

LAPORAN PRAKTIKUM

PEMROGRAMAN BERORIENTASI OBJEK LANJUT

2023



Prepared By:

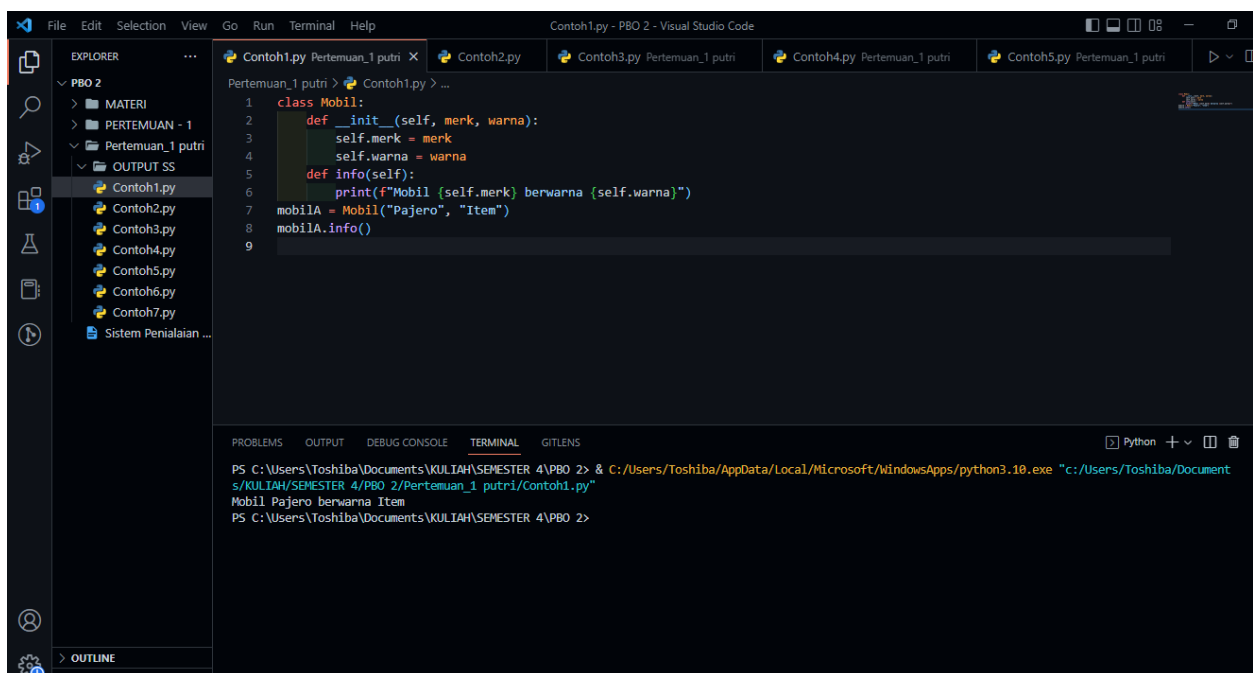
PUTRI

NIM. 210511068

Contoh 1 :

```
class Mobil:
    def __init__(self, merk, warna):
        self.merk = merk
        self.warna = warna
    def info(self):
        print(f"Mobil {self.merk} berwarna {self.warna}")
mobilA = Mobil("Pajero", "Item")
mobilA.info()
```

Output Contoh 1



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left shows a project structure with a folder 'PERTEMUAN - 1' containing a subfolder 'Pertemuan_1 putri' with files 'Contoh1.py' through 'Contoh7.py'. The main editor window displays the code for 'Contoh1.py', which is identical to the code in the first block. The TERMINAL panel at the bottom shows the command prompt output: 'Mobil Pajero berwarna Item'.

Contoh 2 :

```
class Mahasiswa:
    def __init__(self, nama, npm):
        self.nama = nama
        self.npm = npm
    def info(self):
        print(f>Nama: {self.nama}\nNIM: {self.npm}")
mahasiswaB = Mahasiswa("Putri", "210511068")
mahasiswaB.info()
```

Output Contoh 2

Contoh 3 :

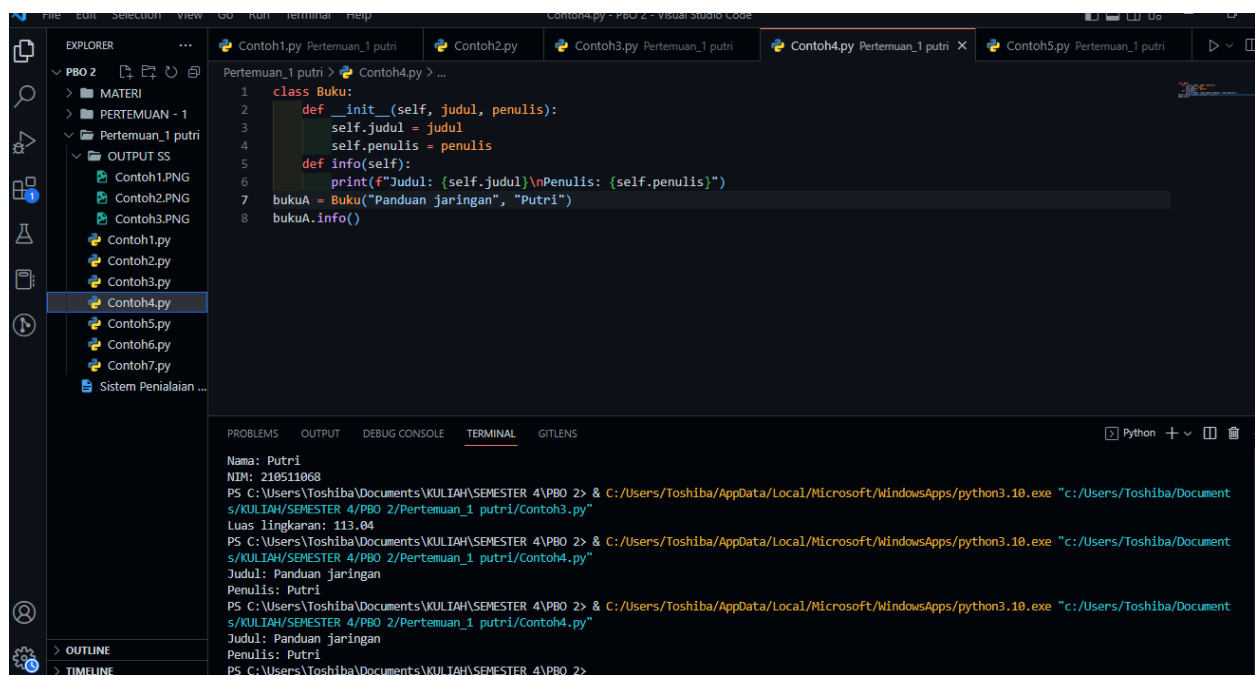
```
class Lingkaran:
    def __init__(self, jari_jari):
        self.jari_jari = jari_jari
    def luas(self):
        return 3.14 * (self.jari_jari ** 2)
lingkaranA = Lingkaran(6)
print(f"Luas lingkaran: {lingkaranA.luas()}")
```

Output Contoh 3

Contoh 4 :

```
class Buku:
    def __init__(self, judul, penulis):
        self.judul = judul
        self.penulis = penulis
    def info(self):
        print(f"Judul: {self.judul}\nPenulis: {self.penulis}")
bukuA = Buku("Panduan jaringan", "Putri")
bukuA.info()
```

Output 4



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left shows a project structure with folders 'MATERI', 'PERTEMUAN - 1', and 'OUTPUT SS'. The file 'Contoh4.py' is selected. The main editor displays the Python code from the previous block. The TERMINAL panel at the bottom shows the output of running the script, which includes the file path and the printed information for the 'bukuA' object.

```
1 class Buku:
2     def __init__(self, judul, penulis):
3         self.judul = judul
4         self.penulis = penulis
5     def info(self):
6         print(f"Judul: {self.judul}\nPenulis: {self.penulis}")
7 bukuA = Buku("Panduan jaringan", "Putri")
8 bukuA.info()
```

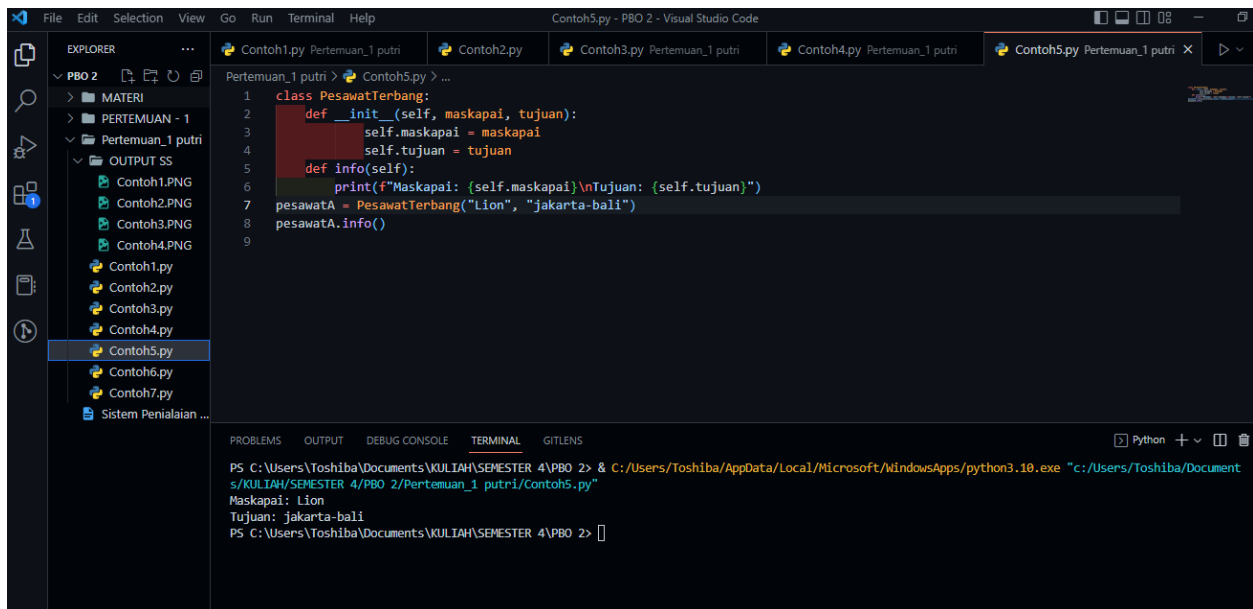
TERMINAL

```
PS C:\Users\Toshiba\Documents\KULIAH\SEMESTER 4\PRO 2> & C:/Users/Toshiba/AppData/Local/Microsoft/WindowsApps/python3.10.exe "c:/Users/Toshiba/Document
s/KULIAH/SEMESTER 4/PRO 2/Peremuan_1 putri/Contoh3.py"
Luas lingkaran: 113.04
PS C:\Users\Toshiba\Documents\KULIAH\SEMESTER 4\PRO 2> & C:/Users/Toshiba/AppData/Local/Microsoft/WindowsApps/python3.10.exe "c:/Users/Toshiba/Document
s/KULIAH/SEMESTER 4/PRO 2/Peremuan_1 putri/Contoh4.py"
Judul: Panduan jaringan
Penulis: Putri
PS C:\Users\Toshiba\Documents\KULIAH\SEMESTER 4\PRO 2> & C:/Users/Toshiba/AppData/Local/Microsoft/WindowsApps/python3.10.exe "c:/Users/Toshiba/Document
s/KULIAH/SEMESTER 4/PRO 2/Peremuan_1 putri/Contoh4.py"
Judul: Panduan jaringan
Penulis: Putri
PS C:\Users\Toshiba\Documents\KULIAH\SEMESTER 4\PRO 2>
```

Contoh 5 :

```
class PesawatTerbang:
    def __init__(self, maskapai, tujuan):
        self.maskapai = maskapai
        self.tujuan = tujuan
    def info(self):
        print(f"Maskapai: {self.maskapai}\nTujuan: {self.tujuan}")
pesawatA = PesawatTerbang("Lion", "jakarta-bali")
pesawatA.info()
```

Output 5



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left shows a project structure with folders 'MATERI', 'PERTEMUAN - 1', and 'Pertemuan_1 putri'. Under 'Pertemuan_1 putri', there is an 'OUTPUT SS' folder containing several PNG files and a 'Contoh5.py' file. The main editor displays the code for 'Contoh5.py', which defines a class 'PesawatTerbang' with an '__init__' method and an 'info' method. The code creates an instance 'pesawataA' and calls its 'info' method. The terminal at the bottom shows the command to run the script, which outputs 'Maskapai: Lion' and 'Tujuan: jakarta-bali'.

```
1 class PesawatTerbang:
2     def __init__(self, maskapai, tujuan):
3         self.maskapai = maskapai
4         self.tujuan = tujuan
5     def info(self):
6         print(f"Maskapai: {self.maskapai}\nTujuan: {self.tujuan}")
7 pesawataA = PesawatTerbang("Lion", "jakarta-bali")
8 pesawataA.info()
9
```

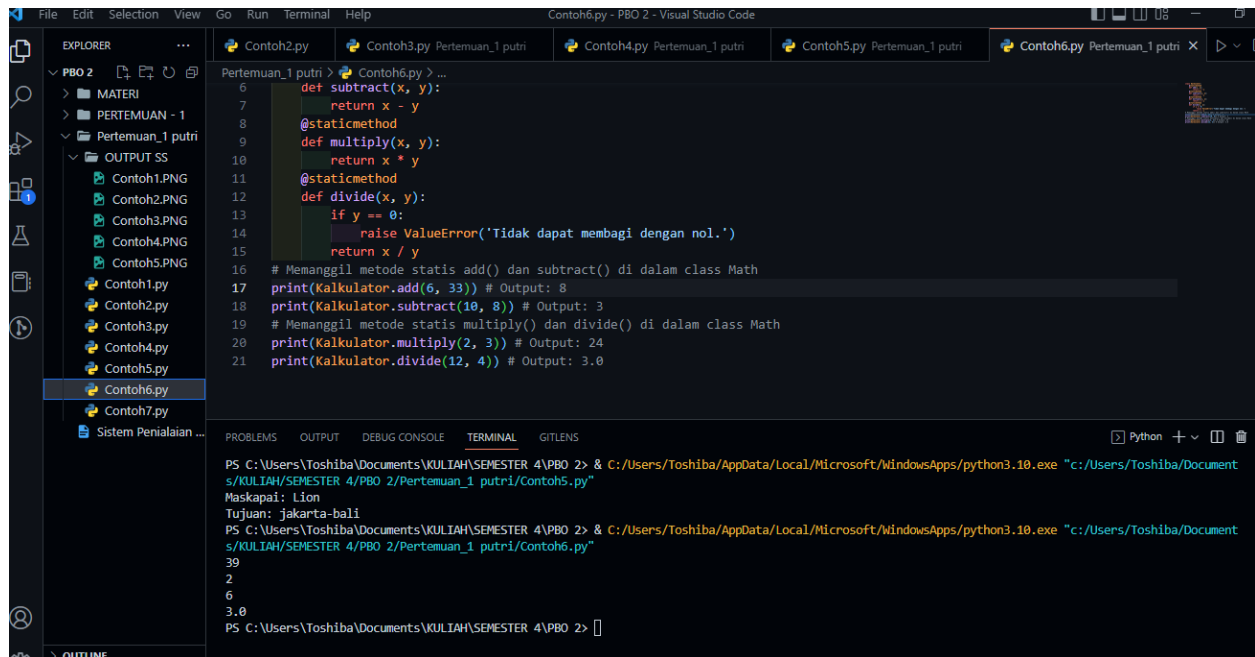
```
PS C:\Users\Toshiba\Documents\KULIAH\SEMESTER 4\PBO 2> & C:/Users/Toshiba/AppData/Local/Microsoft/WindowsApps/python3.10.exe "c:/Users/Toshiba/Document
s/KULIAH/SEMESTER 4/PBO 2/Pertemuan_1 putri/Contoh5.py"
Maskapai: Lion
Tujuan: jakarta-bali
PS C:\Users\Toshiba\Documents\KULIAH\SEMESTER 4\PBO 2> []
```

Contoh 6:

```
class Kalkulator:
    @staticmethod
    def add(x, y):
        return x + y
    @staticmethod
    def subtract(x, y):
        return x - y
    @staticmethod
    def multiply(x, y):
        return x * y
    @staticmethod
    def divide(x, y):
        if y == 0:
            raise ValueError('Tidak dapat membagi dengan nol.')
        return x / y

# Memanggil metode statis add() dan subtract() di dalam class Math
print(Kalkulator.add(6, 33)) # Output: 38
print(Kalkulator.subtract(10, 8)) # Output: 2
# Memanggil metode statis multiply() dan divide() di dalam class Math
print(Kalkulator.multiply(2, 3)) # Output: 6
print(Kalkulator.divide(12, 4)) # Output: 3.0
```

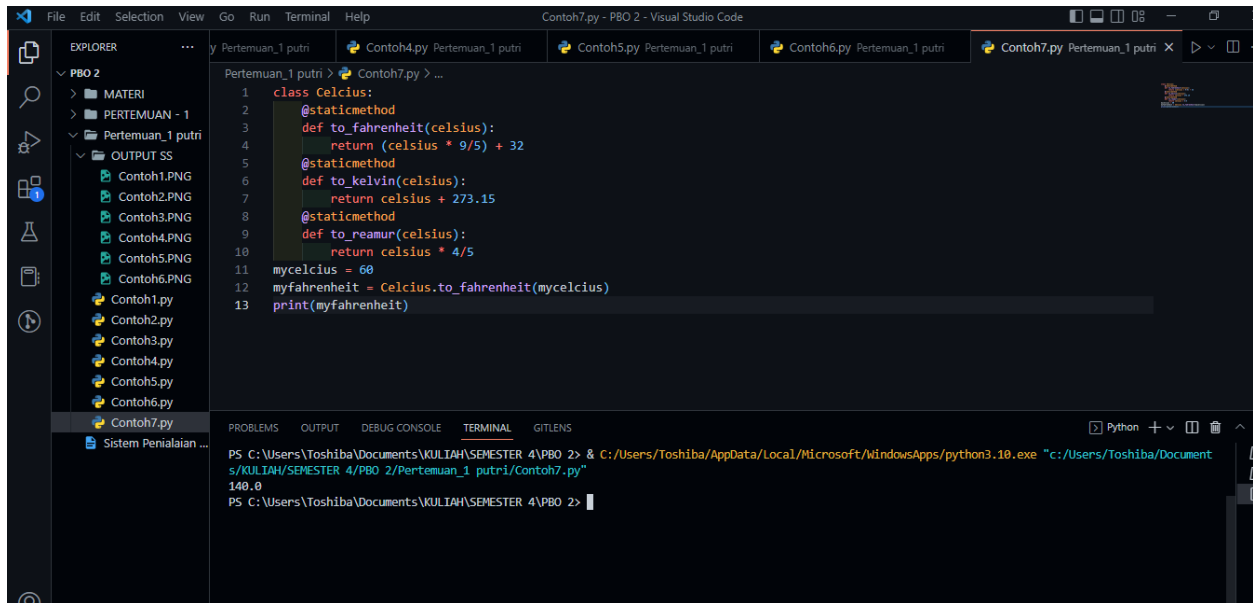
Output 6



Contoh 7 :

```
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 = 60
myfahrenheit = Celcius.to_fahrenheit(mycelcius)
print(myfahrenheit)
```

Output 7



The screenshot displays the Visual Studio Code interface with a Python file named `Contoh7.py` open. The file contains a class `Celcius` with static methods for temperature conversion. The Explorer sidebar on the left shows a project structure with folders `PBO 2` and `PERTEMUAN - 1`, and a file `Contoh7.py` under the `OUTPUT SS` folder. The Terminal at the bottom shows the command to run the script, which outputs the value `140.0`.

```
1 class Celcius:
2     @staticmethod
3     def to_fahrenheit(celsius):
4         return (celsius * 9/5) + 32
5     @staticmethod
6     def to_kelvin(celsius):
7         return celsius + 273.15
8     @staticmethod
9     def to_reamur(celsius):
10        return celsius * 4/5
11 mycelcius = 60
12 myfahrenheit = Celcius.to_fahrenheit(mycelcius)
13 print(myfahrenheit)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS

Python + -

PS C:\Users\Toshiba\Documents\KULIAH\SEMESTER 4\PBO 2> & C:/Users/Toshiba/AppData/Local/Microsoft/WindowsApps/python3.10.exe "c:/Users/Toshiba/Document
s/KULIAH/SEMESTER 4/PBO 2/Pertemuan_1_putri/Contoh7.py"
140.0
PS C:\Users\Toshiba\Documents\KULIAH\SEMESTER 4\PBO 2>

LATIHAN :

Script Codingan

```
class TemperatureConverter:
    def __init__(self, celsius):
        self.celsius = celsius

    def to_reamur(self):
        return (4/5) * self.celsius

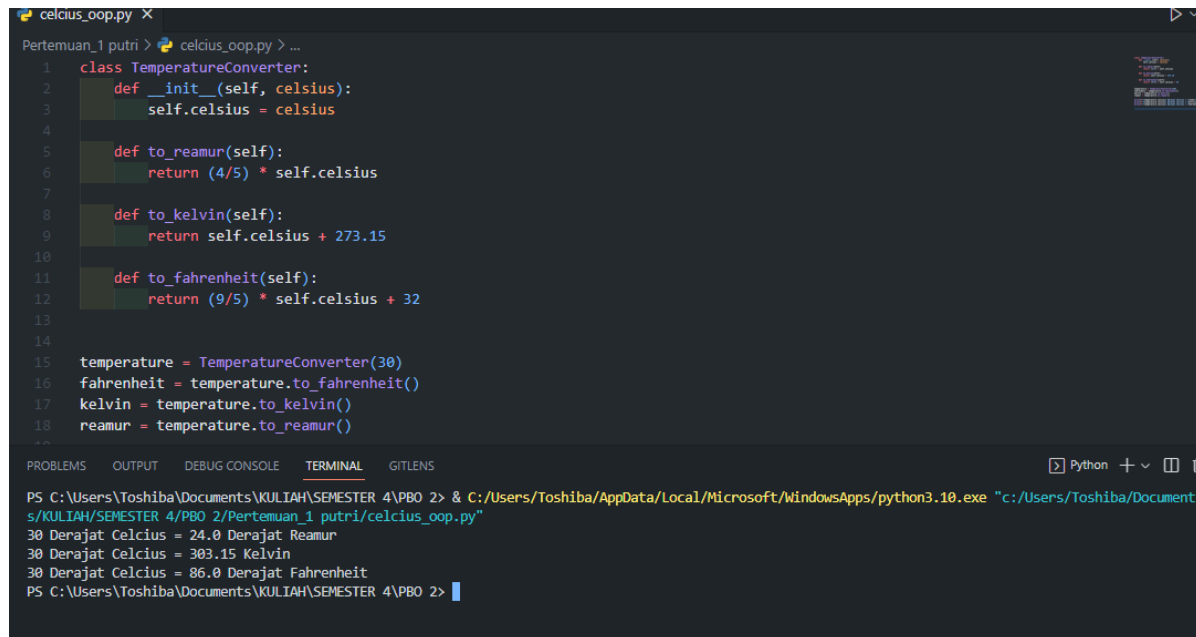
    def to_kelvin(self):
        return self.celsius + 273.15

    def to_fahrenheit(self):
        return (9/5) * self.celsius + 32

temperature = TemperatureConverter(30)
fahrenheit = temperature.to_fahrenheit()
kelvin = temperature.to_kelvin()
reamur = temperature.to_reamur()

print(f"{temperature.celsius} Derajat Celcius = {reamur} Derajat Reamur")
print(f"{temperature.celsius} Derajat Celcius = {kelvin} Kelvin")
print(f"{temperature.celsius} Derajat Celcius = {fahrenheit} Derajat Fahrenheit")
```


Output Screenshot



```
celcius_oop.py X
Pertemuan_1 putri > celcius_oop.py > ...
1 class TemperatureConverter:
2     def __init__(self, celsius):
3         self.celsius = celsius
4
5     def to_reamur(self):
6         return (4/5) * self.celsius
7
8     def to_kelvin(self):
9         return self.celsius + 273.15
10
11    def to_fahrenheit(self):
12        return (9/5) * self.celsius + 32
13
14
15    temperature = TemperatureConverter(30)
16    fahrenheit = temperature.to_fahrenheit()
17    kelvin = temperature.to_kelvin()
18    reamur = temperature.to_reamur()
19
20
PS C:\Users\Toshiba\Documents\KULIAH\SEMESTER 4\PBO 2> & C:/Users/Toshiba/AppData/Local/Microsoft/WindowsApps/python3.10.exe "c:/Users/Toshiba/Document
s/KULIAH/SEMESTER 4/PBO 2/Pertemuan_1 putri/celcius_oop.py"
30 Derajat Celcius = 24.0 Derajat Reamur
30 Derajat Celcius = 303.15 Kelvin
30 Derajat Celcius = 86.0 Derajat Fahrenheit
PS C:\Users\Toshiba\Documents\KULIAH\SEMESTER 4\PBO 2>
```

Evaluasi:

1. Apa yang dimaksud dengan class dalam Python?
 - a. Sebuah fungsi yang mengembalikan nilai tertentu
 - b. Sebuah variabel yang dapat digunakan oleh seluruh program
 - c. Sebuah blueprint untuk membuat objek
 - d. Sebuah fungsi yang digunakan untuk mencari nilai tertentu

JAWAB : C. SEBUAH BLUEPRINT UNTUK MEMBUAT OBJEK

2. Apa yang dimaksud dengan objek dalam Python?
 - a. Sebuah fungsi yang mengembalikan nilai tertentu
 - b. Sebuah variabel yang dapat digunakan oleh seluruh program
 - c. Sebuah blueprint untuk membuat class
 - d. Sebuah instance dari sebuah class

JAWAB : D. SEBUAH INSTANCE DARI SEBUAH CLASS

3. Apa yang dimaksud dengan constructor dalam Python?

- a. Sebuah method yang digunakan untuk mengubah nilai atribut objek
- b. Sebuah method yang digunakan untuk menghapus objek
- c. Sebuah method yang digunakan untuk membuat objek
- d. Sebuah method yang digunakan untuk mengambil nilai atribut objek

JAWAB : 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
- b. Sebuah keyword yang digunakan untuk merujuk pada class saat ini
- c. Sebuah keyword yang digunakan untuk merujuk pada konstruktor saat ini
- d. Sebuah keyword yang digunakan untuk merujuk pada method saat ini

JAWAB : A. SEBUAH KEYWORD YANG DIGUNAKAN UNTUK MERUJUK PADA OBJEK SAAT INI

5. Bagaimana cara membuat objek dari sebuah class?

- A. Dengan menggunakan fungsi init
- B. Dengan menggunakan keyword "new"
- C. Dengan menggunakan fungsi "create"
- D. Dengan menggunakan keyword "instance"

JAWAB : A. DENGAN MENGGUNAKAN FUNGSI INIT

6. Apa yang dimaksud dengan atribut dalam Python?

- A. Variabel yang terkait dengan sebuah objek
- B. Metode yang terkait dengan sebuah objek
- C. Fungsi yang digunakan untuk mengembalikan nilai
- D. Sebuah tipe data yang menyimpan nilai

JAWAB : A. VARIABEL YANG TERKAIT DENGAN SEBUAH OBJEK

7. Apa yang dimaksud dengan metode dalam Python?

- A. Fungsi yang terkait dengan sebuah objek
- B. Variabel yang terkait dengan sebuah objek
- C. Fungsi yang digunakan untuk mengembalikan nilai
- D. Sebuah tipe data yang menyimpan nilai

JAWAB : A. FUNGSI YANG TERKAIT DENGAN SEBUAH OBJEK

8. Bagaimana cara mengakses atribut dari sebuah objek?

- A. Dengan menggunakan tanda titik (.)
- B. Dengan menggunakan tanda kurung []
- C. Dengan menggunakan tanda koma ,
- D. Dengan menggunakan tanda panah ->

JAWAB : A. DENGAN MENGGUNAKAN TANDA TITIK (.)

9. Bagaimana cara memanggil metode dari sebuah objek?

- A. Dengan menggunakan tanda titik (.)
- B. Dengan menggunakan tanda kurung ()
- C. Dengan menggunakan tanda koma ,
- D. Dengan menggunakan tanda panah ->

JAWAB : A. DENGAN MENGGUNAKAN TANDA TITIK (.)