**Rangkuman Aksi Sekuensial**

Kita sudah berada di penghujung materi aksi sekuensial. Sampai sejauh ini, Anda telah memiliki pemahaman mengenai aksi sekuensial Python yang menjadi ilmu mendasar dalam pemrograman, khususnya Python. Mari kita rangkum secara saksama.

**Pengenalan Aksi Sekuensial**

Aksi sekuensial adalah sederetan instruksi yang akan dijalankan oleh komputer berdasarkan urutan penulisannya. Dalam Python, kode yang Anda bangun akan berjalan sesuai dengan urutan perintahnya.

Perlu diperhatikan bahwa ada program yang akan berubah hasilnya jika urutan baris instruksinya diubah. Ada pula program yang hasilnya **tidak** akan berubah jika urutan baris instruksinya diubah.

**Python Interpreter**

Dalam Python, kode program yang Anda bangun akan ditransformasi menjadi kode yang mudah dimengerti oleh mesin menggunakan program compiler atau interpreter. Compiler merupakan program yang akan menerjemahkan bahasa pemrograman menjadi bahasa mesin sebelum dijalankan. Ini artinya program yang Anda bangun secara keseluruhan akan diubah terlebih dahulu semuanya menjadi bahasa mesin.

Hal berbeda terjadi pada interpreter, yang akan menerjemahkan bahasa Python menjadi bahasa mesin satu per satu secara langsung. Hal ini memungkinkan Anda untuk melihat hasil program segera setelah satu baris kode dieksekusi hingga selesai. Implementasi interpreter ada pada mode interaktif Python.

**Block Code**

Sebuah program pada Python dibangun berdasarkan blok-blok kode. Sebuah blok merujuk pada potongan kode program Python yang dijalankan sebagai satu unit. Kode blok dapat berupa modul, fungsi, dan kelas. Contoh kode Python yang memiliki bentuk blok adalah perulangan for.

1. for i in range(10):
2. print(i)

**Case-sensitive**

Python termasuk bahasa pemrograman *case-sensitive*. Ini artinya Python memperlakukan huruf besar dan kecil sebagai karakter yang berbeda dalam penamaan variabel, nama fungsi, atau penulisan kode secara umum. Berikut contoh penerapannya.

1. teks = "Dicoding"
2. Teks = "Indonesia"
4. print(teks)
5. print(Teks)
7. """
8. Output:
9. Dicoding
10. Indonesia
11. """

Pada program di atas, Anda membuat dua variabel dengan nama "teks" dan "Teks". Python akan menganggap bahwa variabel tersebut berbeda, walaupun bagi kita sebagai manusia, kedua hal tersebut memiliki arti yang sama.

**One-liner**

Selain membangun kode berdasarkan bloknya. Anda juga dapat membuat sebuah kode hanya dalam satu baris saja. Konsep ini dikenal sebagai *one-liner*.

*One-liner* merupakan gaya penulisan pada Python yang memungkinkan Anda untuk membuat sebuah kode hanya dalam satu baris. *One-liner* adalah salah satu keunggulan dalam Python yang susah untuk diimplementasikan dalam beberapa bahasa pemrograman yang lainnya.

Tujuan dari *one-liner* ini adalah membuat satu baris kode yang singkat dan jelas. Perlu diingat bahwa tidak semua kode blok dapat dijadikan *one-liner*, seperti deklarasi fungsi, modul, dan kelas.

Ada banyak kode program pada Python yang memiliki versi *one-liner* masing-masing. Salah satu contohnya adalah program penukaran dua variabel.

1. x = 1
2. y = 2
4. x, y = y, x