

****🔥 Tantangan Python: Global vs Local Scope****

Berdasarkan catatanmu, berikut 5 tantangan untuk menguji pemahamanmu tentang variabel global dan lokal:

🎯 1. **Akses Variabel Global**

Buat program dengan:

- Variabel global `nama = "Budi"`.
- Fungsi `sapa()` yang mencetak `Halo, {nama}!` (tanpa parameter).
- Panggil fungsi dan cetak variabel `nama` di luar fungsi.

****Contoh Output:****

```

Halo, Budi!

Budi

```

🎯 2. **Modifikasi Variabel Global**

Diberikan:

```python

counter = 0 # Global

```

1. Buat fungsi `tambah_counter()` yang menambah `counter` dengan 1 (gunakan `global`).

2. Panggil fungsi 3x, lalu cetak `counter`.

****Contoh Output:****

```

3

```

🎯 3. ****Variabel Lokal vs Global****

Buat program dengan:

- Variabel global `x = 10`.
- Fungsi `ubah_x()` yang membuat variabel lokal `x = 20` (tanpa `global`).
- Cetak `x` sebelum, di dalam, dan setelah panggil fungsi.

****Contoh Output:****

```

Sebelum: 10

Di dalam fungsi: 20

Setelah: 10

```

🎯 4. ****Scope Bersarang****

Buat fungsi `luar()` yang:

1. Memiliki variabel lokal `y = 5`.
2. Berisi fungsi `dalam()` yang mengakses `y` dan mencetaknya.
3. Panggil `dalam()` dari `luar()`.

****Contoh Output:****

\\

5

\\

🎯 5. ****Error Handling Scope****

Buat program dengan:

- Fungsi `coba()` yang mencoba mencetak variabel `z` (tanpa mendefinisikannya).
- Di luar fungsi, definisikan `z = 100`.
- Jelaskan mengapa terjadi error/sukses!

****Contoh Output:****

\\

Error: name 'z' is not defined

Karena z tidak ada di scope lokal/global saat fungsi dipanggil!

\\

💡 ****Tips Penyelesaian:****


1. Gunakan `global` untuk modifikasi variabel global di dalam fungsi.
2. Variabel lokal hanya hidup di dalam fungsi.
3. Variabel di fungsi luar bisa diakses oleh fungsi dalam (nested).

****Contoh Solusi Tantangan 2:****

```
```python
counter = 0

def tambah_counter():
 global counter
 counter += 1

tambah_counter()
tambah_counter()
tambah_counter()
print(counter) # Output: 3
```
```

****Selamat Mengerjakan!**** 

****Pahami scope variabel, maka bug akan mudah ditaklukkan!****