

# FUNGSI PERINTAH CIN DALAM BAHASA C++

## FUNGSI PERINTAH CIN C++

Perintah **cin** adalah perintah dasar C++ untuk proses input atau menerima data masukan dari user.

Dengan menggunakan perintah **cin** (di eja sebagai "see-in"), kita bisa membuat program yang lebih interaktif, yakni meminta data dari user / pengguna. Data ini bisa disimpan ke dalam variabel dan diolah lebih lanjut untuk kemudian ditampilkan kembali. Kata **cin** sendiri merupakan singkatan dari **console in**.

Format dasar perintah cin adalah sebagai berikut:

```
cin >> nama_variabel;
```

Berbeda dengan perintah **cout**, untuk perintah **cin** kita menggunakan tanda kurung siku kanan dua kali, atau tanda lebih besar dua kali, yakni karakter **>>**, kemudian diikuti dengan nama variabel yang akan menampung nilai inputan tersebut.

## CONTOH KODE PROGRAM CIN C++

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main()
6 {
7     int harga;
8
9     cout << "Masukkan harga buku: ";
10    cin >> harga;
11
12    cout << "Harga buku C++ adalah: "<< harga << endl;
13    return 0;
14 }
```

Di baris 7, saya mendeklarasikan variabel **harga** dengan tipe data **int**. Ini berarti variabel **harga** bisa diisi dengan angka bulat seperti 1, 4, atau 10000.

Kemudian di baris 9 terdapat perintah **cout** yang akan menampilkan teks "Masukkan harga buku:". Ini adalah instruksi kepada user aplikasi untuk menginput sesuatu.

Perintah **cin >> harga** di baris 10 akan menyebabkan layar console atau cmd berhenti sejenak menunggu user menginput angka. Ketika user mengetik angka dan menekan tombol Enter, maka nilai tersebut akan ditampung ke dalam variabel **harga**.

## CONTOH KODE PROGRAM CIN C++

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int harga;
8
9      cout << "Masukkan harga buku: ";
10     cin >> harga;
11
12     cout << "Harga buku C++ adalah: "<< harga << endl;
13     return 0;
14 }
```

Di baris 7, saya mendeklarasikan variabel **harga** dengan tipe data **int**. Ini berarti variabel **harga** bisa diisi dengan angka bulat seperti 1, 4, atau 10000.

Kemudian di baris 9 terdapat perintah **cout** yang akan menampilkan teks "Masukkan harga buku:". Ini adalah instruksi kepada user aplikasi untuk menginput sesuatu.

Perintah **cin >> harga** di baris 10 akan menyebabkan layar console atau cmd berhenti sejenak menunggu user menginput angka. Ketika user mengetik angka dan menekan tombol Enter, maka nilai tersebut akan ditampung ke dalam variabel **harga**.

# MENGINPUT BANYAK DATA DENGAN COUT

Jika kita ingin menginput lebih dari 1 data, cukup siapkan variabel untuk menampung setiap nilai inputan tersebut:

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      string nama;
8      int umur;
9
10     cout << "Masukkan nama: ";
11     cin  >> nama;
12
13     cout << "Masukkan umur: ";
14     cin  >> umur;
15
16     cout << "\n";
17
18     cout << "Selamat Belajar C++ " << nama << ", Semangat!!" << endl;
19     cout << "Usia anda saat ini " << umur << " tahun" << endl;
20     return 0;
21 }
```

Dalam kode program ini saya meminta user untuk menginput 2 data: **nama** dan **umur**. Karena nama berisi teks, maka tipe datanya adalah **string** seperti di baris 7. Berikut hasil kode program di atas:

# MENGINPUT DATA STRING DENGAN FUNCTION GETLINE()

Dalam bahasa C++, perintah cin tidak bisa memproses nilai inputan teks yang mengandung spasi. Untungnya terdapat solusi praktis untuk masalah ini, yakni dengan function `getline()`.

Berikut contoh penggunaannya:

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      string nama;
8      int umur;
9
10     cout << "Masukkan nama: ";
11     getline(cin,nama);
12
13     cout << "Masukkan umur: ";
14     cin >> umur;
15
16     cout << "\n";
17
18     cout << "Selamat Belajar C++ " << nama << ", Semangat!!" << endl;
19     cout << "Usia anda saat ini " << umur << " tahun" << endl;
20     return 0;
21 }
```

