

Nama : Ditha maharani

NIM : 191100095

1. (Implementasi OOP Dengan Atribut Class)

Koding

```
class VariabelClass:
    #-- Variabel Class
    nama_dosen=''
    VariabelClass =tunjangan=0
    gaber=0
    keterangan=''
    def isi_variabel_class(self):
        #-- Inisial variabel class
        VariabelClass.nama_dosen ='Dilson, M.Kom'
        VariabelClass.VariabelClass =1800000
        VariabelClass.tunjangan =700000
        #print('Nama Dosen : ',VariabelClass.nama_dosen)
    def proses_variabel_class(self):
        #-- Proses Aritmatika Variabel Class
        VariabelClass.gaber =VariabelClass.VariabelClass
+ VariabelClass.tunjangan
    def logika_variabel_class(self):
        if (VariabelClass.VariabelClass < 2000000):
            VariabelClass.keterangan='Gaji Pokok Anda di
Bawah UMR'
        else:
            VariabelClass.keterangan='Gaji Pokok Anda Lum
ayan'
    def cetak_variabel_class(self):
        print('Nama Dosen : ',VariabelClass.nama_dosen)
        print('Gaji Pokok Dosen : ', VariabelClass.Variab
elClass)
        print('Keterangan : ', VariabelClass.keterangan
)
        print('Tunjangan Dosen : ', VariabelClass.tunjang
an)
        print('-----')
        print('Gaji Berih Dosen Rp. ', VariabelClass.gabe
r)
        print('-----')

    def kesimpulannya(self):
        if(VariabelClass.gapok < 2000000):
            print('Kesimpulannya'' Gaji pokok yang diberi r
endah dibawah UMR')
```

```

if __name__ == '__main__':
    #--ciptakan objek
    obj=Gapok()
    obj.isi_variabel_class()
    obj.proses_variabel_class()
    obj.logika_variabel_class()
    obj.cetak_variabel_class()
    obj.kesimpulannya()
    #-- cara mengakses variabel class dalam main
    nama =Gapok.nama_dosen
    print('Nama Dosen : ', nama)

```

Output

```

Nama Dosen :  Dilson, M.Kom
Gaji Pokok Dosen :  1800000
Keterangan    :  Gaji Pokok Anda di Bawah UMR
Tunjangan Dosen :  700000
-----
Gaji Berih Dosen Rp.  2500000
-----
Kesimpulannya Gaji pokok yang diberi rendah dibawah UMR
Nama Dosen :  Dilson, M.Kom

```

2. Class Method Dengan Variabel Instan

```

class Pegawai:

    Total = 0

    def __init__(self, nama_pegawai, jabatan_pegawai, gaji_p
egawai) :
        self.nama_pegawai = nama_pegawai
        self.jabatan_pegawai = jabatan_pegawai
        self.gaji_pegawai = gaji_pegawai
        Pegawai.Total +=1

# Ciptakan objek pegawai dan tambahkan data secara instan pa
da class Pegawai
objpegawai1 = Pegawai ("Ditha", "supervisor", "5000.00")
objpegawai2 = Pegawai ("Della", "encoder", "3500.00")
objpegawai3 = Pegawai ("Ranti", "proofreader", "4000.00")
# Cetak Objek Pegawai
print(objpegawai1.nama_pegawai, objpegawai1.jabatan_pegawai,
objpegawai1.gaji_pegawai)
print(objpegawai2.nama_pegawai, objpegawai2.jabatan_pegawai,
objpegawai2.gaji_pegawai)
print(objpegawai3.nama_pegawai, objpegawai3.jabatan_pegawai,
objpegawai3.gaji_pegawai)
# cetak total pegawai menggunakan atribute class
print(Pegawai.Total)

```

Output:

```

Ditha supervisor 5000.00

```

Della encoder 3500.00
Ranti proofreader 4000.00
Total Pegawai 3

3. Class Dengan Atribut Method

Koding

```
class VariabelObjek:
    def deklarasi_var_metod(self):
        self.kode=''
        self.judul=''
        self.pengarang=''
        self.penerbit=''
        self.harga=0
    def isi_variabel_metod(self):
        #-- Isi biasa
        self.kode='BK-001'
        #-- isi dengan input
        self.judul=input('Judul Buku : ')
        self.pengarang=input('Pengarang Buku : ')
        self.penerbit=input('Penerbit Buku : ')
        self.harga=input('Harga Buku : ')
    def proses_variabel_metod(self):
        #-- Proses Aritmatika Variabel Class
        print(self.pengarang, ', ', self.penerbit)
    def logika_variabel_metod(self):
        if (int(self.harga) > 80000):
            self.keterangan='Mahal'
        else:
            self.keterangan='Murah'
    def cetak_variabel_metod(self):
        print('Kode Nuku : ', self.kode)
        print('Judul Buku : ', self.judul)
        print('Pengarang : ', self.pengarang)
        print('Penerbit : ', self.penerbit)
        print('Harga : ', self.harga)
        print('-----')
        print('Keterangan : ', self.keterangan)
        print('-----')
if __name__=='__main__':
    #--ciptakan objek
    obj=VariabelObjek()
    obj.isi_variabel_metod()
    obj.proses_variabel_metod()
    obj.logika_variabel_metod()
    #----os.system('cls')
    obj.cetak_variabel_metod()
```

Output:

```

Judul Buku : Motivasi
Pengarang Buku : Ditha maharani
Penerbit Buku : Erlangga
Harga Buku : 100000
Nurmayanti , Erlangga
Kode Nuku : BK-001
Judul Buku : Motivasi
Pengarang : Ditha maharani
Penerbit : Erlangga
Harga : 100000
-----
Keterangan : Mahal
-----

```

4. Class Dengan Parameter Methode**Koding:**

```

import os
class ParameterObjek:
    def deklarasi_var_metod(self, kode, judul, pengarang, pe
nerbit, harga,banyak):
        self.kode=kode
        self.juduljudul
        self.pengarang=pengarang
        self.penerbit =penerbit
        self.harga=harga
        self.banyak=banyak
    def isi_variabel_metod(self):
        #-- Isi biasa
        self.kode ='BK-001'
        #-- isi dengan input
        self.judul = input('Judul Buku : ')
        self.pengarang = input('Pengarang Buku : ')
        self.penerbit = input('Penerbit Buku : ')
        self.harga = float(input('Harga Buku : '))
        self.banyak = int(input('Banyak Jual : '))
    def proses_variabel_metod(self):
        #-- Proses Aritmatika Variabel Class
        print(self.pengarang , ', ' , self.penerbit)
    def logika_variabel_metod(self):
        if (int(self.harga) > 80000):
            self.keterangan='Mahal'
        else:
            self.keterangan='Murah'
    def cetak_variabel_metod(self):
        print('Kode Nuku : ', self.kode)
        print('Judul Buku : ', self.judul)
        print('Pengarang : ', self.pengarang)
        print('Penerbit : ', self.penerbit)
        print('Harga {:,.2f} Rp. '.format(self.harga))

```

```

        print('Banyak {:,.2f} '.format(self.banyak))
        #--
Cara proses dan cetak langsung dengan variabel biasa
        jmljual = self.banyak * self.harga
        print('Penjualan {:,.2f} '.format(jmljual))

        print('-----')
        print('Keterangan : ', self.keterangan)
        print('-----')

#-- Main Program ---
if __name__=='__main__':
    #--ciptakan objek
    obj=ParameterObjek()
    #-- panggil metode objek
    obj.isi_variabel_metod()
    obj.proses_variabel_metod()
    obj.logika_variabel_metod()
    #----
    os.system('cls')
    obj.cetak_variabel_metod()
    #-- Deklarasi Var biasa di dalam Main program
    jumlah = obj.banyak * obj.harga
    #-- cara cetak langsung di Main Program
    print('Jumlah Penjualan Rp. {:,.2f}'.format(jumlah))

```

Output:

```

Judul Buku : Muda Kreatif
Pengarang Buku : Ditha maharani
Penerbit Buku : Erlangga
Harga Buku : 100000
Banyak Jual : 100
Nurmayanti , Erlangga
Kode Nuku : BK-001
Judul Buku : Pintu Hati
Pengarang : Ditha maharani
Penerbit : Erlangga
Harga 100,000.00 Rp.
Banyak 100.00
Penjualan 10,000,000.00
-----
Keterangan : Mahal
-----
Jumlah Penjualan Rp. 10,000,000.00

```

5. Class Dengan Magic Methode

Koding:

```

class DenganMagicMetod:
    def __init__(self, a,b):
        # -- INstance Variabel

```

```

        self.a = a
        self.b = b

# Magic Method __add__()
def __add__(self, nilai):
    a = self.a + nilai.a
    b = self.b + nilai.b
    tambah = DenganMagicMetod(a, b)
    return tambah

# Magic Method __sub__()
def __sub__(self, nilai):
    a = self.a - nilai.a
    b = self.b - nilai.b
    kurang = DenganMagicMetod(a, b)
    return kurang

# Magic Method __mul__()
def __mul__(self, nilai):
    a = self.a * nilai.a
    b = self.b * nilai.b
    kali = DenganMagicMetod(a, b)
    return kali

# Magic Method __div__()
def __div__(self, nilai):
    a = self.a / self.b
    b = nilai.a / nilai.b
    bagi = DenganMagicMetod(a, b)
    return bagi

# --
untuk membandingkan nilai objek greater than (__gt__)
def __gt__(self, nilai):
    bil1 = self.a + self.b
    bil2 = nilai.a + nilai.b
    #-- bandingkan nilai objek1 dengan objek 2
    if bil1 > bil2:
        return True # objek yang terbesar ditampilkan
    else:
        return False

# -- operator metode > : __lt__ (self, nilai) => less than
# -- untuk membandingkan nilai objek
def __lt__(self, nilai):
    bil1 = self.a + self.b
    bil2 = nilai.a + nilai.b
    #-- bandingkan nilai objek1 dengan objek 2
    if bil1 < bil2:
        return True # objek yang terbesar ditampilkan
    else:

```

```

        return False
# magic method __str__() # untuk format cetak
def __str__(self):
    return '{} , {}'.format(self.a, self.b)

#-- Main Program
if __name__ == '__main__':
    # -- create objek isikan nilai parameteranya
    # -- artinya = variabel static
    obj1 = DenganMagicMetod(10,20)
    obj2 = DenganMagicMetod(4,6)
    # --

    # -- panggil metod objek
    print('-----')
    print('PENJUMLAHAN')
    print('-----')
    #-- cetak1
    tambah = obj1 + obj2 # use __add__()
    print(tambah) # use __str__()

    #--
    tambah = obj1 + obj1
    print(tambah) # use __str__()
    tambah = obj2 + obj2
    print(tambah) # use __str__()
    #--
    kurang = obj1 - obj2
    print(kurang) # use __str__()
    #-- perkalian
    kali = obj1 * obj2
    print(kali) # use __str__()
    # # ==
    menentukan nilai objek terbesar __gt__ = greater than
    # # ---
    menggunakan operator methode (magic method __gt__(self, nila
i))
    if obj1 > obj2 :
        print('Objek dengan nilai terbesar ==> ', obj1)
    else:
        print('Objek dengan nilai terbesar ==> ', obj2)
    # # ==
    menentukan nilai objek terkecil __lt__ = less than
    # # ---
    menggunakan operator methode (magic method __lt__(self, nila
i))
    if obj1 < obj2 :
        print('Objek dengan nilai terkecil ==> ', obj1)

```

```

else:
    print('Objek dengan nilai terkecil ==> ', obj2)
# =====
# Overriding nilai variabel pada objek
bil3 = 43
print('Hasil timpa nilai Class = ', bil3.__str__())
#--- overriding cara 2
angka1 = 29
angka2 = 49
print(angka1.__str__(), angka2.__str__())
print(tambah.__str__())

```

Output:

```

-----
PENJUMLAHAN
-----
<__main__.DenganMagicMetod object at 0x7fa024e7a7d0>
<__main__.DenganMagicMetod object at 0x7fa024de00d0>
<__main__.DenganMagicMetod object at 0x7fa024e7a7d0>
<__main__.DenganMagicMetod object at 0x7fa024de00d0>
<__main__.DenganMagicMetod object at 0x7fa024de0110>
Objek dengan nilai terbesar ==> <__main__.DenganMagicMetod object
at 0x7fa02aa79650>
Objek dengan nilai terkecil ==> <__main__.DenganMagicMetod object
at 0x7fa02aa79bd0>
Hasil timpa nilai Class = 43
29 49
<__main__.DenganMagicMetod object at 0x7fa024e7a7d0>

```