

LAPORAN PRAKTIKUM

PEMROGRAMAN BERORIENTASI OBJEK LANJUT

2023



Prepared By:

M.Muchibbulloh

R2(B)

210511078

Tugas Praktikum 7 :

Buatlah 3 aplikasi untuk menghitung volume dan luas permukaan selain dari contoh diatas menggunakan teknik Metaprogramming. Hasilnya diupload ke github masing-masing.

Jawaban :

1. Kubus

```
class Kubus:
    def __init__(self, sisi):
        self.sisi = sisi

    def volume(self):
        return self.sisi ** 3

    def luas(self):
        return 6 * self.sisi ** 2

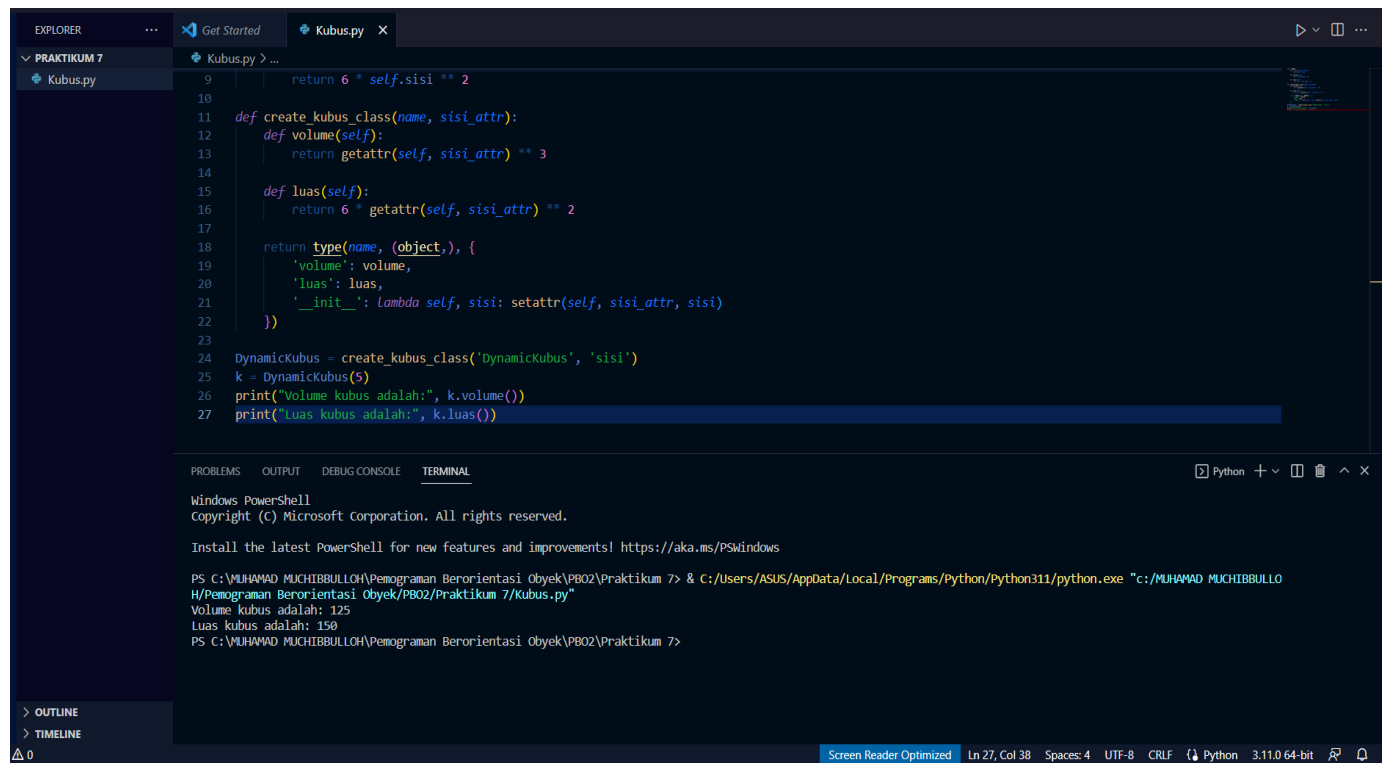
def create_kubus_class(name, sisi_attr):
    def volume(self):
        return getattr(self, sisi_attr) ** 3

    def luas(self):
        return 6 * getattr(self, sisi_attr) ** 2

    return type(name, (object,), {
        'volume': volume,
        'luas': luas,
        '__init__': lambda self, sisi: setattr(self, sisi_attr, sisi)
    })

DynamicKubus = create_kubus_class('DynamicKubus', 'sisi')
k = DynamicKubus(5)
print("Volume kubus adalah:", k.volume())
print("Luas kubus adalah:", k.luas())
```

Output :



The screenshot shows a Visual Studio Code editor window with a file named `Kubus.py` open. The code defines a `create_kubus_class` function that takes a class name and a parameter name, and returns a class with `volume` and `luas` methods. The `volume` method calculates the volume of a cube (side length cubed), and the `luas` method calculates the surface area (6 times side length squared). The script then creates a `DynamicKubus` class and instantiates it with `5`. It prints the volume and surface area of the instance.

```
9         return 6 * self.sisi ** 2
10
11     def create_kubus_class(name, sisi_attr):
12         def volume(self):
13             return getattr(self, sisi_attr) ** 3
14
15         def luas(self):
16             return 6 * getattr(self, sisi_attr) ** 2
17
18         return type(name, (object,), {
19             'volume': volume,
20             'luas': luas,
21             '__init__': lambda self, sisi: setattr(self, sisi_attr, sisi)
22         })
23
24     DynamicKubus = create_kubus_class('DynamicKubus', 'sisi')
25     k = DynamicKubus(5)
26     print("Volume kubus adalah:", k.volume())
27     print("Luas kubus adalah:", k.luas())
```

The terminal output shows the execution of the script, displaying the volume and surface area of the cube.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\MUHAMMAD MUCHIBULLOH\Pemogramran Berorientasi Obyek\PB02\Praktikum 7> & C:/Users/ASUS/AppData/Local/Programs/Python/Python311/python.exe "c:\MUHAMMAD MUCHIBULLO
H\Pemogramran Berorientasi Obyek\PB02\Praktikum 7\kubus.py"
Volume kubus adalah: 125
Luas kubus adalah: 150
PS C:\MUHAMMAD MUCHIBULLOH\Pemogramran Berorientasi Obyek\PB02\Praktikum 7>
```

The status bar at the bottom indicates the file is at line 27, column 38, with 4 spaces, UTF-8 encoding, CRLF line endings, and is running Python 3.11.0 64-bit.

2. Bola :

```
import math

class Bola:
    def __init__(self, r):
        self.r = r

    def volume(self):
        return 4/3 * math.pi * self.r ** 3

    def luas(self):
        return 4 * math.pi * self.r ** 2

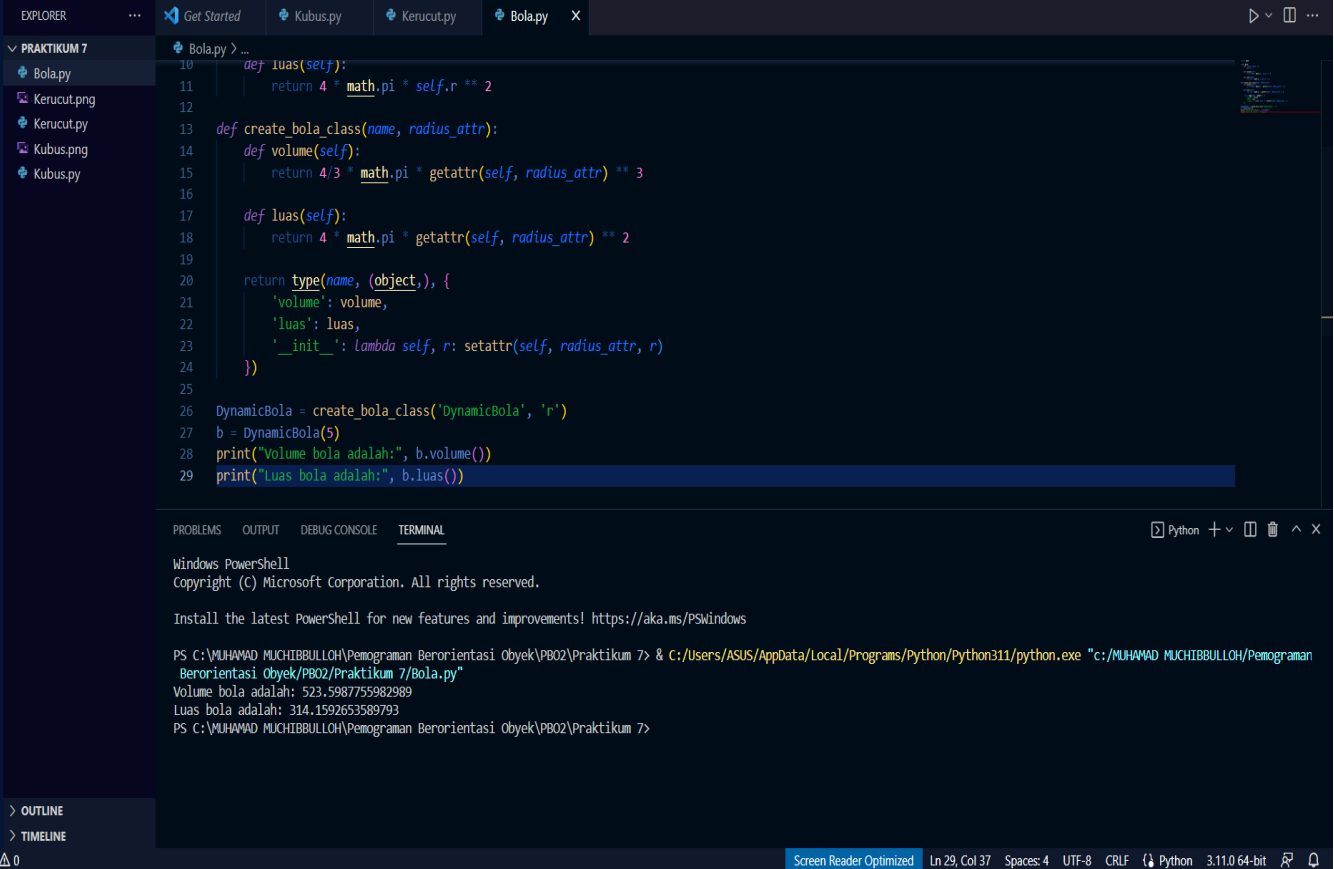
def create_bola_class(name, radius_attr):
    def volume(self):
        return 4/3 * math.pi * getattr(self, radius_attr) ** 3

    def luas(self):
        return 4 * math.pi * getattr(self, radius_attr) ** 2

    return type(name, (object,), {
        'volume': volume,
        'luas': luas,
        '__init__': lambda self, r: setattr(self, radius_attr, r)
    })

DynamicBola = create_bola_class('DynamicBola', 'r')
b = DynamicBola(5)
print("Volume bola adalah:", b.volume())
print("Luas bola adalah:", b.luas())
```

Output :



The image shows a Visual Studio Code editor window with a Python file named `Bola.py` open. The file contains a class `DynamicBola` with methods `volume` and `luas` (area). The `__init__` method takes a radius `r` and sets it as an attribute. The `volume` method calculates the volume of a sphere using the formula $V = \frac{4}{3} \pi r^3$, and the `luas` method calculates the surface area using $A = 4 \pi r^2$. The script creates an instance `b` of `DynamicBola` with a radius of 5 and prints the volume and area.

```
10 def luas(self):
11     return 4 * math.pi * self.r ** 2
12
13 def create_bola_class(name, radius_attr):
14     def volume(self):
15         return 4/3 * math.pi * getattr(self, radius_attr) ** 3
16
17     def luas(self):
18         return 4 * math.pi * getattr(self, radius_attr) ** 2
19
20     return type(name, (object,), {
21         'volume': volume,
22         'luas': luas,
23         '__init__': lambda self, r: setattr(self, radius_attr, r)
24     })
25
26 DynamicBola = create_bola_class('DynamicBola', 'r')
27 b = DynamicBola(5)
28 print("Volume bola adalah:", b.volume())
29 print("Luas bola adalah:", b.luas())
```

The terminal output shows the execution of the script, displaying the volume and area of the sphere:

```
PS C:\MUHAMMAD MUCHIBULLOH\Pemrograman Berorientasi Obyek\p802\Praktikum 7> & C:/Users/ASUS/AppData/Local/Programs/Python/Python311/python.exe "c:/MUHAMMAD MUCHIBULLOH/Pemrograman Berorientasi Obyek/p802/Praktikum 7/Bola.py"
Volume bola adalah: 523.5987755982989
Luas bola adalah: 314.1592653589793
PS C:\MUHAMMAD MUCHIBULLOH\Pemrograman Berorientasi Obyek\p802\Praktikum 7>
```

The status bar at the bottom indicates the file is at line 29, column 37, with 4 spaces, UTF-8 encoding, and CRLF line endings. It also shows the Python version as 3.11.0 64-bit.

3. Kerucut :

```
import math

class Kerucut:
    def __init__(self, r, t):
        self.r = r
        self.t = t

    def volume(self):
        return math.pi * self.r ** 2 * self.t / 3

    def luas(self):
        s = math.sqrt(self.r ** 2 + self.t ** 2)
        return math.pi * self.r * s + math.pi * self.r ** 2

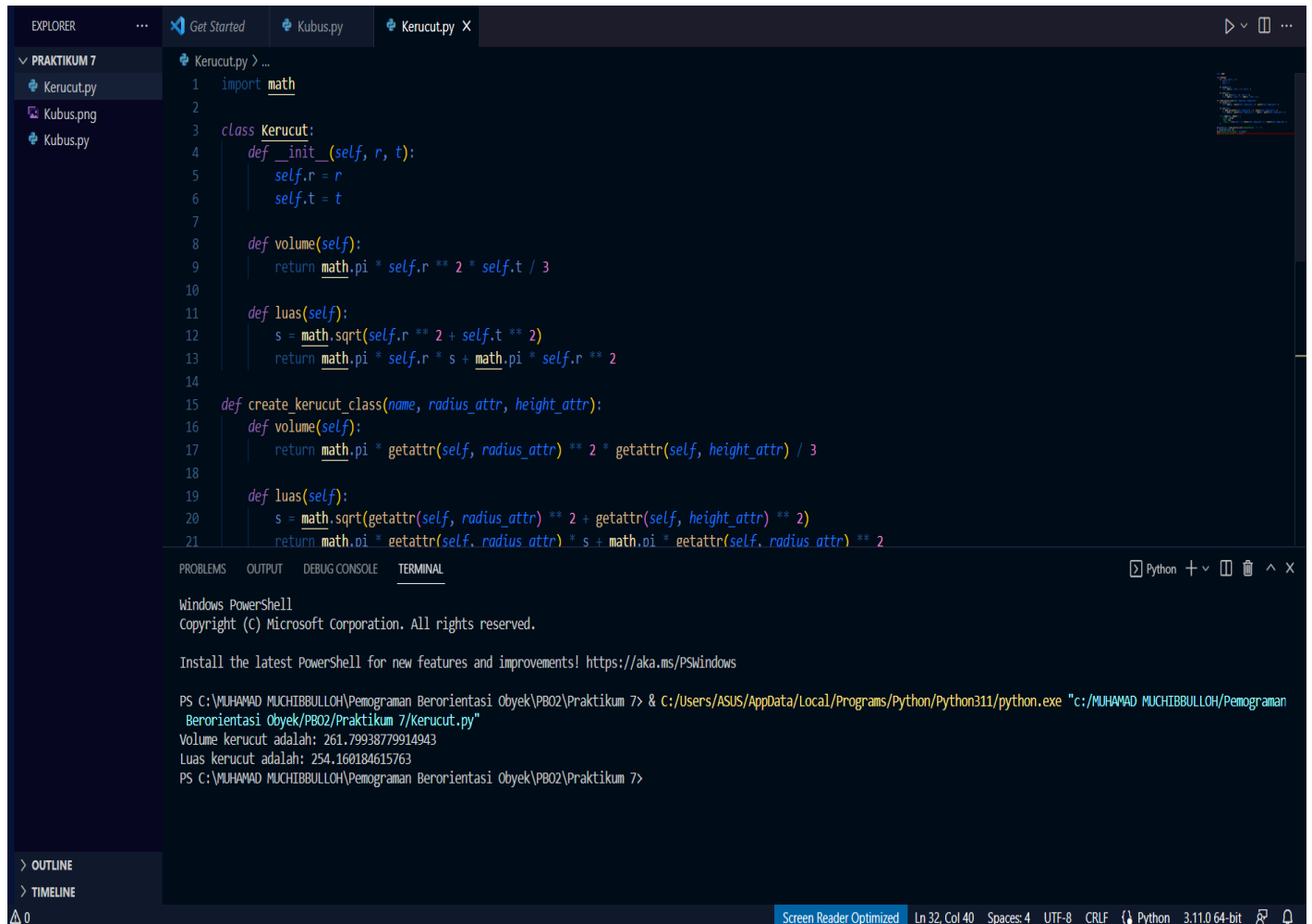
def create_kerucut_class(name, radius_attr, height_attr):
    def volume(self):
        return math.pi * getattr(self, radius_attr) ** 2 * getattr(self,
height_attr) / 3

    def luas(self):
        s = math.sqrt(getattr(self, radius_attr) ** 2 + getattr(self,
height_attr) ** 2)
        return math.pi * getattr(self, radius_attr) * s + math.pi *
getattr(self, radius_attr) ** 2

    return type(name, (object,), {
        'volume': volume,
        'luas': luas,
        '__init__': lambda self, r, t: setattr(self, radius_attr, r) or
setattr(self, height_attr, t)
    })

DynamicKerucut = create_kerucut_class('DynamicKerucut', 'r', 't')
k = DynamicKerucut(5, 10)
print("Volume kerucut adalah:", k.volume())
print("Luas kerucut adalah:", k.luas())
```

Output :



```
EXPLORER  ...  Get Started  Kubus.py  Kerucut.py X  ...  
PRAKTIKUM 7  
Kerucut.py  
Kubus.png  
Kubus.py  
Kerucut.py > ...  
1  import math  
2  
3  class Kerucut:  
4      def __init__(self, r, t):  
5          self.r = r  
6          self.t = t  
7  
8      def volume(self):  
9          return math.pi * self.r ** 2 * self.t / 3  
10  
11      def luas(self):  
12          s = math.sqrt(self.r ** 2 + self.t ** 2)  
13          return math.pi * self.r * s + math.pi * self.r ** 2  
14  
15  def create_kerucut_class(name, radius_attr, height_attr):  
16      def volume(self):  
17          return math.pi * getattr(self, radius_attr) ** 2 * getattr(self, height_attr) / 3  
18  
19      def luas(self):  
20          s = math.sqrt(getattr(self, radius_attr) ** 2 + getattr(self, height_attr) ** 2)  
21          return math.pi * getattr(self, radius_attr) * s + math.pi * getattr(self, radius_attr) ** 2  
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  
Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.  
  
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows  
  
PS C:\MUHAMAD MUCHIBULLOH\Pemograman Berorientasi Obyek\PB02\Praktikum 7> & C:/Users/ASUS/AppData/Local/Programs/Python/Python311/python.exe "c:/MUHAMAD MUCHIBULLOH/Pemograman Berorientasi Obyek/PB02/Praktikum 7/Kerucut.py"  
Volume kerucut adalah: 261.79938779914943  
Luas kerucut adalah: 254.160184615763  
PS C:\MUHAMAD MUCHIBULLOH\Pemograman Berorientasi Obyek\PB02\Praktikum 7>  
0  Screen Reader Optimized  Ln 32, Col 40  Spaces: 4  UTF-8  CRLF  Python  3.11.0 64-bit
```