Nama : Muhammad Baihaqi

Nim : 180511114

Kelas : K1

Pemrograman Berorientasi Objek Lanjut

Latihan 1

Class buku :

Script :

```
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("Harry Potter and the Philosopher's Stone", "J.K. Rowling")
bukuA.info()
```

Tampilan :

Class Celsius :

Script :

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

Tampilan :

Class kalkulator :

Script :

```
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
print(Kalkulator.add(3, 5))
print(Kalkulator.subtract(10, 7))
print(Kalkulator.multiply(4, 6))
print(Kalkulator.divide(12, 4))
```

Tampilan :

Class lingkaran :

Script :

```
class Lingkaran:
    def __init__(self, jari_jari):
        self.jari_jari = jari_jari
    def luas(self):
        return 3.14 * (self.jari_jari ** 2)
lingkaranA = Lingkaran(7)
print(f"Luas lingkaran: {lingkaranA.luas()}")
```

Tampilan :

Class mahasiswa :

Script :

```
class Mahasiswa:
    def __init__(self, nama, npm):
        self.nama = nama
        self.npm = npm
    def info(self):
        print(f"Nama: {self.nama}\nNPM: {self.npm}")
mahasiswaB = Mahasiswa("Ahmad", "123456789")
mahasiswaB.info()
```

Tampilan :

Class mobil :

Script :

```
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("Toyota", "Hitam")
mobilA.info() # Output: Mobil Toyota berwarna Hitam
```

Tampilan :

Class pesawat terbang :

Script :

```
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("Garuda Indonesia", "Jakarta - Bali")
pesawatA.info()
```

Tampilan :