

# **Tipe Data, Variabel, dan Operator di Bahasa Pemrograman Python**

**Hartono, S.Pd., M.T.I**

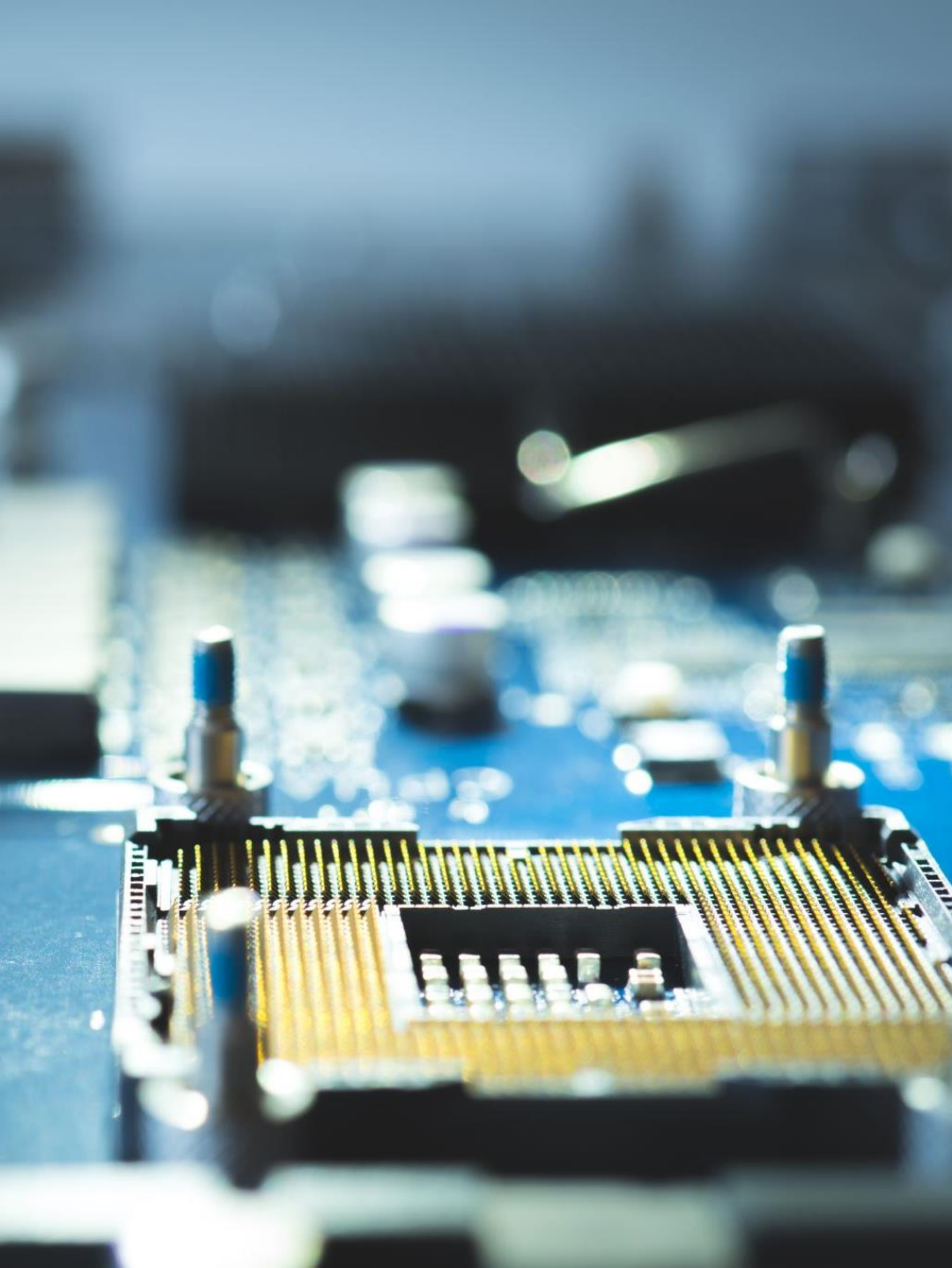
**Universitas Muhammadiyah Kotabumi**



---

## Tipe Data

- Tipe data merupakan fondasi utama yang memungkinkan kita untuk memahami, mengelola, dan memanipulasi berbagai jenis informasi yang diolah oleh program komputer.
- Setiap jenis data memiliki sifat dan aturan tertentu yang menginformasikan komputer tentang bagaimana data tersebut harus diperlakukan.
- Ini membantu komputer untuk memahami cara memori dialokasikan, bagaimana operasi matematika diterapkan, dan bagaimana data ditampilkan.

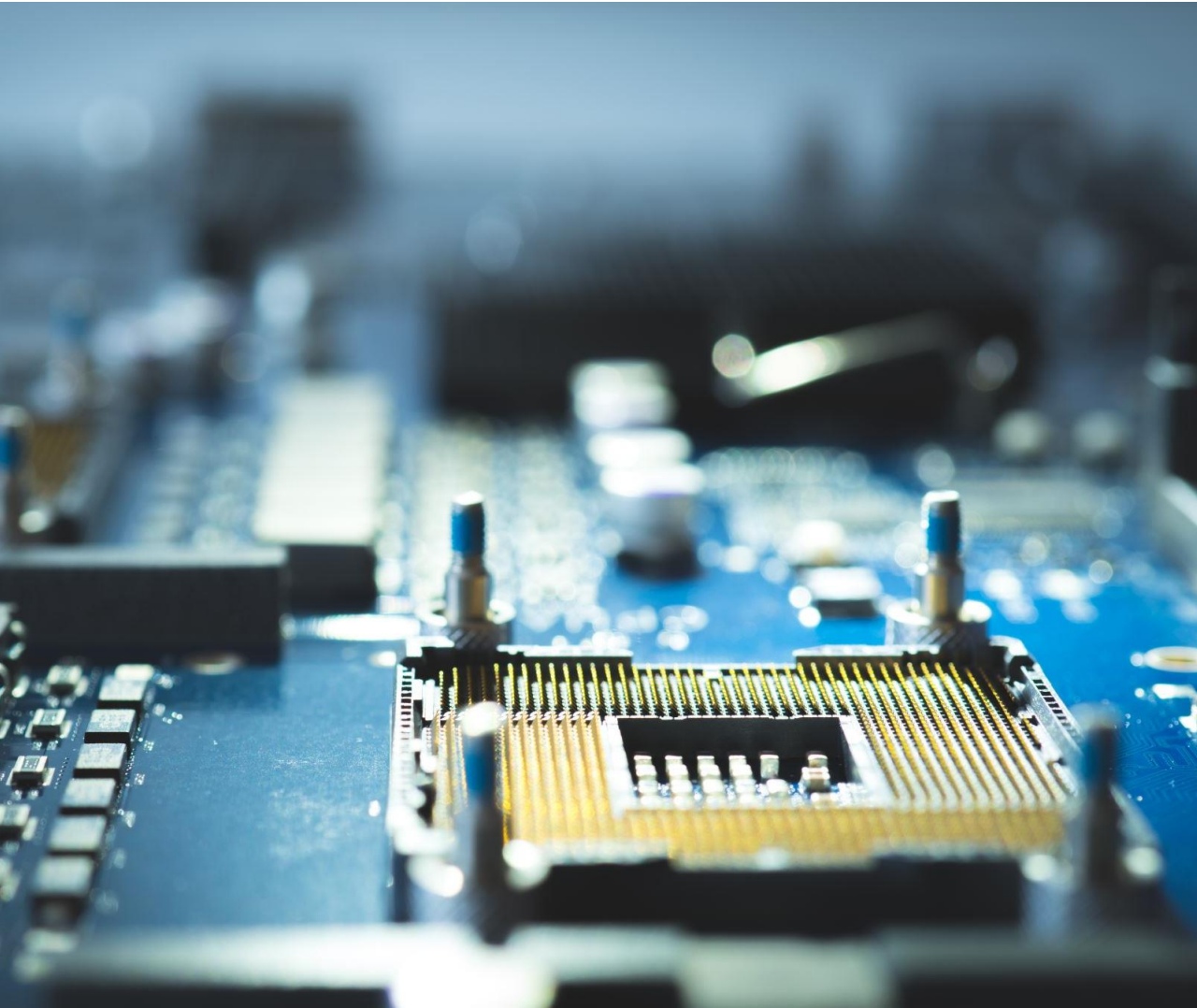


---

## ... Tipe Data

- Tipe data membantu mewakili beragam informasi, dari angka hingga teks, status kebenaran, tanggal, waktu, hingga struktur data yang kompleks seperti list dan dict.
- Dalam pemrograman, memahami tipe data penting karena ini mempengaruhi bagaimana data tersebut diperlakukan dalam memori, bagaimana operasi matematika atau manipulasi dapat dilakukan, dan bagaimana data tersebut ditampilkan kepada pengguna.
- Tipe data juga mempengaruhi bagaimana kita berinteraksi dengan input dan output dalam program.
- Dalam bahasa pemrograman seperti Python, setiap tipe data memiliki fitur dan metode yang spesifik yang dapat digunakan untuk manipulasi.

# ... Tipe Data



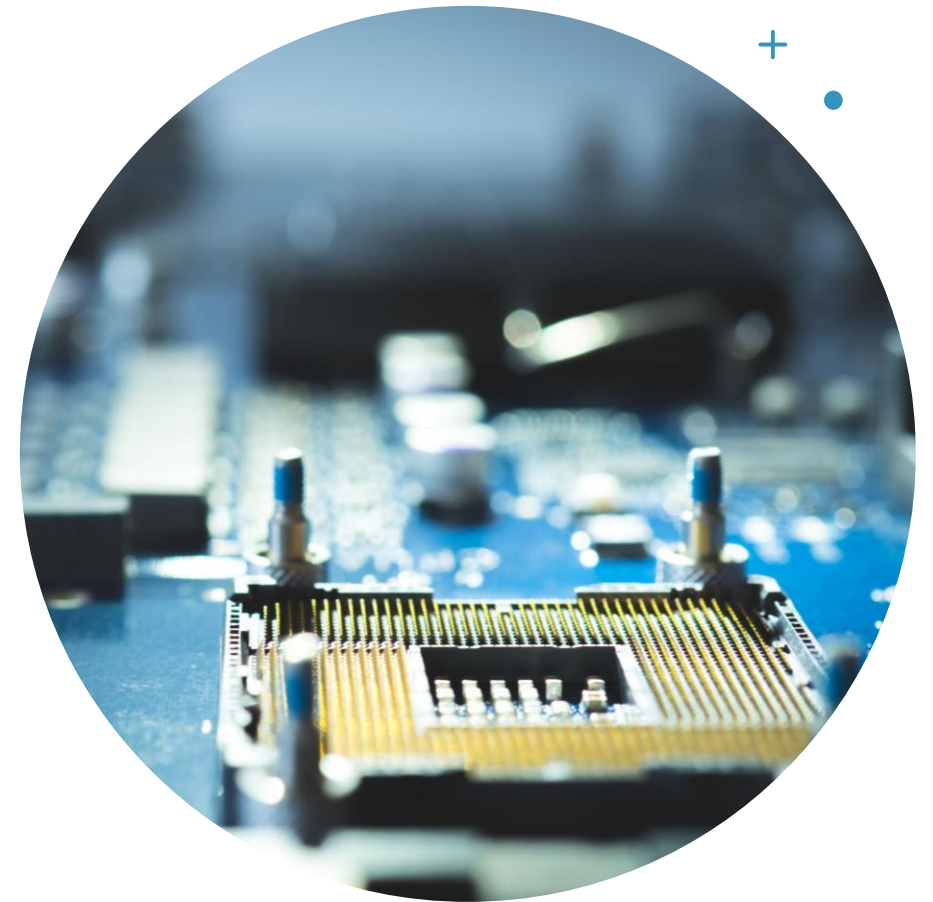
- Dalam bahasa pemrograman Python, ada beberapa jenis tipe data yang memungkinkan Anda untuk merepresentasikan berbagai jenis informasi.
- Setiap jenis tipe data memiliki karakteristik dan fungsionalitas unik yang memungkinkan Anda mengelola data dengan cara yang sesuai.



---

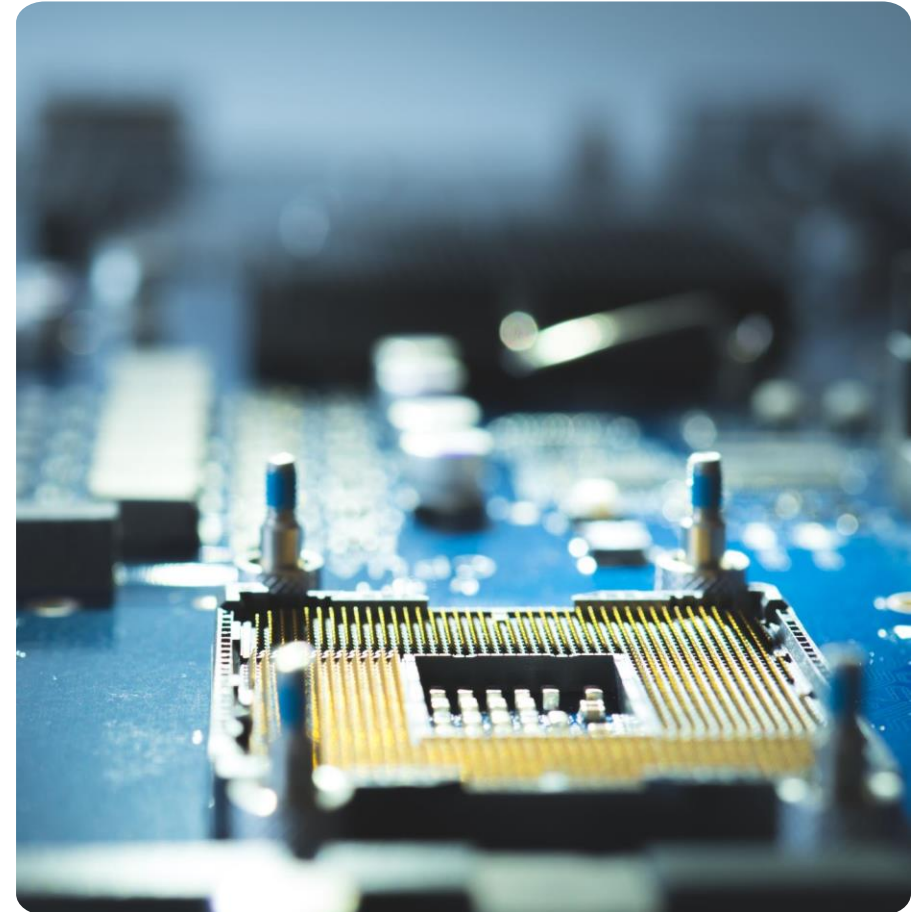
# ... Tipe Data

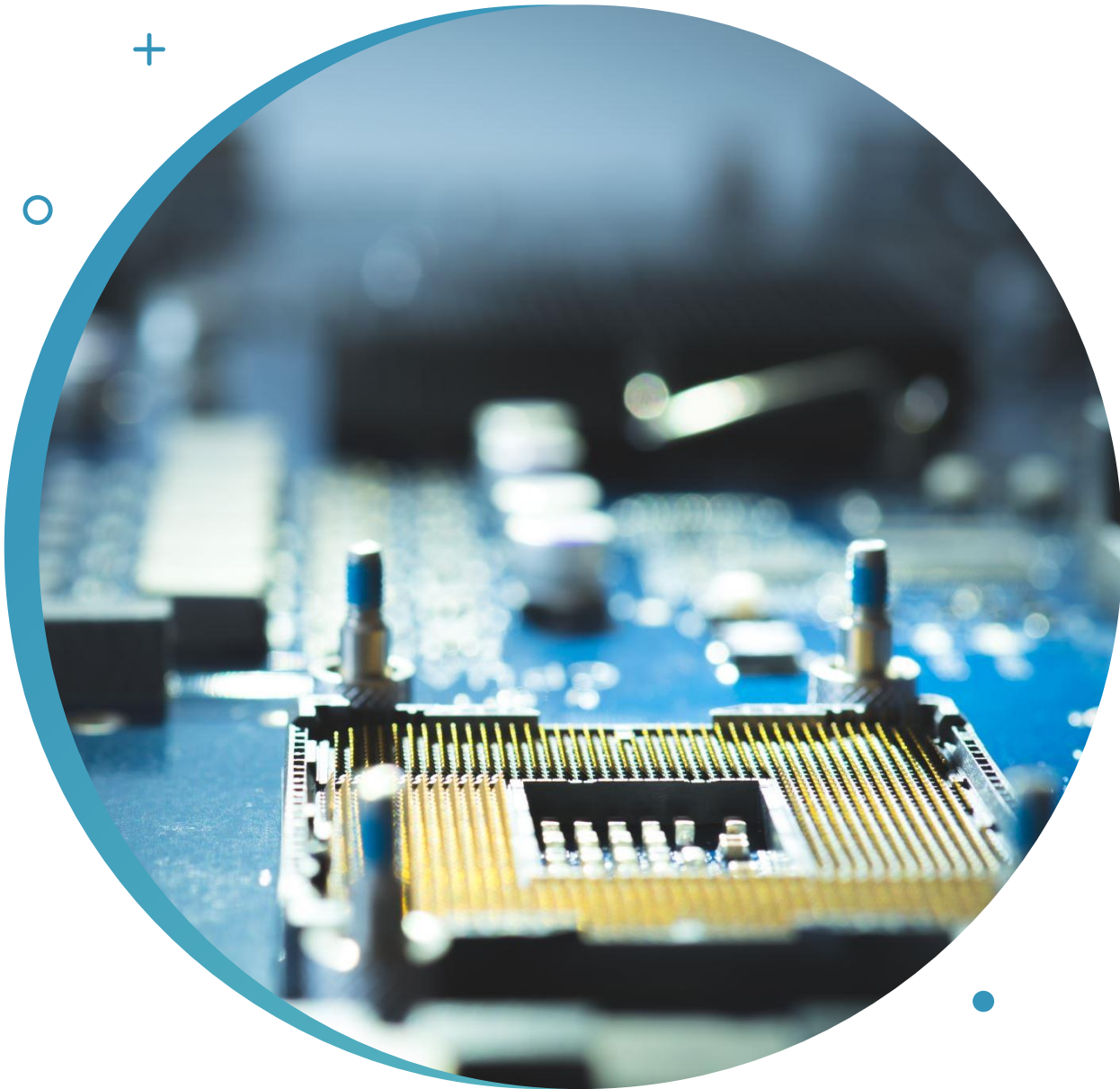
- **str (String):** Tipe data str digunakan untuk merepresentasikan teks atau karakter. Ini adalah tipe data yang digunakan untuk merepresentasikan urutan karakter, seperti kata atau kalimat.
- **int (Integer):** Tipe data int digunakan untuk merepresentasikan bilangan bulat. Ini mencakup semua bilangan bulat positif dan negatif tanpa bagian pecahan.
- **float (Floating-Point):** Tipe data float digunakan untuk merepresentasikan bilangan pecahan atau desimal. Ini dapat mengandung bagian pecahan dan eksponen.
- **bool (Boolean):** Tipe data bool digunakan untuk merepresentasikan nilai kebenaran. Ini hanya memiliki dua nilai: True (benar) dan False (salah). Digunakan untuk pengambilan keputusan dan pengendalian alur program.



## ... Tipe Data

- **list (List):** Tipe data list digunakan untuk membuat kumpulan elemen yang dapat berisi campuran tipe data lainnya. List adalah struktur data dinamis yang dapat diubah setelah dibuat.
- **tuple (Tuple):** Tipe data tuple mirip dengan list, tetapi bersifat tidak dapat diubah (immutable). Ini berarti elemen-elemen dalam tuple tidak dapat diubah setelah tuple dibuat.
- **dict (Dictionary):** Tipe data dict digunakan untuk membuat kamus atau daftar pasangan kunci-nilai. Setiap kunci mengaitkan nilai tertentu, mirip dengan kamus bahasa di mana kata adalah kunci dan artinya adalah nilai.
- **set (Set):** Tipe data set digunakan untuk membuat kumpulan elemen unik tanpa urutan tertentu. Set tidak mengizinkan adanya duplikat dalam koleksinya.





# Operator dan Operasi Dasar

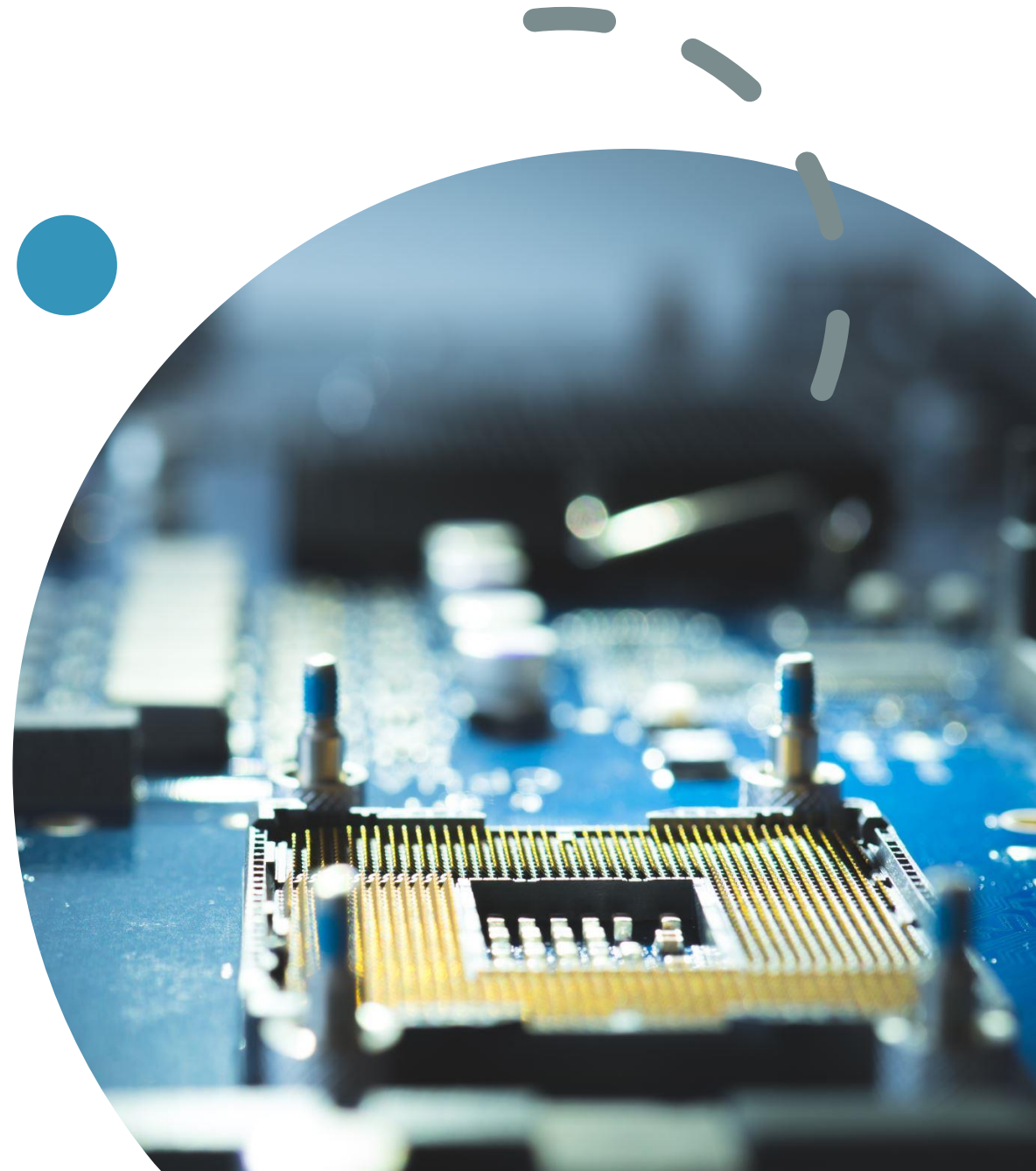
- Operasi dasar dan operator dalam Python adalah elemen penting dalam pemrograman yang memungkinkan Anda untuk melakukan berbagai manipulasi dan perhitungan pada data.
- Ini memungkinkan Anda untuk memproses dan memanipulasi data sesuai kebutuhan program Anda.



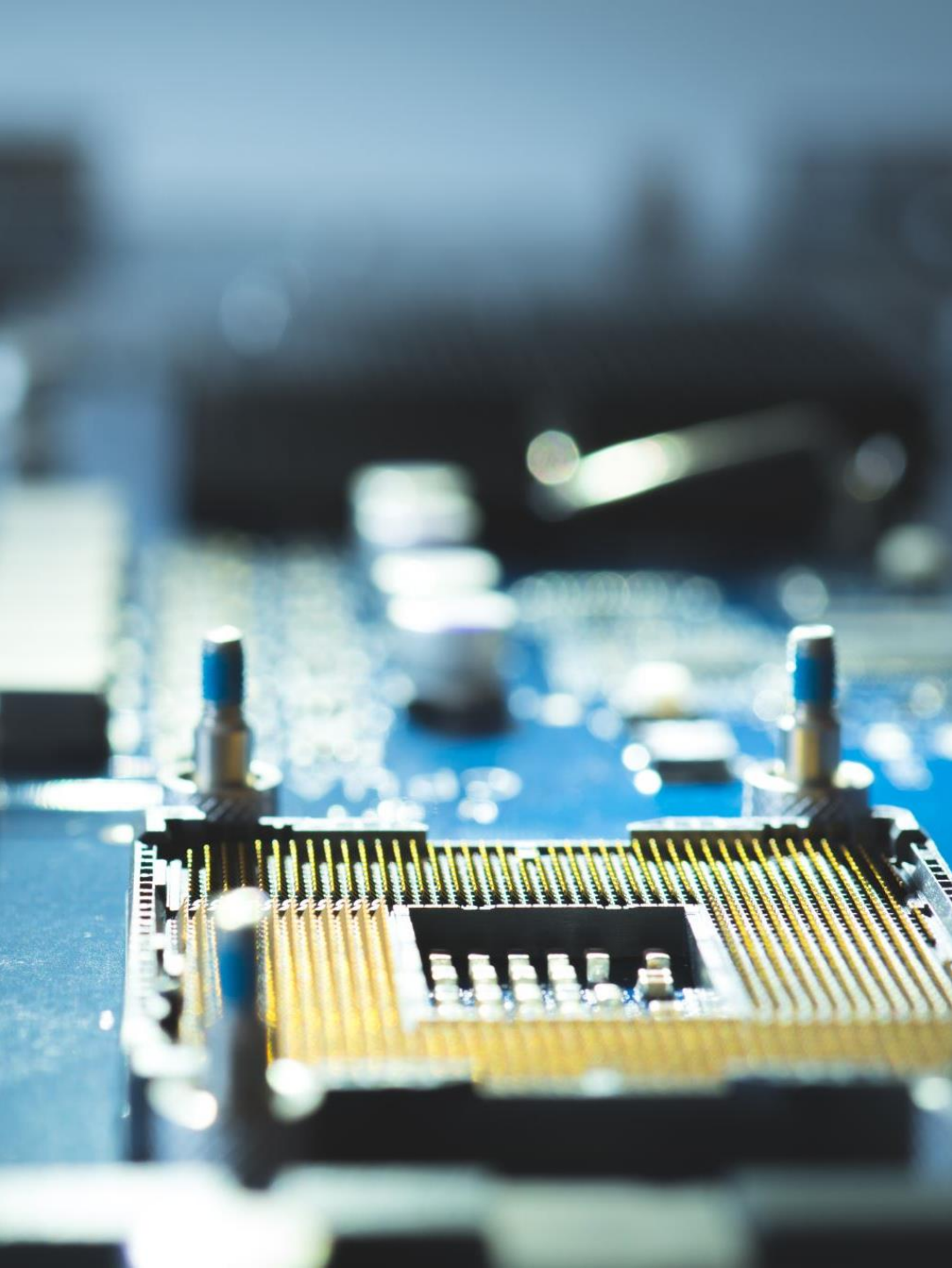
# Operasi Aritmatika

Operator aritmatika digunakan untuk melakukan operasi matematika dasar.  
Contoh:

- $+$  : Penjumlahan.
- $-$  : Pengurangan.
- $*$  : Perkalian.
- $/$  : Pembagian.
- $\%$  : Modulus (sisa hasil bagi).
- $**$  : Pangkat.







---

## Operasi Perbandingan

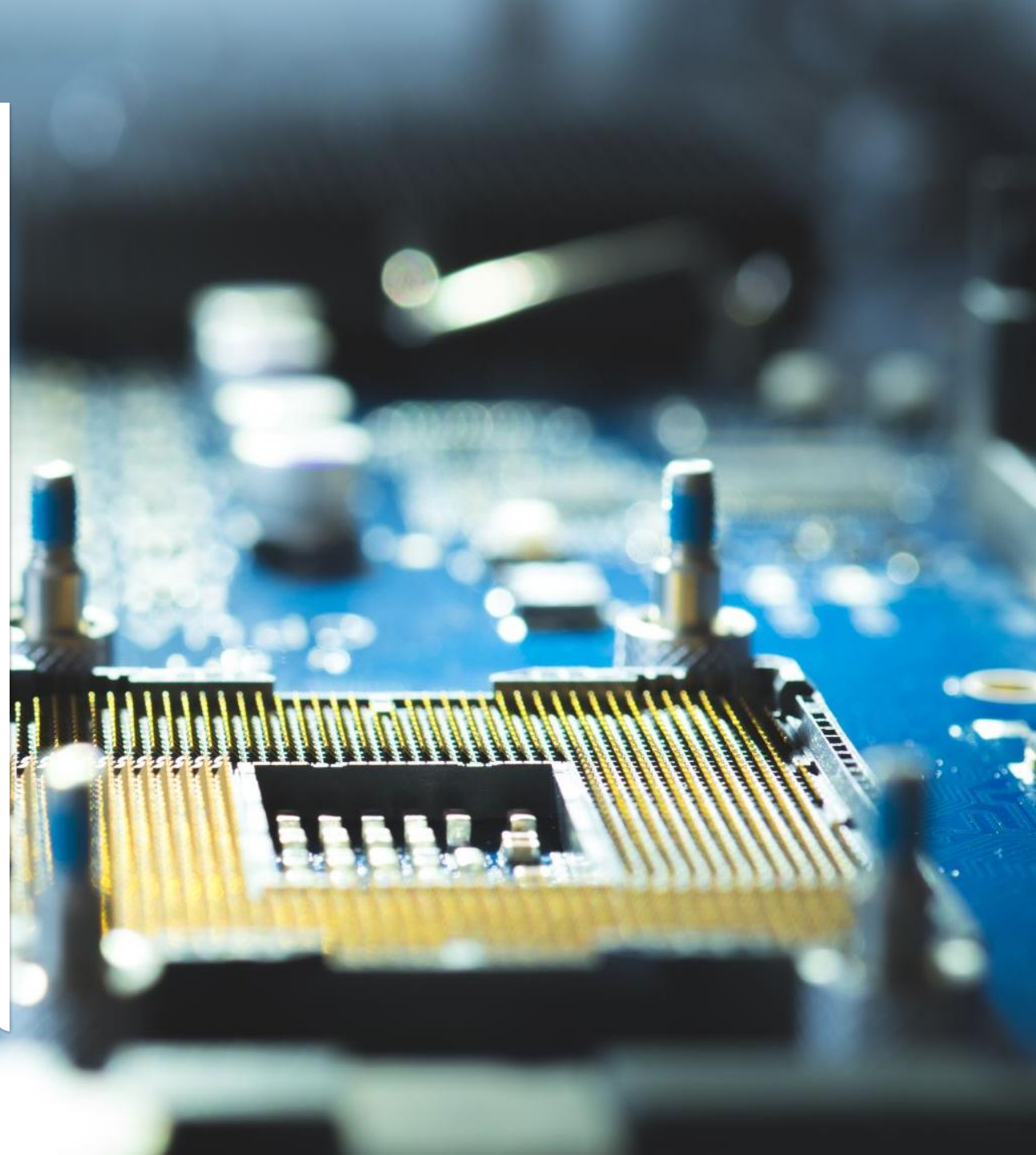
Operator perbandingan digunakan untuk membandingkan dua nilai. Hasil operasi ini adalah nilai kebenaran (True atau False). Contoh:

- `==` : Sama dengan.
- `!=` : Tidak sama dengan.
- `<` : Kurang dari.
- `>` : Lebih dari.
- `<=` : Kurang dari atau sama dengan.
- `>=` : Lebih dari atau sama dengan.

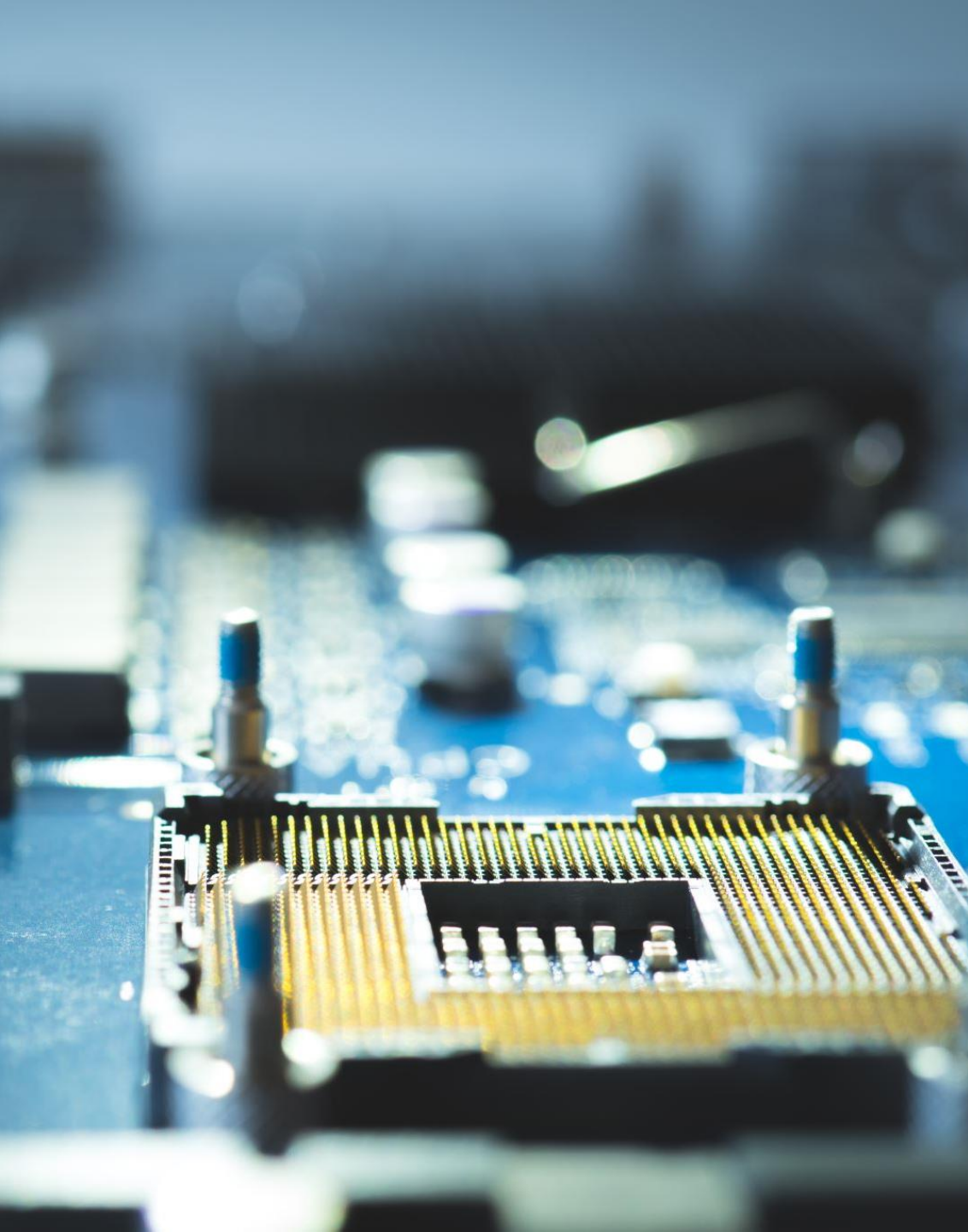
# Operasi Logika

Operator logika digunakan untuk menggabungkan dan memanipulasi nilai kebenaran. Contoh:

- and : Kedua kondisi harus benar.
- or : Salah satu kondisi harus benar.
- not : Membalikkan nilai kebenaran.







# Operasi Penugasan

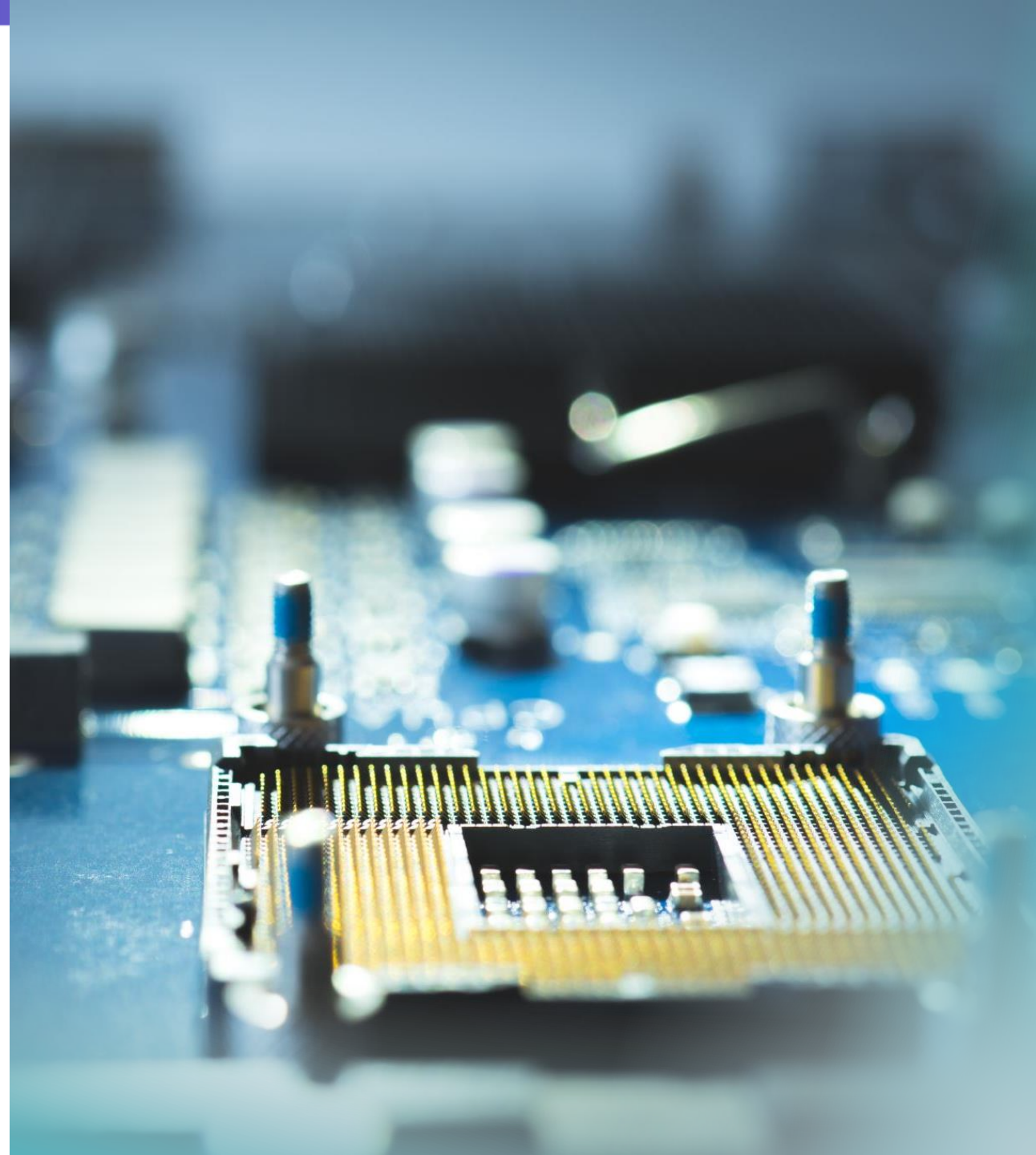
Operator penugasan digunakan untuk memberi nilai pada variabel. Contoh:

- `=` : Memberi nilai.
- `+=` : Menambah dan memberi nilai.
- `-=` : Mengurangi dan memberi nilai.
- `*=` : Mengalikan dan memberi nilai.
- `/=` : Membagi dan memberi nilai.
- `%=` : Mengambil sisa hasil bagi dan memberi nilai.

# Operasi Bitwise

Operator bitwise digunakan untuk melakukan operasi pada level bit dalam data. Contoh:

- `&` : Bitwise AND.
- `|` : Bitwise OR.
- `^` : Bitwise XOR.
- `~` : Bitwise NOT.
- `<<` : Left shift (geser ke kiri).
- `>>` : Right shift (geser ke kanan).





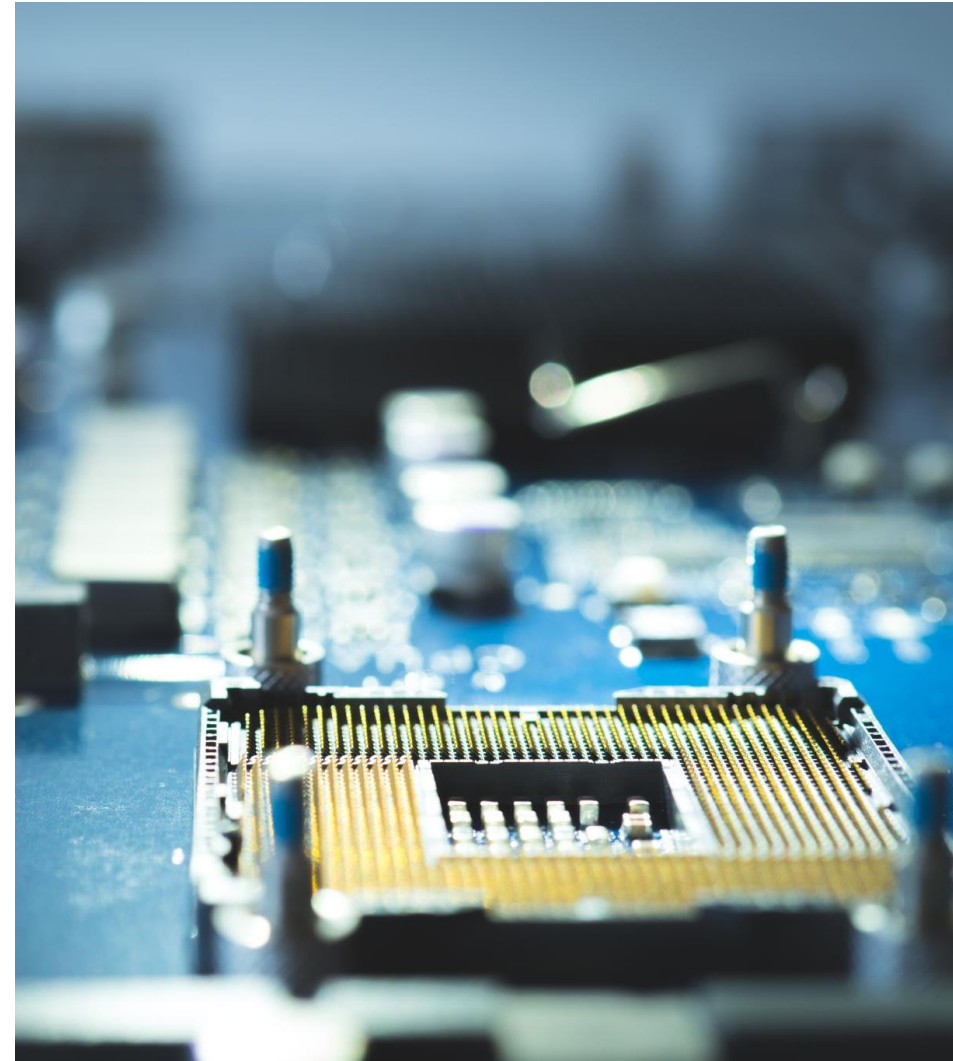


# Penggunaan Variabel

- **Penyimpanan Data:** Variabel digunakan untuk menyimpan berbagai jenis data, seperti angka, teks, nilai kebenaran, dan sebagainya. Ini memungkinkan data tersebut tetap tersedia untuk digunakan dalam program.
- **Memberi Label pada Data:** Dengan memberi nama pada variabel, Anda memberikan label pada data tersebut. Ini membuat data menjadi lebih mudah diidentifikasi dan diakses dalam kode. Misalnya, variabel "usia" bisa digunakan untuk menyimpan usia seseorang.
- **Mempermudah Penggunaan dan Pemeliharaan:** Dengan menggunakan variabel, Anda tidak perlu mengingat alamat memori yang tepat untuk setiap data. Variabel mempermudah penggunaan dan pemeliharaan kode karena Anda dapat merujuk pada data dengan nama yang berarti.

## ... Penggunaan Variabel

- **Manajemen Data:** Variabel memungkinkan Anda menyimpan data secara terstruktur dan logis. Anda dapat mengelompokkan data yang berhubungan dalam variabel-variabel tertentu, membantu dalam mengatur dan mengelola data.
- **Mengurangi Redundansi Kode:** Dengan menggunakan variabel, Anda dapat menyimpan nilai atau data sekali dan menggunakannya berkali-kali dalam berbagai bagian kode. Ini mengurangi redundansi dan membuat kode lebih ringkas





---

## Contoh Variabel dan Nilainya

- nama = "John"
  - usia = 30
  - tinggi = 175.5
  - pelajar = True
- 
- print("Nama:", nama)
  - print("Usia:", usia)
  - print("Tinggi:", tinggi)
  - print("Pelajar:", pelajar)