

MANAGEMENT KARYAWAN

Nama : Muhammad Nabil Abdillah

NIM : 202412053

1. Source Code

```
❶ parent class.py >-
❷     class Karyawan:
❸         def __init__(self, nama, id_karyawan, gaji_pokok):
❹             self.nama = nama
❺             self.id_karyawan = id_karyawan
❻             self.gaji_pokok = gaji_pokok
❼
➋         def hitung_gaji(self):
⌃             return self.gaji_pokok
⌄
⌂         def info(self):
⌃             return f"Karyawan : {self.nama}, ID: {self.id_karyawan}, Gaji: {self.hitung_gaji()}"
⌄
⌂
⌂     class Manager(Karyawan):
⌃         def __init__(self, nama, id_karyawan, gaji_pokok, tunjangan):
⌄             super().__init__(nama, id_karyawan, gaji_pokok)
⌅             self.tunjangan = tunjangan
⌆
⌂         def hitung_gaji(self):
⌃             return self.gaji_pokok + self.tunjangan
⌄
⌂         def info(self):
⌃             return f"Manager : {self.nama}, ID: {self.id_karyawan}, Gaji: {self.hitung_gaji()}"
⌄
⌂
⌂     class Programmer(Karyawan):
⌃         def __init__(self, nama, id_karyawan, gaji_pokok, bonus):
⌄             super().__init__(nama, id_karyawan, gaji_pokok)
⌅             self.bonus = bonus
⌆
⌂         def hitung_gaji(self):
⌃             return self.gaji_pokok + self.bonus
⌄
⌂         def info(self):
⌃             return f"Programmer : {self.nama}, ID: {self.id_karyawan}, Gaji: {self.hitung_gaji()}"
⌄
⌂
⌂ # --- Program utama ---
⌂ m = Manager("Muhammad Nabil Abdillah", "202412053", 17000000, 3000000)
⌂ p = Programmer("Nugroho", "P001", 15000000, 1000000)
⌂
⌂ print(m.info())
⌂ print(p.info())
⌄
```

2. Output

```
PS C:\Users\Acer-PC\OneDrive\Documents\VA - KULIAH\SEMESTER 3\PEPROGRAMAN BERORIENTASI OBIEK\PYTHON
SHT 3> & C:/Users/Acer-PC/AppData/Local/Programs/Python/Python313/python.exe "C:/Users/Acer-PC/OneDrive/Documents/VA - KULIAH\SEMESTER 3\PEPROGRAMAN BERORIENTASI OBIEK\PYTHON\SHT 3\parent\class.py"
Manager : Muhammad Nabil Abdillah, ID: 202412053, Gaji: 20000000
Programmer : Nugroho, ID: P001, Gaji: 16000000
PS C:\Users\Acer-PC\OneDrive\Documents\VA - KULIAH\SEMESTER 3\PEPROGRAMAN BERORIENTASI OBIEK\PYTHON
SHT 3>
```

3. Penjelasan

Class Karyawan berfungsi sebagai dasar untuk merepresentasikan data karyawan dalam sistem. Method `__init__()` menginisialisasi atribut nama, `id_karyawan`, dan `gaji_pokok`, sedangkan `hitung_gaji()` mengembalikan gaji pokok dan `info()` menampilkan informasi karyawan. Class ini menjadi dasar penerapan inheritance dan polymorphism bagi class turunan seperti Manager dan Programmer.

Class Manager merupakan turunan dari Karyawan yang menambahkan atribut tunjangan dan menggunakan `super()` untuk memanggil constructor induk. Method `hitung_gaji()` di-override untuk menambah tunjangan, dan `info()` menampilkan informasi manajer beserta total gajinya, sebagai contoh penerapan polymorphism.

Class Programmer merupakan turunan dari Karyawan dengan tambahan atribut **bonus**. Menggunakan `super()` untuk inisialisasi atribut dasar, lalu `hitung_gaji()` di-override untuk menambah bonus dan `info()` menampilkan informasi lengkap beserta total gaji, menunjukkan penyesuaian perilaku dari class induk.