

LAPORAN PRAKTIKUM

PEMROGRAMAN BERORIENTASI OBJEK LANJUT

2023



Prepared By:

NAMA : FAJAR SODIK
NIM : 210511101
KELAS : R3/C

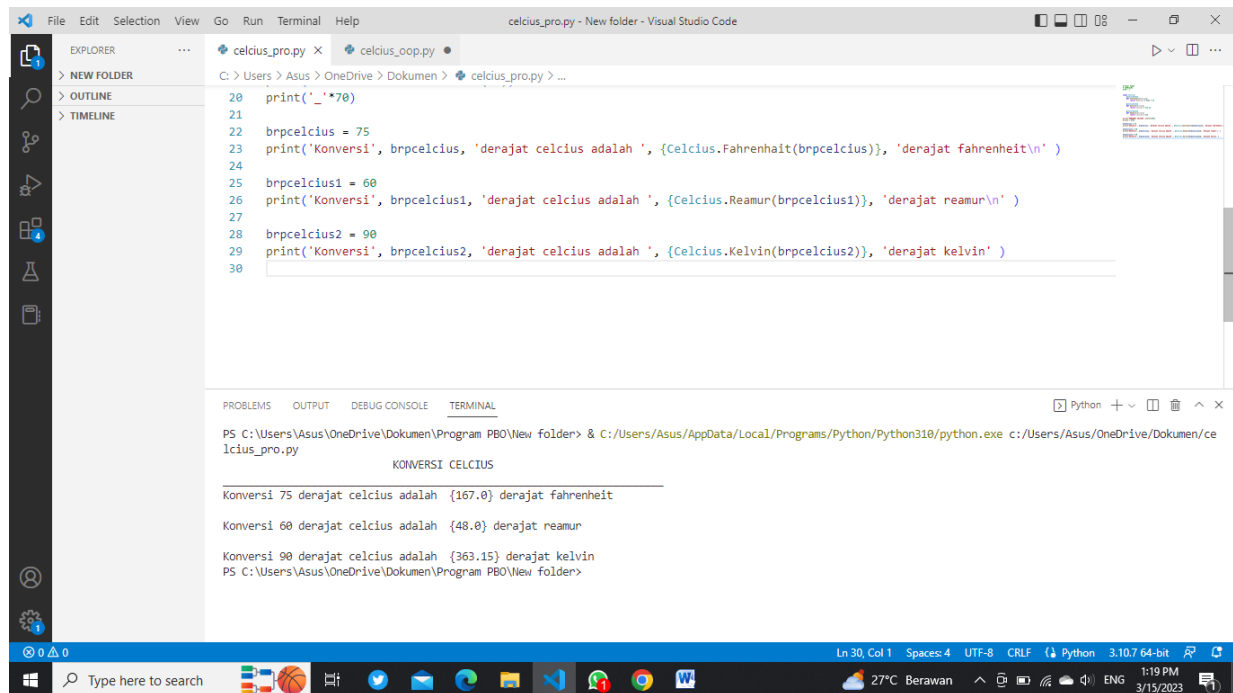
Evaluasi:

1. Apa yang dimaksud dengan class dalam Python?
D. Sebuah fungsi yang digunakan untuk mencari nilai tertentu
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?
C. Sebuah method yang digunakan untuk membuat 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?
B. Dengan menggunakan keyword "new"
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 ()

Celcius pro

```
# Fajar Sodik  
# 2105111101  
# R3 - c
```

```
class Celcius:  
    @staticmethod  
    def Fahrenheit(celcius):  
        return (celcius * 9/5) + 32  
  
    @staticmethod  
    def Kelvin(celcius):  
        return celcius + 273.15  
  
    @staticmethod  
    def Reamur(celcius):  
        return celcius * 4/5  
  
print('KONVERSI CELCIUS'.center(70))  
print('_'*70)  
  
brpcelcius = 75  
print('Konversi', brpcelcius, 'derajat celcius adalah ',  
{Celcius.Fahrenheit(brpcelcius)}, 'derajat fahrenheit\n' )  
  
brpcelcius1 = 60  
print('Konversi', brpcelcius1, 'derajat celcius adalah ',  
{Celcius.Reamur(brpcelcius1)}, 'derajat reamur\n' )  
  
brpcelcius2 = 90  
print('Konversi', brpcelcius2, 'derajat celcius adalah ',  
{Celcius.Kelvin(brpcelcius2)}, 'derajat kelvin' )
```



Celcius OOP

```
# Fajar Sodik  
# 2105111101  
# R3 - c
```

```
class Celcius:  
    def __init__(self, celcius):  
        self.celcius = celcius  
  
    def Fahrenheit(self):  
        return (self.celcius * 9/5) + 32  
  
    def Kelvin(self):  
        return self.celcius + 273.15  
  
    def Reamur(self):  
        return self.celcius * 4/5  
  
print('KONVERSI CELCIUS OOP'.center(70))  
print('_'*70)  
  
C_F = 75  
celciusA = Celcius(C_F)  
print(f'Konversi', C_F, 'derajat celcius adalah ', {celciusA.Fahrenheit()} ,  
      'derajat fahrenheit\n' )  
  
C_R = 60  
CelciusB = Celcius(C_R)  
print('Konversi', C_R , 'derajat celcius adalah ', {celciusA.Reamur()} ,  
      'derajat reamur\n' )  
  
C_K = 90  
CelciusC = Celcius(C_K)  
print('Konversi', C_K , 'derajat celcius adalah ', {celciusA.Kelvin()} ,  
      'derajat kelvin' )
```

The image shows a Visual Studio Code window with a Python file named `celcius_oop.py`. The code defines a class `Celcius` with methods for converting Celsius to Fahrenheit, Kelvin, and Reamur. The terminal shows the execution of the code, which prints the class name and then performs three conversions: 75 Celsius to Fahrenheit (167.0), 60 Celsius to Reamur (60.0), and 90 Celsius to Kelvin (348.15).

```
1 # Fajar Sodik
2 # 210511101
3 # R3 - c
4
5 class Celcius:
6     def __init__(self, celcius):
7         self.celcius = celcius
8
9     def Fahrenheit(self):
10        return (self.celcius * 9/5) + 32
11
12    def Kelvin(self):
13        return self.celcius + 273.15
14
15    def Reamur(self):
16        return self.celcius * 4/5
17
18 print('KONVERSI CELCIUS OOP'.center(70))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS C:\Users\Asus\OneDrive\Program PBO\New folder> & C:/Users/Asus/AppData/Local/Programs/Python/Python310/python.exe c:/Users/Asus/OneDrive/Dokumen/celcius_oop.py

KONVERSI CELCIUS OOP

Konversi 75 derajat celcius adalah {167.0} derajat fahrenheit

Konversi 60 derajat celcius adalah {60.0} derajat reamur

Konversi 90 derajat celcius adalah {348.15} derajat kelvin

PS C:\Users\Asus\OneDrive\Program PBO\New folder>

Ln 4, Col 1 Spaces: 4 UTF-8 CRLF Python 3.10.7 64-bit

27°C Berawan 1:22 PM 3/15/2023