# Contoh Program Array C++ dan Penjelasannya

Dibandingkan anda harus mendeklarasikan banyak variabel dengan nama yang hampir sama misalnya nomor1, nomor2, ... , nomor10 maka lebih baik menggunakan array.

Berikut informasi mengenai array:

**Secara pengertian:** Array atau larik (dalam Bahasa Indonesia) adalah sebuah tipe data bentukan yang terdiri dari sejumlah komponen dengan tipe yang sama.

Dengan menggunakan array kita dapat menyimpan banyak data dengan satu nama.

*(Silahkan baca:*[*Penjelasan lengkap tentang Array*](https://kodedasar.com/pengertian-array/)*)*

**Table Of Contents**

* [Deklarasi & Inisialisasi Array](https://kodedasar.com/array-cpp/#deklarasi--inisialisasi-array)
* [#1 Contoh Program C++ Array 1 Dimensi](https://kodedasar.com/array-cpp/#1-contoh-program-c-array-1-dimensi)
* [#2 Contoh Program C++ Array 2 Dimensi](https://kodedasar.com/array-cpp/#2-contoh-program-c-array-2-dimensi)
* [Menarik Kesimpulan](https://kodedasar.com/array-cpp/#menarik-kesimpulan)

**Deklarasi & Inisialisasi Array**

Untuk mendeklarasikan sebuah array pada c++ anda hanya perlu menuliskan:

tipe\_data nama\_array [ ukuran\_array ];

Saya akan memberikan sedikit catatan:

* **Tipe Data** - Sebagai catatan, tipe data yang digunakan pada array adalah tipe data primitif. Silahkan baca: [perbedaan tipe data primitif dan komposit](https://kodedasar.com/pengertian-tipedata/).
* **Nama Array** - Anda bebas memberi nama array yang akan digunakan.
* **Ukuran Array** - Digunakan untuk menentukan banyak index, anda dapat menentukan sesuai kebutuhkan anda. Namun harus lebih dari 0.

Berikut adalah contoh pendeklarasian tipe data array pada program c++:

int daftarNilai [20];

Faktanya, anda dapat melakukan inisialisasi terhadap array atau memberikan nilai awal:

int daftarNilai [5] = {10,9,10,8,9};

Dengan catatan, jumlah inisialisasi tidak boleh lebih dari ukuran array.

Namun anda juga dapat mendeklarasikan array sekaligus memberikan inisialisasi tanpa harus memberikan tambahan deklarasi ukuran array, sebagai berikut:

int daftarNilai [] = {10,9,8,10,9};

**#1 Contoh Program C++ Array 1 Dimensi**

Berikut cara mendeklarasikan array 1 dimensi pada C++:

tipedata [totalindex];

//atau

tipedata [] = { nilai1, nilai2, dst };

Jadi, ada dua cara mendeklarasikan sebuah array berdimensi satu cara pertama cukup dengan menggunakan tipe data dan total index yang diinginkan, contoh:

int umur[5];

Artinya, anda menyediakan varibel array bernama umur dengan tipe data integer yang mampu menampung 5 data yang sama mulai dari index 0 - 4.

**Ingat**: Index pada array selalu dimulai dari nol.

Kemudian untuk mengisi nilai untuk tiap index pada array dengan cara:

umur[0] = 18;

umur[1] = 19;

umur[2] = 20;

umur[3] = 21;

umur[4] = 22;

Cara kedua, anda bisa langsung memberi nilai untuk tiap index pada saat pendeklarasian array. Contoh kodenya sebagai berikut:

int umur[] = {18,19,20,21,22};

Untuk penerapan array dalam program biasanya dipadukan dengan [perulangan for, while-do, dan do-while](https://kodedasar.com/perulangan-cpp/). Pilih salah satu namun yang paling umum adalah for.

Berikut contoh program sederhana array 1 dimensi pada c++:

#include <iostream>

using namespace std;

int umur[] = {18, 19, 20, 21, 22};

int n, hasil=0;

int main ()

{

for ( n=0 ; n<5 ; ++n )

{

hasil+= umur[n];

}

cout << hasil;

return 0;

}

Dalam program diatas, saya ingin menjumlahkan nilai dari tiap index umur dan hasilnya saya simpan kedalam variabel hasil untuk ditampilkan di akhir program.

**#2 Contoh Program C++ Array 2 Dimensi**

Sama halnya dengan array 1 dimensi, array 2 dimensi juga memiliki cara yang sama dalam pendeklarasiannya. Berikut contoh array berdimensi 2:

Deklarasi array 2 dimensi:

tipedata [totalbaris][totalkolom];

Pemberian nilai array 2 dimensi:

namatabel [indexbaris][indexkolom] = nilai;

Sehingga untuk memberi nilai secara manual menjadi sebagai berikut:

tabel[0][0]= 1; tabel[0][1]= 2; tabel[0][2]= 3; tabel[0][3]= 4; tabel[0][4]= 5;

tabel[1][0]= 2; tabel[1][1]= 4; tabel[1][2]= 6; tabel[1][3]= 8; tabel[1][4]= 10;

tabel[2][0]= 3; tabel[2][1]= 6; tabel[2][2]= 9; tabel[2][3]= 12; tabel[2][4]= 15;

**Catatan**: anda akan menggunakan looping bersarang untuk mengakses array dua dimensi (array dua dimensi seperti sebuah tabel dengan baris dan kolom).

Berikut ini merupakan contoh program sederhana array 2 dimensi pada c++:

#include <iostream>

using namespace std;

int x,y;

int tabel[3][5] = {

{1, 2, 3, 4, 5},

{2, 4, 6, 8, 10},

{3, 6, 9, 12, 15}

};

int main ()

{

for (x=0; x<3; x++)

for (y=0; y<5; y++)

{

cout << tabel[x][y] << " ";

}

cout << "\n";

}

Dalam program diatas kita mendeklarasikan variabel x dan y sebagai iterator.

Lalu kita mendeklarasikan varibel tabel dengan tipe array dua dimensi (jumlah baris = 3, jumlah kolom = 5). Selain itu, kita langsung melakukan inisialisasi pada array tersebut.

Kemudian dalam program utama kita menggunakan nested loop dengan fungsi for untuk mengakses baris dan kolom secara otomatis supaya dapat mencetak isi array tabel.

**Menarik Kesimpulan**

Keunggulan array dibandingkan variabel biasa akan anda rasakan dalam fleksibilitas program, sebagai contoh ketika anda ingin menampung daftar nama mahasiswa.

Dibandingkan anda membuat banyak variabel misalnya: nama1, nama2, nama2, dan seterusnya. Akan lebih baik jika kita menggunakan tipe data array.

Karena kita dapat menyimpan data-data tersebut menjadi satu nama variabel.