram or not

str1 = input("enter the string:")
str2 = str1. lower()
str3 = set(str2)
pangram = [i for i in str3 if ord(i) in range(ord('a'),ord('z')+1)]
if len(pangram) == 26:
    print("pangram")
else:
    print("not pangram")

enter the string:dshdtfbuiybg
not pangram

In [20]: #Q9. Write a Python program that accepts an integer (n) and computes the value of
#n+nn+nnnn.
#Sample value of n is 5
#Expected Result: 615

n = int(input("enter a number:"))
num = (n+ ((n*10) + n) + ((n*100)+(n*10)+n))
print(num)

enter a number:5
615

In [25]: #Q10. Write a python program to take input from console in following fashion
#23 54 12#98 3 17
#and generate the corresponding two list having integers inside (not string)
#x = [23, 54, 12]
#y = [98, 3, 17]

str1 = input()
a = str1.index("#")
a1 = str1 [0:a].split()
a2 = str1[a+1:].split()
list1 = [eval(i) for i in a1]
list2 = [eval(i) for i in a2]
print(list1)
print(list2)

23 54 12# 98 3 17
[23, 54, 12]
[98, 3, 17]

In [26]: #Q11. Write a program that accepts a comma separated sequence of words as input and
#prints the words in a comma-separated sequence after sorting them alphabetically.
#Suppose the following input is supplied to the program:
#without,hello,bag,world
#Then, the output should be:
#bag,hello,without,world

a = input("input words:")
a_list = a.split(",")
a_list.sort()
print(','.','.').join(a_list)

input words:without,hello,bag,world
bag,hello,without,world

In [9]: #Q12. Write a Python function to find the name of person obtained highest marks in
#exam from given dictionary
#Example dictionary
#d = {'Student': ['Rahul', 'Kishore', 'Vidhya', 'Raakhi'],
# 'Marks': [57,87,67,79]}
#Output: Kishore

d = {'Student': ['Rahul', 'Kishore', 'Vidhya', 'Raakhi'],
'Marks': [57,87,67,79]}
a = d['Student']
b = d['Marks']
c=max(b)
d=b.index(c)
print(a[d])

Kishore

In [29]: #Q13. Write a program that accepts a sentence and calculate the number of letters and
#digits.
#Suppose the following input is supplied to the program:
#hello world! 123
#Then, the output should be:
#LETTERS 10
#DIGITS 3

x = input("Enter the string for calculation: ")
letters = 0
digits = 0
for i in x:
    if i.isalpha():
        letters += 1
    elif i.isdigit():
        digits += 1
print("LETTERS",letters)
print("DIGITS",digits)

Enter the string for calculation: hello world!123
LETTERS 10
DIGITS 3

In [ ]: #Q14. Write a python function which creates a new dictionary of students from a given
#Dataset of various subject to a specific subject or topic only.
#Example Data:
#d = {'Name': ['Akash', 'Soniya', 'Vishakha', 'Akshay', 'Rahul', 'Vikas'],
# 'Subject': ['Python', 'Java', 'Python', 'C', 'Python', 'Java'],
# 'Ratings': [8.4, 7.8, 8, 9, 8.2, 5.6]}
#Input: Python
#Output:
#newData = {'Name': ['Akash', 'Vishakha', 'Rahul'],
# 'Subject': ['Python', 'Python', 'Python'],
# 'Ratings': [8.4, 8, 8.2]}

d = {'Name': ['Akash', 'Soniya', 'Vishakha', 'Akshay', 'Rahul', 'Vikas'],
'Subject': ['Python', 'Java', 'Python', 'C', 'Python', 'Java'],
'Ratings': [8.4, 7.8, 8, 9, 8.2, 5.6]}

i = input()
list1 = list(zip(d['Name'],d['Subject'],d['Ratings']))
list2 = []
for j in range(0,len(list1)):
    if (list1[j][0] == i) or (list1[j][1] == i) or (list1[j][2] == i):
        list2.append(list1[j])
x = list(zip(*list2))
new_data = {}
new_data["Name"] = list(x[0])
new_data["Subject"] = list(x[1])
new_data["Ratings"] = list(x[2])
print(new_data)

#2nd

d = {'Name': ['Akash', 'Soniya', 'Vishakha', 'Akshay', 'Rahul', 'Vikas'],
'Subject': ['Python', 'Java', 'Python', 'C', 'Python', 'Java'],
'Ratings': [8.4, 7.8, 8, 9, 8.2, 5.6]}

i = input()
list1 = list(zip(d['Name'],d['Subject'],d['Ratings']))
list2 = []
for j in range(0,len(list1)):
    if (list1[j][0] == i) or (list1[j][1] == i) or (list1[j][2] == i):
        list2.append(list1[j])
keys = d.keys()
y = dict(zip(keys, zip(*list2)))
print(y)

#3rd

d = {'Name': ['Akash', 'Soniya', 'Vishakha', 'Akshay', 'Rahul', 'Vikas'],
'Subject': ['Python', 'Java', 'Python', 'C', 'Python', 'Java'],
'Ratings': [8.4, 7.8, 8, 9, 8.2, 5.6]}

i = input()
keys = d.keys()
list1 = list(zip(d['Name'],d['Subject'],d['Ratings']))
list2 = []
for j in range(0,len(list1)):
    if (list1[j][0] == i) or (list1[j][1] == i) or (list1[j][2] == i):
        list2.append(list1[j])
x = list(zip(*list2))
list3 = [list(x[0]),list(x[1]),list(x[2])]
y = dict(zip(keys, list3))
print(y)

In [1]: #Q15. Define a class with a generator which can iterate the numbers, which are divisible
#by 7, between a given range 0 and n.

def putNumbers(n):
    i=0
    while i<n:
        j=1
        i=i+1
        if j%7==0:
            yield j
for i in putNumbers(100):
    print(i)

0
7
14
21
28
35
42
49
56
63
70
77
84
91
98

In [2]: #Q16. A robot moves in a plane starting from the original point (0,0). The robot can
#move toward UP, DOWN, LEFT and RIGHT with a given steps. The trace of robot
#movement is shown as the following:
#UP 5
#DOWN 3
#LEFT 3
#RIGHT 2
#The numbers after the direction are steps. Please write a program to compute the
#distance from current position after a sequence of movement and original point.
#If the distance is a float, then just print the nearest integer.
#Example:
#If the following tuples are given as input to the program:
#UP 5
#DOWN 3
#LEFT 3
#RIGHT 2
#Then, the output of the program should be:
#2

import math
x = 0
y = 0
UP = eval(input("UP: "))
DOWN = eval(input("DOWN: "))
LEFT = eval(input("LEFT: "))
RIGHT = eval(input("RIGHT: "))
x = x + UP - DOWN
y = y + RIGHT - LEFT
distance = int(math.sqrt(x**2 + y**2))
print(distance)

UP: 5
DOWN: 3
LEFT: 4
RIGHT: 3
2

In [ ]:
```