

LAPORAN PRAKTIKUM

PEMROGRAMAN BERORIENTASI OBJEK LANJUT

2023



Prepared By:

Nama: Mohamad Maulana
Kelas: R3(C)
Nim: 210511121



Edit dengan WPS Office

Soal Praktikum:

1. Buatlah Class yang mengimplementasikan Prosedural, beri nama: celcius_pro.py
2. Buatlah Class yang mengimplementasikan Object Oriented Programming, beri nama: celcius_oop.py
3. Tulis nama, nim, dan kelas Anda di dalam Script

Jawaban berupa 2 buah screenshot script dan output tampilan hasilnya dikirim ke email : (freddy.wicaksono@umc.ac.id) dengan subject: **Praktikum-1 PBO2 2023** pada body email tuliskan NIM, Nama Lengkap, dan Kelas

(Kesalahan judul Subject bisa berakibat tidak dinilai)

1. Celcius_pro.py

```
class Celcius:
    @staticmethod
    def to_fahrenheit(celsius):
        return (celsius * 9/5) + 32

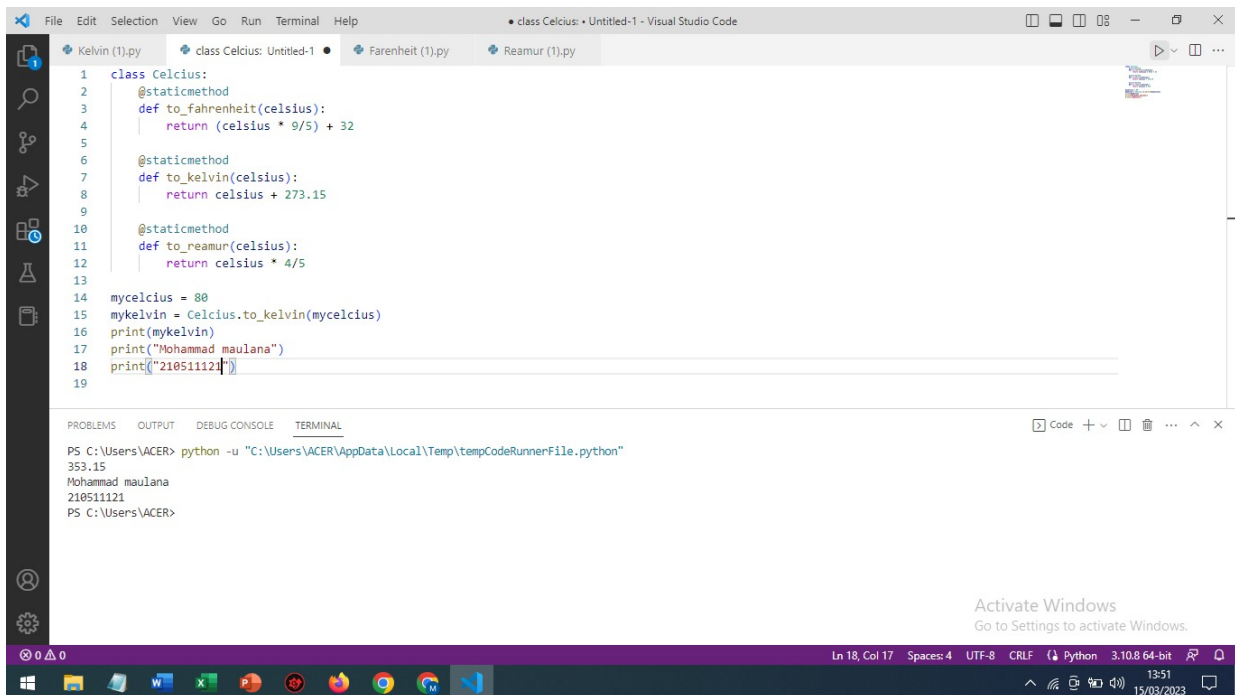
    @staticmethod
    def to_kelvin(celsius):
        return celsius + 273.15

    @staticmethod
    def to_reamur(celsius):
        return celsius * 4/5
```

```
mycelcius = 80
mykelvin = Celcius.to_kelvin(myclcius)
print(mykelvin)
print("Mohamad Maulana")
print("210511121")
```



print("R3 Teknik Informatika atau kelas C")



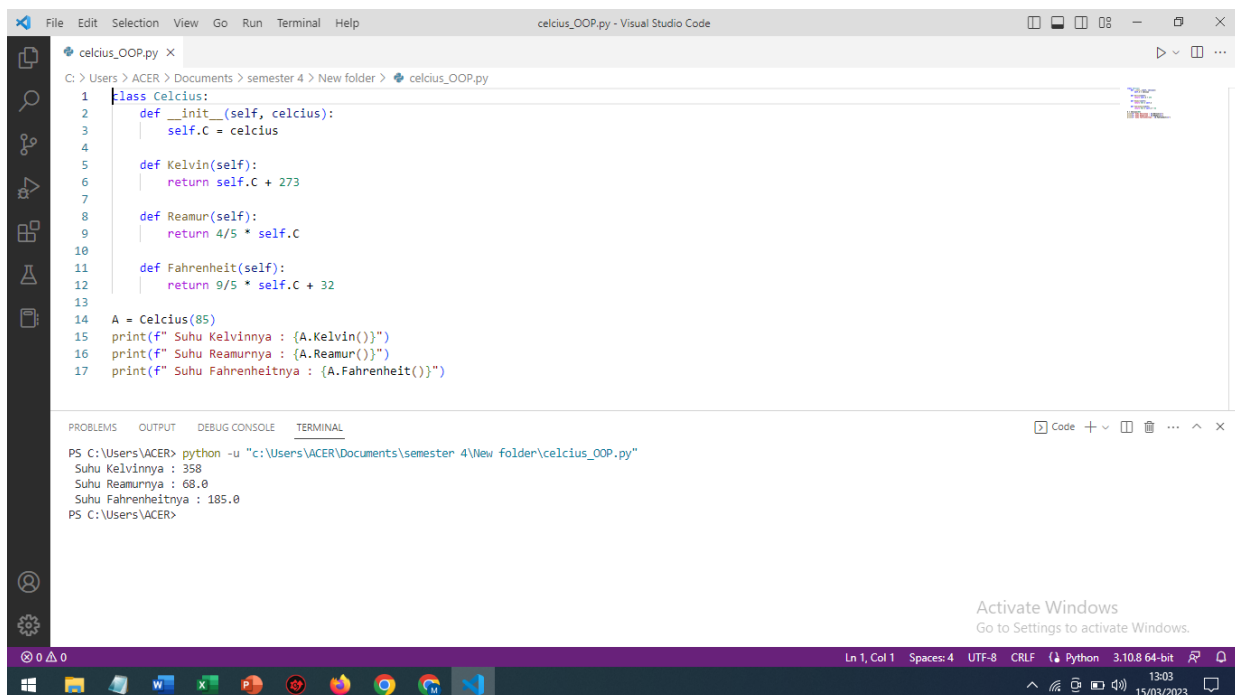
The screenshot shows a Visual Studio Code window with a file named 'class Celcius: Untitled-1'. The code defines a class 'Celcius' with three static methods: 'to_fahrenheit', 'to_kelvin', and 'to_reamur'. It also creates an instance 'mycelcius' and calls the 'to_kelvin' method. The terminal output shows the execution of the script, displaying the results of the conversions and the class name.

```
1 class Celcius:
2     @staticmethod
3     def to_fahrenheit(celsius):
4         return (celsius * 9/5) + 32
5
6     @staticmethod
7     def to_kelvin(celsius):
8         return celsius + 273.15
9
10    @staticmethod
11    def to_reamur(celsius):
12        return celsius * 4/5
13
14    mycelcius = 80
15    mykelvin = Celcius.to_kelvin(mycelcius)
16    print(mykelvin)
17    print("Mohammad Maulana")
18    print("210511121")
19
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS C:\Users\VACER> python -u "C:\Users\VACER\AppData\Local\Temp\tempCodeRunnerFile.py"
353.15
Mohammad Maulana
210511121
PS C:\Users\VACER>

2. Celcius_OOP.py



The screenshot shows a Visual Studio Code window with a file named 'celcius_OOP.py'. The code defines a class 'Celcius' with an 'init' method and three methods: 'Kelvin', 'Reamur', and 'Fahrenheit'. It creates an instance 'A' and calls the methods. The terminal output shows the execution of the script, displaying the results of the conversions and the class name.

```
1 class Celcius:
2     def __init__(self, celcius):
3         self.C = celcius
4
5     def Kelvin(self):
6         return self.C + 273
7
8     def Reamur(self):
9         return 4/5 * self.C
10
11    def Fahrenheit(self):
12        return 9/5 * self.C + 32
13
14    A = Celcius(85)
15    print(f" Suhu Kelvinnya : {A.Kelvin()}")
16    print(f" Suhu Reamurnya : {A.Reamur()}")
17    print(f" Suhu Fahrenheitnya : {A.Fahrenheit()}")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS C:\Users\VACER> python -u "c:\Users\VACER\Documents\semester 4\New folder\celcius_OOP.py"
Suhu Kelvinnya : 358
Suhu Reamurnya : 68.0
Suhu Fahrenheitnya : 185.0
PS C:\Users\VACER>



Edit dengan WPS Office

Evaluasi:

1. Apa yang dimaksud dengan class dalam Python?

C. Sebuah blueprint untuk membuat objek

2. Apa yang dimaksud dengan objek dalam Python?

B. Sebuah variabel yang dapat digunakan oleh seluruh program

3. Apa yang dimaksud dengan constructor dalam Python?

A. Sebuah method yang digunakan untuk mengubah nilai atribut objek

4. Apa yang dimaksud dengan self dalam Python?

A. Sebuah keyword yang digunakan untuk merujuk pada objek saat ini

5. Bagaimana cara membuat objek dari sebuah class?

D. Dengan menggunakan keyword "instance"



6. Apa yang dimaksud dengan atribut dalam Python?

A. Variabel yang terkait dengan sebuah objek

7. Apa yang dimaksud dengan metode dalam Python?

A. Fungsi yang terkait dengan sebuah objek

8. Bagaimana cara mengakses atribut dari sebuah objek?

A. Dengan menggunakan tanda titik (.)

9. Bagaimana cara memanggil metode dari sebuah objek?

B. Dengan menggunakan tanda kurung ()

Tugas Minggu 1:

sBuatlah 3 buah class (Fahrenheit, Reamur, dan Kelvin) yang mengimplementasikan OOPdimana setiap class memiliki kemampuan untuk melakukan konversi ke Temperatur yang lain.

Jawaban berupa 3 buah screenshot script beserta hasilnya dikirim ke email(freddy.wicaksono@umc.ac.id) dengan subject: **Tugas-1 PBO2 2023**

```
class Kelvin:
    def __init__(self, kelvin):
        self.K = kelvin

    def Celcius(self):
        return self.K - 273

    def Reamur(self):
        return 4/5 * (self.K - 273)

    def Fahrenheit(self):
        return 9/5 * (self.K - 273) + 32
```

```
A = Kelvin(69)
print(f" Suhu Celciusnya : {A.Celcius()}")
print(f" Suhu Reamurnya : {A.Reamur()}")
print(f" Suhu Fahrenheitnya : {A.Fahrenheit()}")
```



The screenshot shows a Visual Studio Code window with a Python file named `Kelvin_OOP.py`. The code defines a `Kelvin` class with methods for converting Kelvin to Celsius, Reamur, and Fahrenheit. It also creates an instance `A` of the `Kelvin` class and prints the results of these conversions.

```
1 class Kelvin:
2     def __init__(self, kelvin):
3         self.k = kelvin
4
5     def Celcius(self):
6         return self.k - 273
7
8     def Reamur(self):
9         return 4/5 * (self.k - 273)
10
11     def Fahrenheit(self):
12         return 9/5 * (self.k - 273) + 32
13
14 A = Kelvin(65)
15 print(f" Suhu Celciusnya : {A.Celcius()}")
16 print(f" Suhu Reamurnya : {A.Reamur()}")
17 print(f" Suhu Fahrenheitnya : {A.Fahrenheit()}")
```

The terminal output shows the following results:

```
PS C:\Users\VACER\Documents\semester 4\New folder> python -u "c:\Users\VACER\Downloads\Kelvin_OOP.py"
Suhu Celciusnya : -208
Suhu Reamurnya : -166.4
Suhu Fahrenheitnya : -342.40000000000003
PS C:\Users\VACER\Documents\semester 4\New folder>
```

class Fahrenheit:

```
def __init__(self, fahrenheit):
    self.f = fahrenheit
```

```
def Celcius(self):
    return 5/9 * (self.f - 32)
```

```
def Reamur(self):
    return 4/9 * (self.f - 32)
```

```
def Kelvin(self):
    return 5/9 * (self.f - 32) + 273
```

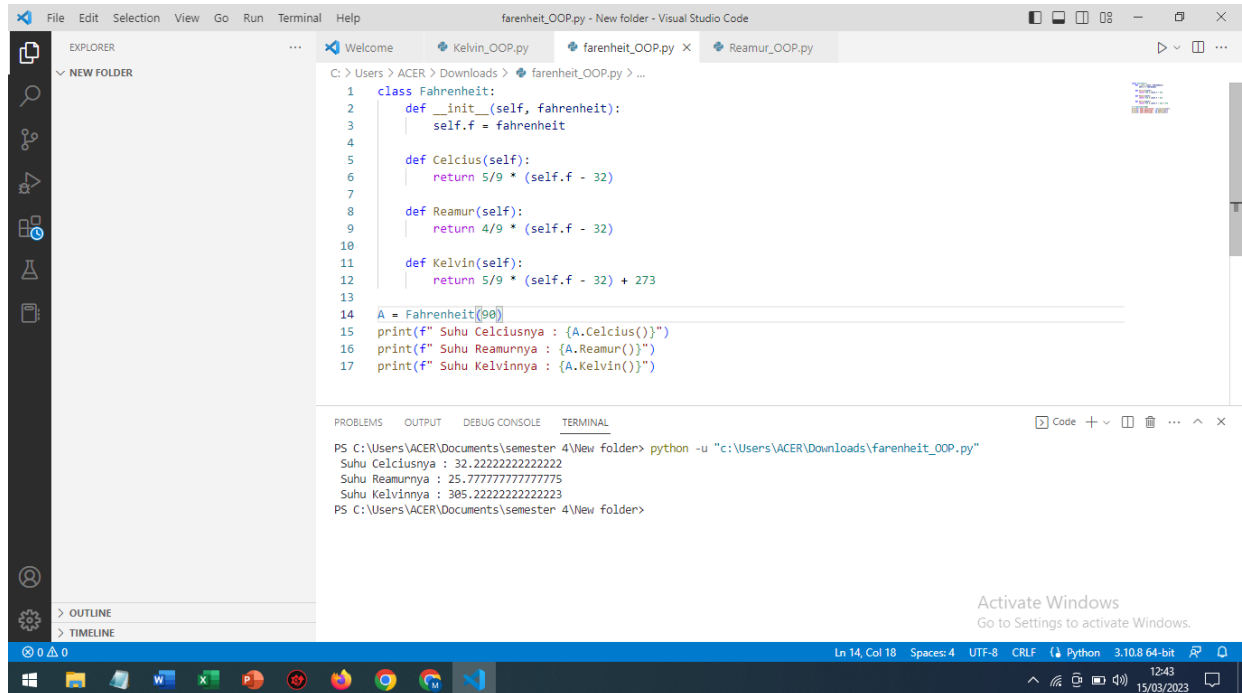
A = Fahrenheit(69)

```
print(f" Suhu Celciusnya : {A.Celcius()}")
```

```
print(f" Suhu Reamurnya : {A.Reamur()}")
```

```
print(f" Suhu Kelvinnya : {A.Kelvin()}")
```





```
1 class Fahrenheit:
2     def __init__(self, fahrenheit):
3         self.f = fahrenheit
4
5     def Celcius(self):
6         return 5/9 * (self.f - 32)
7
8     def Reamur(self):
9         return 4/9 * (self.f - 32)
10
11    def Kelvin(self):
12        return 5/9 * (self.f - 32) + 273
13
14    A = Fahrenheit(90)
15    print(f" Suhu Celciusnya : {A.Celcius()}")
16    print(f" Suhu Reamurnya : {A.Reamur()}")
17    print(f" Suhu Kelvinnya : {A.Kelvin()}")
```

```
PS C:\Users\ACER\Documents\semester 4\New folder> python -u "c:\Users\ACER\Downloads\fareinheit_OOP.py"
Suhu Celciusnya : 32.22222222222222
Suhu Reamurnya : 25.777777777777775
Suhu Kelvinnya : 305.22222222222223
PS C:\Users\ACER\Documents\semester 4\New folder>
```

```
class Reamur:
    def __init__(self, reamur):
        self.R = reamur

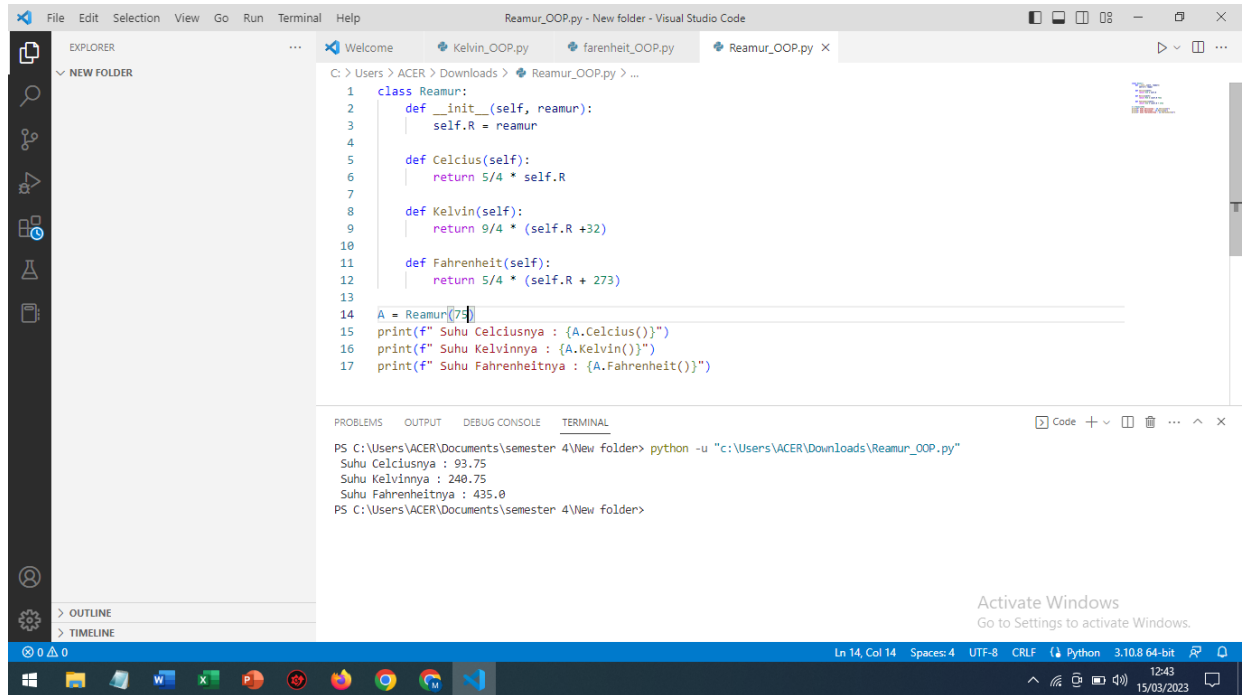
    def Celcius(self):
        return 5/4 * self.R

    def Kelvin(self):
        return 9/4 * (self.R + 32)

    def Fahrenheit(self):
        return 5/4 * (self.R + 273)

A = Reamur(69)
print(f" Suhu Celciusnya : {A.Celcius()}")
print(f" Suhu Kelvinnya : {A.Kelvin()}")
print(f" Suhu Fahrenheitnya : {A.Fahrenheit()}")
```





Edit dengan WPS Office