**Ronak Patel**

**DAY -25 TASK**

**TASK: OBJECT ORIENTED**

1.Write a program that calculates and prints the value according to the given formula:

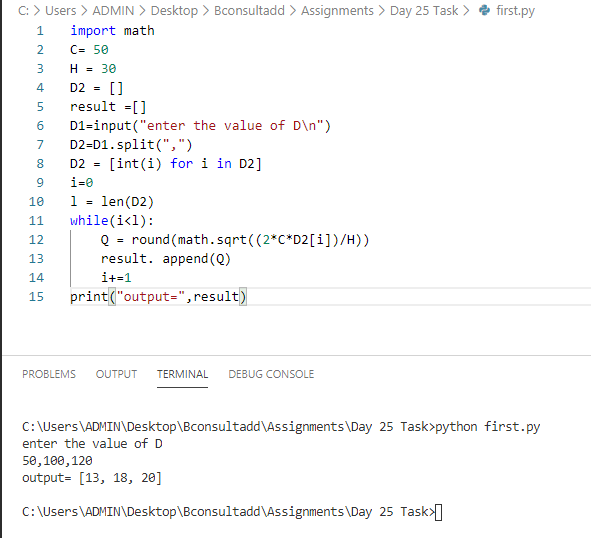
Q= Square root of [(2\*C\*D)/H]

Following are the fixed values of C and H:

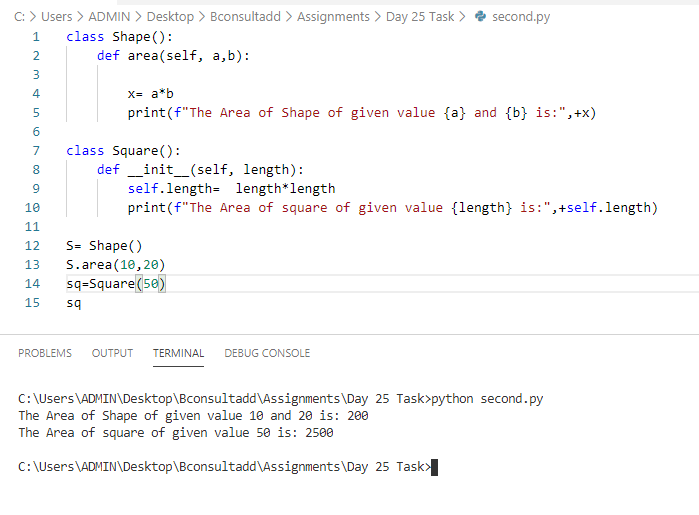
C is 50. H is 30.

D is the variable whose values should be input to your program in a comma-separated sequence.

Ans :-



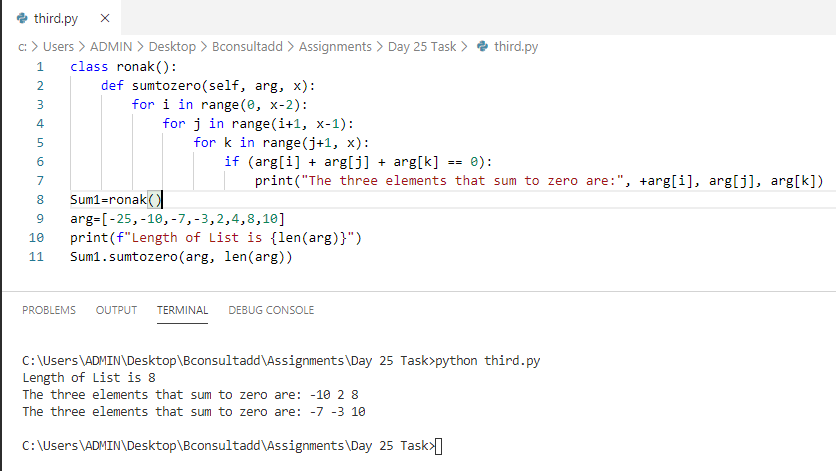
2)  Define a class named Shape and its subclass Square. The Square class has an init function which takes a length as argument. Both classes have an area function which can print the area of the shape where Shape’s area is 0 by default.



3) Create a class to find the three elements that sum to zero from a set of n real numbers.

Input array: [-25,-10,-7,-3,2,4,8,10]

Output: [[-10,2,8],[-7,-3,10]]



class ronak():

    def sumtozero(self, arg, x):

        for i in range(0, x-2):

            for j in range(i+1, x-1):

                for k in range(j+1, x):

                    if (arg[i] + arg[j] + arg[k] == 0):

                        print("The three elements that sum to zero are:", +arg[i], arg[j], arg[k])

Sum1=ronak()

arg=[-25,-10,-7,-3,2,4,8,10]

print(f"Length of List is {len(arg)}")

Sum1.sumtozero(arg, len(arg))

4)

a) class Test:

    def \_\_init\_\_(self):

        self.x = 0

class Derived\_Test(Test):

    def \_\_init\_\_(self):

        self.y = 1

def main():

    b = Derived\_Test()

    print(b.x,b.y)

main()

Output Ans :- above code will give error because there is no declaration of variable x and y

b)class A:

    def \_\_init\_\_(self, x= 1):

        self.x = x

class der(A):

    def \_\_init\_\_(self,y = 2):

        super().\_\_init\_\_()

        self.y = y

def main():

    obj = der()

    print(obj.x, obj.y)

main()

Output Ans :- output is 1 and 2 bcoz we are using super which also includes x.

c) class A:

    def \_\_init\_\_(self,x):

        self.x = x

    def count(self,x):

        self.x = self.x+1

class B(A):

    def \_\_init\_\_(self, y=0):

        A.\_\_init\_\_(self, 3)

        self.y = y

    def count(self):

        self.y += 1

def main():

    obj = B()

    obj.count()

    print(obj.x, obj.y)

main()

Output Ans :- Output is 3 and 1 because last value of X is specified as 3 and initial value of y is 0 but incrementing it to 1 whereas x value I not incremented.

d) class A:

    def \_\_init\_\_(self):

        self.multiply(15)

        print(self.i)

    def multiply(self, i):

        self.i = 4 \* i;

class B(A):

    def \_\_init\_\_(self):

        super().\_\_init\_\_()

    def multiply(self, i):

        self.i = 2 \* i;

obj = B()

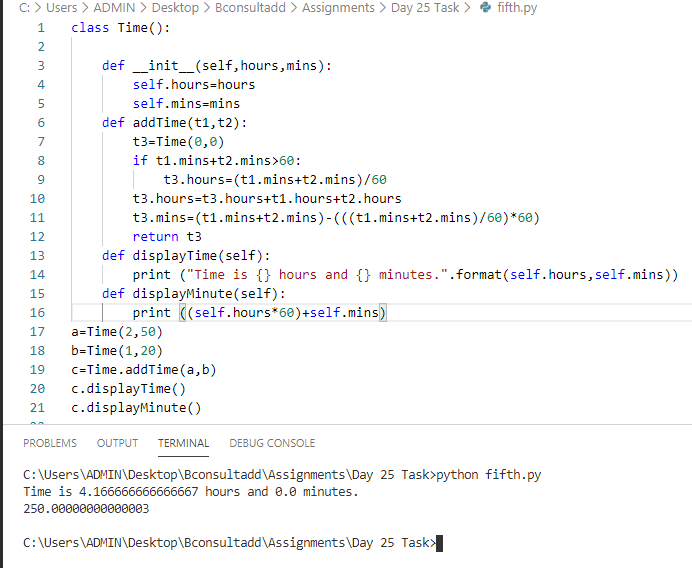
Output Ans :- o/p is 30 because class B’s multiply method is executed with i=15 i.e overriding class A.

5) Create a Time class and initialize it with hours and minutes.

Make a method addTime which should take two time object and add them. E.g.- (2 hour and 50 min)+(1 hr and 20 min) is (4 hr and 10 min)

Make a method displayTime which should print the time.

Make a method DisplayMinute which should display the total minutes in the Time. E.g.- (1 hr 2 min) should display 62 minute.



6) Write a Person class with an instance variable, , and a constructor that takes an integer, , as a parameter. The constructor must assign  to  after confirming the argument passed as  is not negative; if a negative argument is passed as , the constructor should set  to  and print Age is not valid, setting age to 0.. In addition, you must write the following instance methods:

1. yearPasses() should increase the  instance variable by .
2. amIOld()  should perform the following conditional actions:
   * If , print You are young..
   * If  and , print You are a teenager..
   * Otherwise, print You are old..

Sample Input:

4

-1

10

16

18

Sample Output:

Age is not valid, setting age to 0.

You are young.

You are young.

You are young.

You are a teenager.

You are a teenager.

You are old.

You are old.

You are old.

Ans:- class Person:

    def \_\_init\_\_(self,initialAge):

        if initialAge <0:

            self.age= 0

            print('Age is not valid, setting age to 0.')

        else:

            self.age=initialAge

    def amIOld(self):

        if self.age<13:

            print('You are young.')

        elif self.age>=13 and self.age<18:

            print('You are a teenager.')

        else:

            print('You are old.')

    def yearPasses(self):

        self.age+=1

y = int(input())

for i in range(0, y):

    age = int(input())

    x = Person(age)

    x.amIOld()

    for j in range(0, 3):

        x.yearPasses()

    x.amIOld()

    print("")

