

```
# 1. Write a program where the user enters a paragraph about their favorite place. The
program should count how many times the words "and," "the," and "a" appear, but ignore
case. If the count of "and" exceeds 5, print "Too many connections in your story!"
# s=input("Enter paragraph ").split(" ")
# c=0
# for i in s:
#     if i.lower()=='and':
#         c+=1
# if c>5:
#     print('Too many connections in your story!')
# output:
# Enter paragraph Enter paragraph nter paragraph hi my name nad my collage is bmit and am
380 degree and ioo and yuio and jijji AND and
# Too many connections in your story!
```

```
# 2. Design a program where two secret codes are provided as inputs (numbers). Perform
addition if the first code is greater than the second, multiplication if the second is greater, and
subtraction if they are equal. If both numbers are even, divide their product by 2 before
displaying it.
# code1=int(input("Enter code1 "))
# code2=int(input("Enter code2 "))
# if code1>code2:
#     print(code2+code1)
# elif code2>code1:
#     print(code1*code2)
# elif code2==code1:
#     print(code2-code1)
# elif code1%2==0 and code2%2==0:
#     print(code1*code2/2)
# output:
# Enter code1 2
# Enter code2 2
# 0
# Enter code1 10
# Enter code2 20
# 200
# Enter code1 10
# Enter code2 2
# 10.0
```

3.Create a program that accepts a mix of strings and numbers as input in a single line (comma-separated). Separate them into two lists: one for integers and one for strings. If the list of integers has more than 5 items, print the average; otherwise, display the concatenated string list in reverse order.

```
# si=input("Enter string ").split(",")
# li=[]
# ls=[]

# for i in si:
#     try:
#         li.append(int(i))
#     except:
#         ls.append(i)
# print(li)
# print(ls)
# output:
# Enter string lavanya,123,456,miranam,nagesh,8990
# [123, 456, 8990]
# ['lavanya', 'miranam', 'nagesh']
```

4. Write a program where the user is asked their age and preferred mode of transportation (e.g., car, bike, or walking). If their age is under 18, print "Focus on your studies!" regardless of transportation. If they choose "bike" and are over 60, warn them: "Take it slow!" If neither condition is met, print their input details.

```
# age=int(input("Enter age "))
# trans=input("Enter transportation ")
# if age<18:
#     print("Focus on your studies")
# elif age>60 and trans=='bike':
#     print("Take it slow")
# else:
#     print("age:",age,"Transportation:",trans)
# output:
# Enter age 21
# Enter transportation bike
# age: 21 Transportation: bike
# Enter age 61
# Enter transportation bike
# Take it slow
# Enter age 16
# Enter transportation bike
```

Focus on your studies

5. Ask the user to enter a number. Generate a sequence where each term is double the previous one, starting from 1. Stop if the next number exceeds the user's input. However, skip any number divisible by 4 in the sequence.

```
# n=int(input("Enter no "))
```

```
# n=int(input("Enter no "))
```

```
# i=1
```

```
# while((i*i)<=n):
```

```
#     r=i*i
```

```
#     if r%4==0:
```

```
#         i+=1
```

```
#         continue
```

```
#     else:
```

```
#         print(r)
```

```
#     i+=1
```

```
# output:
```

```
# 1
```

```
# 9
```

6. Write a program that takes two numbers as input. Create a multiplication table for the first number up to the second number. If the result of multiplication is a prime number, print "Prime Alert!" instead of the value.

```
# n1=int(input("Enter no1 "))
```

```
# n2=int(input("Enter no2 "))
```

```
# for i in range(1,n2+1):
```

```
#     r=n1*i
```

```
#     f=0
```

```
#     for j in range(2,r):
```

```
#         if r%j==0:
```

```
#             f=1
```

```
#     if f==1:
```

```
#         print(r)
```

```
# else:
```

```
#     print('Prime Alert')
```

```
# output:
```

```
# Enter no1 3
```

```
# Enter no2 10
```

```
# Prime Alert
```

```
# 6
```

```
# 9
```

```
# 12
```

```
# 15
```

```
# 18
```

```
# 21
```

```
# 24
```

```
# 27
```

```
# 30
```

```
# 7. Define a function that calculates the "mystery value" of a number. The mystery value is the sum of its digits raised to the power of their positions (e.g., for 123, it's  $(1^1 + 2^2 + 3^3)$ ). Write a program that takes a number as input and prints its mystery value.
```

```
# n=int(input("Enter no "))
```

```
# sum=0
```

```
# for i in range(1,len(n)+1):
```

```
#     sum=sum+(int(n[i-1]) ** i)
```

```
# print(sum)
```

```
# output:32
```

```
# 8. Ask the user for a character and a number. Print a pyramid pattern of that character up to the given number of rows. The twist: Replace all characters in odd rows with "*".
```

```
# chr=input("enter charecter ")
```

```
# n=int(input("Enter no "))
```

```
# for i in range(1,n+1):
```

```
#     for s in range(n-i):
```

```
#         print(" ",end="")
```

```
#     for c in range(1,2*i):
```

```
#         if i%2!=0:
```

```
#             print("*",end="")
```

```
#         else:
```

```
#             print(s)
```

```
#             print(chr,end="")
```

```
# print()
```

```
# output:
# enter charecter #
# Enter no 5
"""
    *

    ###
    *****

    #####
    *****
    """
```

9. Write a program that takes a list of numbers as input. Use a lambda function to filter out all numbers divisible by 3. Then, print the square of the remaining numbers using another lambda function.

```
# l=[1,2,3,4,9]
# (lambda x: x if x%3!=0)
# l1=list(filter(lambda x : x%3!=0,l))
# r=list(map(lambda x:x*x,l1))
# print(r)
# output:
# [1, 4, 16]
```

10. Create a program where the user enters a list of numbers. Use map to double each number, filter to keep only those greater than 10, and reduce to find the sum of the filtered numbers. Import only the required function(s) from functools.

```
# l=[1,2,3,4,5]
# d=list(map(lambda x:x**x,l))
# f=list(filter(lambda x:x>10,d))
# from functools import reduce
# print(d)
# print(f)
# print(reduce(lambda x,y:x+y,f))
# output:
# [1, 4, 27, 256, 3125]
# [27, 256, 3125]
# 3408
```

11. Define a global variable named `result`. Inside a function, modify the variable to store the sum of a list of numbers provided by the user. After calling the function, print the global variable to verify the change.

```
# result=0
# def fun(l):
#     global result
#     for i in l:
#         result+=i

# fun([10,20,30,40])
# print(result)
# output:
# 100
```

12. Write a function that generates a random number between 1 and 10 (without importing a library—write your own logic). Keep the generated number in a local variable. Print the random number only if the user guesses it correctly within 3 attempts.

```
# n=7

# def gess():
#     global n
#     for i in range(3):
#         u=int(input("Enter your gess "))
#         if n==u:
#             print('Gessed')
#             return
#         else:
#             print("Not correct")
#     print("Try next time")
# gess()
# output:
# Enter your gess 3
# Not correct
# Enter your gess 5
# Not correct
# Enter your gess 7
# Gessed
# Enter your gess 3
# Not correct
# Enter your gess 3
# Not correct
# Enter your gess 4
# Not correct
# Try next time
```

13. Define a global dictionary to store the number of times each vowel appears in a user-provided string. Write a function that updates the dictionary and another that displays it. The updates should persist across multiple function calls.

```
# d={'v':0}
# def vowelcount():
#     global d
#     c='aiouaeAIOUE'
#     s=input("enter string ")
#     for i in s:
#         if i in c:
#             d['v']+=1
# vowelcount()
# print(d['v'])
# d['v']=0
# vowelcount()
# print(d['v'])
# output:
# enter string lavanya
# 3
# enter string samarth
# 2
```

14. Guess the Word Game

Create a program where the user has to guess a 5-letter word you predefined. They have 7 attempts. After each guess, provide feedback: "Correct letters in the right place," "Correct letters in the wrong place," or "Completely wrong."

```
# g='lavan'
# for i in range(7):
#     s=input("Enter 5 letters string")
#     if s==g:
#         print("Correct letters in the right place")
#         break
#     else:
#         c=0
#         for i in s:
#             if i in g:
#                 c+=1
#         if c==5:
#             print("Correct letters in the wrong place")
#         else:
#             print("Completely Wrong")
```

```
# output
# Enter 5 latters string lavan
# Correct letters in the right place
# c:-----
# Enter 5 latters string lvaan
# Correct letters in the wrong place
# Enter 5 latters string kiytu
# Completly Wrong
# Enter 5 latters string vanl
# Completly Wrong
# Enter 5 latters string vaanl
# Correct letters in the wrong place
# Enter 5 latters string liiop
# Completly Wrong
# Enter 5 latters string iioop
# Completly Wrong
# Enter 5 latters string iuuiui
# Completly Wrong
```

15. Magic Matrix

```
# Write a program to accept a square matrix size and generate a "magic matrix" (where the
sum of numbers in any row, column, or diagonal is the same). The twist: If the size is less than
3, print "Too small to be magical!"
```