



UNIVERSITAS
GADJAH MADA

W11 – Pemrograman Lanjut

Oleh :

Alim Satria Fi'i Wijaya Kusuma (21/483503/SV/20304)

Modul OOP-1



UNIVERSITAS GADJAH MADA

CLASS

+ Code

+ Markdown

```
class Parrot :  
    species = "bird"  
  
    ##### ATTRIBUTE #####  
    def __init__(self, name, age) :  
        self.name = name  
        self.age = age  
  
papi = Parrot("papi", 3)  
greeny = Parrot("greeny", 5)  
  
print("papi is a " + papi.species)  
print("greeny is also a " + greeny.species)  
  
print(papi.name + " is " + str(papi.age) + " years old")  
print(greeny.name + " is " + str(greeny.age) + " years old")
```

✓ 0.6s

```
papi is a bird  
greeny is also a bird  
papi is 3 years old  
greeny is 5 years old
```

Dalam pemrograman berbasis OOP (Object Oriented Programming), **class** merupakan tempat untuk menampung attribute dan method atau dinamakan dengan object.

attribute merupakan sifat-sifat khas yang hanya dimiliki oleh object tersebut. dalam contoh pemrograman disamping, attribute yang digunakan adalah name dan age

METHOD

```
class Parrot :  
    species = "bird"  
  
    ##### ATTRIBUTE #####  
    def __init__(self,name,age) :  
        self.name = name  
        self.age = age  
  
    def sing(self,song) :  
        return "{} sings {}".format(self.name,song)  
  
    def dance(self) :  
        return "{} is now dancing".format(self.name)  
  
papi = Parrot("papi",10)  
  
print(papi.sing("Happy"))  
print(papi.dance())
```

✓ 0.5s

```
papi sings Happy  
papi is now dancing
```

Method merupakan fungsi-fungsi yang hanya dapat dilakukan oleh suatu object. dari contoh pemrograman disamping method yang digunakan adalah sing dan dance.

Modul OOP-2



UNIVERSITAS GADJAH MADA

str

```
class class_str :
    def __init__(self,value = None) :
        self.var = value
    def __str__(self) :
        return self.var
```

```
objek_1 = class_str("objek")
objek_2 = class_str(5)
```

```
print("ini output : " + objek_1.var)
print("ini output : " + objek_2.var)
```

⊗ 0.1s

ini output : objek

```
-----
TypeError                                Traceback (most recent call last)
/home/alimsatria/Documents/MATERI KULIAH/PEMROGRAMAN LANJUT/WEK_11/W11_MODUL.ipynb Cell 10' in <module>
      8 objek_2 = class_str(5)
     10 print("ini output : " + objek_1.var)
----> 11 print("ini output : " + objek_2.var)

TypeError: can only concatenate str (not "int") to str
```

__str__

merupakan salah satu properties method yang berfungsi untuk mengembalikan nilai variable string. selain string nilai variable tidak dikembalikan (tidak menjadi objek).

Modul OOP-2



UNIVERSITAS GADJAH MADA

__dict__

digunakan untuk menampilkan isi dari sebuah class / objek . Apabila perintah ini digunakan untuk membuka isi dari objek. maka isinya berupa fungsi-fungsi yang tidak bersifat hidden. apabila perintah digunakan untuk membuka isi class maka isi akan tertampil semua

```
dict

class classy :
    varia = 1

    def __init__(self) :
        self.var = 2

    def method(self) :
        pass

    def __hidden(self) :
        pass

obj = classy()

print(obj.__dict__)
print(classy.__dict__)

✓ 0.3s

{'var': 2}
{'__module__': '__main__', 'varia': 1, '__init__': <function classy.__init__ at 0x7fad423d4dc0>, 'method': <function classy.method at 0x7fad423d45e0>, '__class__': <function classy.__hidden at 0x7fad423d4160>, '__dict__': <attribute '__dict__' of 'classy' objects>, '__weakref__': <attribute '__weakref__' of 'classy' objects>, '__doc__': None}
```

bases

```
class super_one :  
    pass  
class super_two :  
    pass  
class sub(super_one,super_two) :  
    pass  
  
def printBases(cls) :  
    print('(',end='')  
  
    for x in cls.__bases__ :  
        print(x.__name__,end=' ')  
  
    print(')')  
  
printBases(super_one)  
printBases(super_two)  
printBases(sub)
```

✓ 0.6s

```
(object )  
(object )  
(super_one super_two )
```

__bases__

Perintah bases digunakan untuk menampilkan basis object yang digunakan pada sebuah class.

pada contoh program disamping ditampilkan basis dari class sub adalah super_one dan super_two.

Tugas Modul OOP-1



UNIVERSITAS GADJAH MADA

TUGAS

```
class kardus :  
  
    def __init__(self,panjang,lebar,tinggi) :  
        self.panjang = panjang  
        self.lebar    = lebar  
        self.tinggi   = tinggi  
  
    def volume_kardus(self) :  
        return self.panjang*self.lebar*self.tinggi  
  
    def luas_permukaan_kardus(self) :  
        return 2*(self.panjang*self.lebar + self.panjang*self.tinggi + self.tinggi*self.lebar)  
  
    def massa_jenis(self,masa) :  
        return masa / self.volume_kardus()
```

```
kotak_biru = kardus(10,8,4)  
print("KOTAK BIRU : ")  
print("volume : " + str(kotak_biru.volume_kardus()))  
print("luas permukaan kardus : " + str(kotak_biru.luas_permukaan_kardus()))  
print("massa jenis : " + str(kotak_biru.massa_jenis(5)))  
  
print()  
  
kotak_merah = kardus(15,5,1)  
print("KOTAK MERAH : ")  
print("volume : " + str(kotak_merah.volume_kardus()))  
print("luas permukaan kardus : " + str(kotak_merah.luas_permukaan_kardus()))  
print("massa jenis : " + str(kotak_merah.massa_jenis(5)))
```

```
KOTAK BIRU :  
volume : 320  
luas permukaan kardus : 304  
massa jenis : 0.015625
```

```
KOTAK MERAH :  
volume : 75  
luas permukaan kardus : 190  
massa jenis : 0.06666666666666667
```