

PYTHON

ASSIGNMENT BOOK



CONSULTADD INC, TRAINING TEAM

@ 2020

TASK ONE

NUMBERS AND VARIABLES

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'

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

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

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

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.

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

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

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

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?



TASK TWO

OPERATORS AND DECISION MAKING STATEMENT

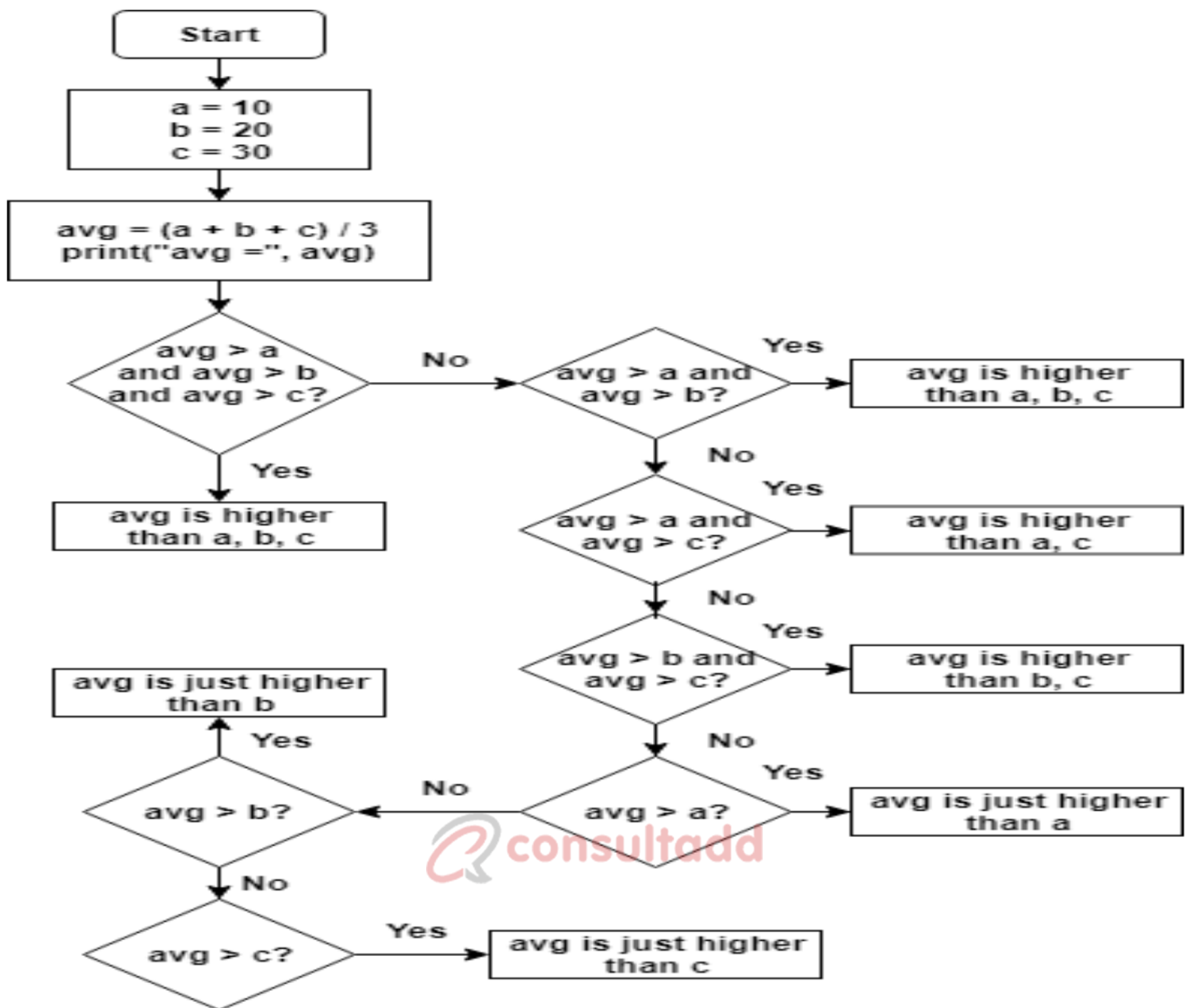
1. Write a program in Python to perform the following operation:

- 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.

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

- 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.

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



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"**

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.

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

- `x=123`

```
for i in x:  
    print(i)
```

- `i = 0`

```
while i < 5:  
    print(i)  
    i += 1  
    if i == 3:  
        break  
    else:  
        print("error")
```



- `count = 0`

```
while True:  
    print(count)  
    count += 1  
    if count >= 5:  
        Break
```

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

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

9. Read the two parts of the question below:

- 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)



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

- counter=1

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”.



Welcome to

INNOVATION WITH PYTHON

Training

Copyright © 2020 Consultadd Inc | All Rights Reserved

