

**LAPORAN PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK (PBO)**

**PRAKTIKUM 3**



2411102441052

Angga Maulana Saputra

**FAKULTAS SAINS DAN TEKNOLOGI**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**UNIVERSITAS MUHAMMADIYAH KALIMANTAN TIMUR**

**EMAIL : 2411102441052@umkt.ac.id**

## Latihan Praktikum 3:

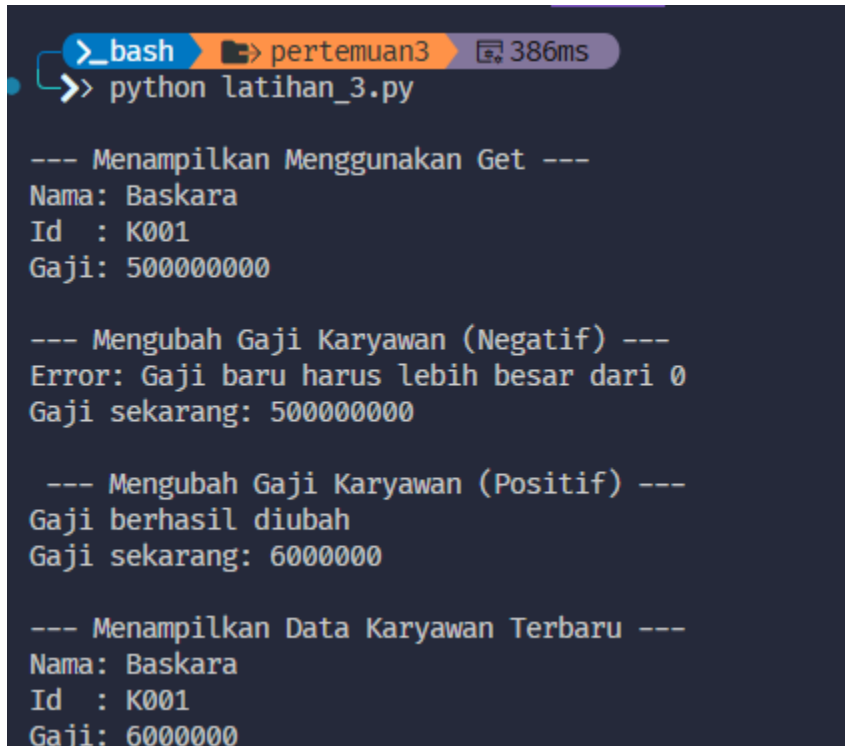
- a. Screenshoot kode latihan\_3.py.

```

1 class Karyawan:
2     def __init__(self, nama, id_karyawan, gaji):
3         self.__nama = nama
4         self.__id_karyawan = id_karyawan
5         self.__gaji = gaji
6
7     def get_nama(self):
8         return self.__nama
9
10    def get_id(self):
11        return self.__id_karyawan
12
13    def get_gaji(self):
14        return self.__gaji
15
16    def set_nama(self, nama_baru):
17        if nama_baru == "":
18            print("Nama baru tidak boleh kosong")
19        else:
20            self.__nama = nama_baru
21            print("Nama berhasil diubah")
22
23    def set_gaji(self, gaji_baru):
24        if gaji_baru <= 0:
25            print("Error: Gaji baru harus lebih besar dari 0")
26        else:
27            self.__gaji = gaji_baru
28            print("Gaji berhasil diubah")
29
30    karyawan1 = Karyawan("Baskara", "K001", 500000000)
31
32    print("\n--- Menampilkan Menggunakan Get ---")
33    print(f>Nama: {karyawan1.get_nama()}")
34    print(f"Id : {karyawan1.get_id()}")
35    print(f"Gaji: {karyawan1.get_gaji()}")
36
37    print("\n--- Mengubah Gaji Karyawan (Negatif) ---")
38    karyawan1.set_gaji(-5000000)
39    print(f"Gaji sekarang: {karyawan1.get_gaji()}")
40
41    print("\n --- Mengubah Gaji Karyawan (Positif) ---")
42    karyawan1.set_gaji(6000000)
43    print(f"Gaji sekarang: {karyawan1.get_gaji()}")
44
45    print("\n--- Menampilkan Data Karyawan Terbaru ---")
46    print(f>Nama: {karyawan1.get_nama()}")
47    print(f"Id : {karyawan1.get_id()}")
48    print(f"Gaji: {karyawan1.get_gaji()}")

```

- b. Screenshoot hasil eksekusi program menunjukkan proses validasi.



```

>_bash  ➡ pertemuan3  386ms
>> python latihan_3.py

--- Menampilkan Menggunakan Get ---
Nama: Baskara
Id : K001
Gaji: 500000000

--- Mengubah Gaji Karyawan (Negatif) ---
Error: Gaji baru harus lebih besar dari 0
Gaji sekarang: 500000000

--- Mengubah Gaji Karyawan (Positif) ---
Gaji berhasil diubah
Gaji sekarang: 60000000

--- Menampilkan Data Karyawan Terbaru ---
Nama: Baskara
Id : K001
Gaji: 60000000

```

Hasil ini menunjukkan:

- Validasi gagal: saat gaji negatif atau nama kosong, program menolak dan data lama tidak berubah.
  - Validasi berhasil: saat gaji valid, program menerima dan memperbarui data.
- c. Refleksi singkat: Enkapsulasi disebut sebagai salah satu pilar fundamental OOP karena konsep ini melindungi data dengan membatasi akses langsung ke atribut dan hanya memperbolehkan perubahan melalui method yang sudah divalidasi. Dengan demikian, integritas data lebih terjaga, kode menjadi lebih aman, serta lebih mudah dipelihara dan dikembangkan.