1. Create three variables in a single line and assign values to them in such a manner that each one of

them belongs to a different data type.

E.g. :

a = 1,

b = 2.01,

c = 'string'

x,y,z=5,10.5,'Soumya'

print(x,y,z)

2. Create a variable of type complex and swap it with another variable of type integer.

x,y=2+3j,10

x,y=y,x

print(x)

print(y)

3. Swap two numbers using a third variable and do the same task without using any third variable.

#Using third variable

a=10

b=20

temp=a

a=b

b=temp

print(a)

print(b)

#Without using variable

a,b=10,20

a,b=b,a

print(a)

print(b)

4. Write a program that takes input from the user and prints it using both Python 2.x and Python 3.x

Version.

#Python2

x=raw\_input("Enter a value: ")

print x

#Python3

x=input("Enter a value: ")

print(x)

5. Write a program to complete the task given below:

Ask users to enter any 2 numbers in between 1-10 , add the two numbers and keep the sum in

another variable called z. Add 30 to z and store the output in variable result and print result as the

final output.

print("Enter 2 numbers between 1 to 10")

x=int(input("Input first number: "))

y=int(input("Input second number: "))

z=x+y

a=z+30

print("The final result is",a)

6. Write a program to check the data type of the entered values.

HINT: Printed output should say - The data type of the input value is : int/float/string/etc

x=eval(input("Enter a value: "))

a=type(x)

print("The data type of the input value is", a)

7. Create Variables using formats such as Upper CamelCase, Lower CamelCase, SnakeCase and

UPPERCASE.

(Refer: <https://capitalizemytitle.com/camel-case/>)

lowerCamelVariable= 100

UpperCamelVariable= 55.5

snake\_case\_variable= a

UPPERCASEVARIABLE= 2+3i

8. If one data type value is assigned to ‘a’ variable and then a different data type value is assigned to ‘a’

again. Will it change the value? If Yes then Why?

Yes, a will change the value, because a is a variable, and it can take different values assigned to it.

TASK TWO

OPERATORS AND DECISION MAKING STATEMENT

1. If a number is divisible by 3 it should print “Consultadd” as a string

If a number is divisible by 5 it should print “Python Training” as a string

If a number is divisible by both 3 and 5 it should print “Consultadd - Python Training” as a

string.

x=int(input("Enter a number: "))

if x%3==0 and x%5==0:

    print("Consultadd-Python Training")

elif x%3==0:

    print("Consultadd")

elif x%5==0:

    print("Python Training")

2. Ask user to choose the following option first:

If User Enter 1 - Addition

If User Enter 2 - Subtraction

If User Enter 3 - Division

If User Enter 4 - Multiplication

If User Enter 5 - Average

Ask user to enter two numbers and keep those numbers in variables num1 and num2

respectively for the first 4 options mentioned above.

Ask the user to enter two more numbers as first and second for calculating the average as

soon as the user chooses an option 5.

At the end if the answer of any operation is Negative print a statement saying “NEGATIVE”

NOTE: At a time a user can only perform one action.

x=int(input("Choose the operation required 1-Addition 2-subtraction 3-Division 4-Multiplication 5-Average :"))

if x not in range(1,6):

    print("Option Invalid!")

else:

    num1=int(input("Enter the first number: "))

    num2=int(input("Enter the second number: "))

    if x==1:

        y=num1+num2

    elif x==2:

        y=num1-num2

    elif x==3:

        y=num1/num2

    elif x==4:

        y=num1\*num2

    elif x==5:

        print("Enter two more numbers: ")

        first=int(input("Enter the 3rd number: "))

        second=int(input("Enter the 4th number: "))

        y=(num1+num2+first+second)/4

if y<0:

    print("NEGATIVE")

else:

    print(" The output of the selected operation",y)

1. Write a program in Python to perform the following operation:

2. Write a program in Python to perform the following operator based task:

3. Write a program in Python to implement the given flowchart:

a,b,c=10,20,30

avg=(a+b+c)/3

print("avg= ",avg)

if avg>a and avg>b and avg>c:

    print("Avg is higher than a,b,c")

elif avg>a and avg>b:

    print("Avg is greater than a,b")

elif avg>a and avg>c:

    print("Avg is greater than a,c")

elif avg>b and avg>c:

    print("Avg is greater than b,c")

elif avg>a:

    print("Avg is just higher than a")

elif avg>b:

    print("Avg is just higher than b")

elif avg>c:

    print("Avg is just higher than c")

4. Write a program in Python to break and continue if the following cases occurs:

If user enters a negative number just break the loop and print “It’s Over”

If user enters a positive number just continue in the loop and print “Good Going”

while True:

    i=int(input("Enter a number: "))

    if i<0:

        print("Its over")

        break

    elif i>0:

        print("Good going")

        continue

5. Write a program in Python which will find all such numbers which are divisible by 7 but are not a

multiple of 5, between 2000 and 3200.

x=range(2000,3201)

for i in x:

    if i%7==0 and i%5!=0:

        print(i)

6. What is the output of the following code examples?

x=123

for i in x:

print(i)

TypeError: 'int' object is not iterable

i = 0

while i < 5:

print(i)

i += 1

if i == 3:

break

else:

print(“error”)

print(“error”)

^

SyntaxError: invalid character '“' (U+201C)

count = 0

while True:

print(count)

count += 1

if count >= 5:

Break

0

1

2

3

4

Traceback (most recent call last):

File "C:\Users\kini\_\OneDrive\Desktop\New folder\soumya.py", line 10, in <module>

Break

NameError: name 'Break' is not defined

7. Write a program that prints all the numbers from 0 to 6 except 3 and 6.

Expected output: 0 1 2 4 5

Note: Use ‘continue’ statement

x=range(7)

for i in x:

    if i==3 or i==6:

        continue

    print(i)

Write a program such that it asks users to “guess the lucky number”. If the correct number is

guessed the program stops, otherwise it continues forever.

Modify the program so that it asks users whether they want to guess again each time. Use two

variables, ‘number’ for the number and ‘answer’ for the answer to the question whether they want

to continue guessing. The program stops if the user guesses the correct number or answers “no”. (

The program continues as long as a user has not answered “no” and has not guessed the correct

number)

counter=1

8. Write a program that accepts a string as an input from the user and calculate the number of digits

and letters.

Sample input: consul72

Expected output: Letters 6 Digits 2

x=input("Enter a string: ")

d=l=0

for i in x:

    if i.isdigit():

        d+=1

    elif i.isalpha():

        l+=1

    else:

        pass

print("Letters: ",l)

print("digits: ",d)

9. Read the two parts of the question below:

10. Write a program that asks five times to guess the lucky number. Use a while loop and a counter,

such as

While counter <= 5:

print(“Type in the”, counter, “number”

counter=counter+1

The program asks for five guesses (no matter whether the correct number was guessed or not). If the

correct number is guessed, the program outputs “Good guess!”, otherwise it outputs “Try again!”.

After the fifth guess it stops and prints “Game over!”.11. In the previous question, insert break after the “Good guess!” print statement. break will terminate

the while loop so that users do not have to continue guessing after they found the number. If the user

does not guess the number at all, print “Sorry but that was not very successful”