**# Doubts from those 100 questions: 17, 18, 40, 54, 70, 71, 75 to 81, 85, 86, 93, 94, 99, 100**

**17. Write a program to reverse a given number.**

The method we used:

b=int(input())

str1=str(b)

str1[ : :-1]

The method provided in solutions:

num = int(input("Enter a number: "))

reverse = 0

while num != 0:

reverse = reverse \* 10 + num % 10

num //= 10

print(f"Reversed number is {reverse}")

(We couldn't understand the two steps after \*while num!=0\*)

**18. Write a program to count the number of digits in a given number.**

Method we used:

c=int(input())

str2=str(c)

len(str2)

Method provided in solutions:

num = int(input("Enter a number: "))

count = 0

while num != 0:

num //= 10

count += 1

print(f"Number of digits is {count}")

(couldnt understand what num//=10 is)

**40. Write a program to check the strength of a password.**

(we couldnt understand what is going on after if condition in the following solution, what is re.search? what is r'...'?,etc.)

import re

password = input("Enter a password: ")

if len(password) < 8:

print("Password is too short")

elif not re.search(r'[a-z]', password):

print("Password must contain at least one lowercase letter")

elif not re.search(r'[A-Z]', password):

print("Password must contain at least one uppercase letter")

elif not re.search(r'[0-9]', password):

print("Password must contain at least one digit")

elif not re.search(r'[!@#$%^&\*(),.?":{}|<>]', password):

print("Password must contain at least one special character")

else:

print("Password is strong")

**(Kindly provide the solutions to the following problems):**

54. Write a program to count the number of words in a string.

58. Write a program to count the frequency of each character in a string.

70. Write a program to calculate pay based on different work shifts.

71. Write a program to print Pascal’s triangle.

75. Write a program to count the frequency of each digit in a number.

76. Write a program to convert a binary number to a decimal number.

77. Write a program to convert a decimal number to a binary number.

78. Write a program to print a number pattern.

79. Write a program to print various star patterns.

80. Write a program to create simple ASCII art using loops.

81. Write a program to count the frequency of each element in a list.

85. Write a program to invert keys and values in a dictionary.

86. Write a program to demonstrate packing and unpacking of tuples.

93. Write a program to find the uncommon elements in two lists.

94. Write a program to remove all occurrences of a specified element.

99. Write a program to generate all permutations of a list.

100. Write a program to find the longest word in a string.

**# Doubts from Python Functions exercises (50 programs): 4,8,10,21,24,26,28,32,35,36, 41 to 50**

**4.** Write a function to find the nth Fibonacci number with a default value for n as 10. (We have printed Fibonacci series successfully, but we don't know how to print the nth term as said in the question)

def fibo(n=10):

a=0

b=1

for i in range(n):

if n==1:

print(a)

break

if n==2:

print(a,b)

break

else:

print(a)

a,b=b,a+b

fibo()

**(Kindly provide solutions to the following 4 ques):**

8. Create a function that prints the current date and time with an optional format string defaulting to a standard format.

10. Write a function that generates a list of random numbers with default values for the list size and range.

22. Create a function that accepts a list and an integer and returns the element at the given position.

24. Write a function that finds the dot product of two vectors using positional arguments.

**26. Implement a function that swaps the values of two variables using positional arguments.**

(We did it in the following 2 ways):

def swap(a,b):

a,b=b,a

print(f"Swapped values:\na={a}\nb={b}")

swap(2,3)

or

def swapped(a,b):

print(f"a={b},b={a}")

swapped(2,3)

(Are these 2 methods correct? Are we supposed to do it using some other way?)

**28. Create a function that finds the median of three numbers using positional arguments.**

(We tried in the following way but some nonetype error is occuring in the 3rd line while calling the function):

def median(l1):

l2=l1.sort(reverse=True)

if (len(l2)+1)/2==l2[1]:

print(f"Median:{l2[1]}")

**32. Create a function that accepts any number of keyword arguments and prints them in a formatted string.**

(we did it this way, is it correct?)

def string(a=0,b=0):

print(f"Your formatted string:{a} {b}")

**(Kindly provide solutions to these questions):**

35. Create a function that accepts any number of dictionaries and merges them using \*\*kwargs.

36. Implement a function to print all positional and keyword arguments in a formatted string using \*args and \*\*kwargs.

39. Implement a function that finds the longest string among the given positional arguments using \*args.

40. Write a function that accepts any number of lists and returns their concatenation using \*args.

41. Create a function that accepts any number of positional arguments and returns a tuple of their squares.

42. Implement a function to filter even numbers from a list of numbers using \*args.

43. Write a function that accepts any number of keyword arguments and returns a dictionary of their uppercase keys.

44. Create a function that accepts any number of positional arguments and finds their average using \*args.

45. Implement a function that accepts any number of keyword arguments and returns a list of their keys and values.

46. Write a function that accepts any number of lists and returns a flattened list using \*args.

47. Create a function that accepts any number of keyword arguments and returns a string of their concatenated values.

48. Implement a function to find the union of multiple sets using \*args.

49. Write a function that accepts any number of positional and keyword arguments and prints them in reverse order.

50. Create a function that accepts any number of keyword arguments and returns a dictionary with keys sorted alphabetically.