Python

1. WAP to input 2 strings and swap the strings
2. WAP to generate 4 random numbers in the range 0-26 and print their average
3. WAP to generate and print a random uppercase or lowercase alphabet. Try these:
    - Create a string containing all alphabets and then select a random alphabet.
    - Check the module string
4. WAF **get_si()** that takes Principle, Rate and Time as arguments and returns the Simple Interest.
5. WAF **get_amount()** that takes Principle, Rate and Time as arguments and returns the Total amount using the get_si() function from above to calculate the SI. Also provide Rate = 10 and Time = 1 as default arguments.
6. WAP **get_ci()** that takes Principle, Rate and Time as arguments and returns the Compound Interest.
7. WAP **get_q_r()** taking 2 numbers as parameters and returns the quotient and remainder in the form of a tuple.
8. WAP to find the length of hypotenuse of a right angled triangle, input the height and base from user.
9. WAP to input number of seconds and print in days, hours, minutes and seconds
   ex: input = 10000
   Output = 0 day 2 hour 46 minute 40 second
10. Check your version of python interpreter without opening the interpreter (Which command needs to be used on the command line).
11. Find output:

```
X = 2
X *= 3
X = X%4
Y = - X
print(X,Y)
```

12. Find output:

```
def funct():
    pass

print(funct())
```

13. WAP to input the real and imaginary part of a complex number and print it on screen. output should look like

    re:10 im:20

    if the user gives input as **10** and **20.** Save the code in a script **complex.py**

14. Update **complex.py** and create a functions **input_complex()** and **print_complex().**

    [Note: DO not use the built-in complex class]

    - **input_complex()** :

      takes input from user and returns the real and imaginary part in a tuple.

    - **print_complex(re, im):**

      takes the 2 arguments real and imaginary part as arguments and prints the number on screen in the format:

      re:10 im:20

    Now use the 2 functions to input 2 complex numbers and print them on screen

15. Update **complex.py** to add another function

    - **add():**

      takes 4 arguments: (re1, im1, re2, im2) denoting the real and imaginary parts of the complex numbers to be added. The function adds and returns a single tuple denoting the resulting complex number.

      Ex: **add (10, 20, 5, -30)** should return **(15, -10).**

    - **sub():** similar to add

16. Update **complex.py** with the functions:

    - **conj(re, im):** returns the conjugate as a tuple
    - **mul(re1, im1, re2, im2):** returns the product of 2 complex numbers
    - **div(re1, im1, re2, im2):** returns the division of 2 complex numbers as a tuple.

HELP:

To Revise complex numbers you may refer:

https://medium.com/@trainer.cpp/getting-around-complex-numbers-77308982178d

For tuples run this code and observe:

```
t = (10, 20)
print( t[0] )
print( t[1] )
```