Pengertian Manipulator

Manipulator adalah fungsi pembantu yang memungkinkan untuk mengontrol input / output stream. Manipulator pada umumnya digunakan untuk mengatur tampilan layar. Ada beberapa fungsi manipulator yang disediakan oleh C++, antara lain. endl, end, flush( ), dec( ), hex( ), oct( ), setbase( ), setw( ), setfill( ), setprecision( ) dan setosflags( ).

Macam-macam Manipulator

endl

endl merupakan suatu fungsi manipulator yang digunakan untuk menyisipkan karakter NewLine atau mengatur pindah baris (\n). Fungsi ini sangat berguna untuk piranti keluaran berupa file di disk. File header yang harus disertakan adalah file header iostream.

Contoh Program:

#include <iostream >

using namespace std;

int main()

{

char nim[9]="121213141", nama[15]="Belajarcpp.com";

char alamat[20]="Purwokerto";

cout<<"Nim :"<<nim<<endl;

cout<<"Nama :"<<nama<<endl;

cout<<"Alamat :"<<alamat<<endl;

return 0;

}

ends

ends merupakan suatu fungsi manipulator yang digunakan untuk menambah karakter null ( nilai ASCII NOL ) kederetan suatu karakter. Fungsi ini akan berguna untuk mengirim sejumlah karakter ke file di disk atau modem dan mangakhirinya dengan karakter NULL. File header yang harus disertakan adalah file header iostream.

Contoh :

#include <iostream>

using namespace std;

int main( )

{

int a, b, c, d;

cout<<"Masukan Nilai A : "; cin>>a;

cout<<"Masukan Nilai B : "; cin>>b;

c = a % b;

d = a \* b;

cout<<"Hasil dari C = A % B adalah "<<c<<ends;

cout<<"Hasil dari D = A \* B adalah "<<d<<ends;

return 0;

}

dec, oct dan hex

dec, oct dan hex merupakan suatu fungsi manipulator yang digunakan untuk menampilkan data dalam bentuk desimal(bilangan berbasis 10), oktal(bilangan berbasis 8) dan hexadesimal(bilangan berbasis 16). File header yang harus disertakan adalah file header <iomanip> .

Contoh :

# include <iostream>

# include <iomanip>

using namespace std;

int main()

{

int nilai = 15320;

cout<<"Nilai Awal= "<<nilai<<endl;

cout<<"Nilai ke Octal = "<<oct<<nilai<<endl;

cout<<"Nilai ke Hexadesimal = "<<hex<<nilai<<endl;

cout<<"Nilai ke Desimal = "<<dec<<nilai<<endl;

return 0;

}

setprecision() & cout.precision()

Fungsi setprecision() dan cout.precision() adalah suatu fungsi manipulator yang digunakan untuk mengatur jumlah digit desimal yang ingin ditampilkan. Fungsi ini biasa pada fungsi cout( ), untuk menggunakan setprecision() kita membutuhkan deklarasi file header <iomanip> sedangkan cout.precision() cukup menggunakan library <iostream>.

Bentuk Penulisan

setprecision(angka)

//Atau

cout.precision(angka)

Contoh :

# include <iostream>

# include <iomanip>

using namespace std;

int main( ){

float a,b,c;

a = 25.23;

b = 12.54;

c = a \* b;

cout<<setiosflags(ios::fixed);

cout<<setprecision(0)<<c<<endl;

cout<<setprecision(1)<<c<<endl;

cout<<setprecision(2)<<c<<endl;

cout<<setprecision(3)<<c<<endl;

cout<<setprecision(4)<<c<<endl;

cout<<setprecision(5)<<c<<endl;

return 0;

}

setbase( )

setbase( ) merupakan suatu fungsi manipulator yang digunakan untuk konversi bilangan Octal, Decimal dan Hexadecimal. File header yang harus disertakan file header <iomanip>.

Bentuk penulisannya :

setbase(basis bilangan);

Base bilangan merupakan base dari masing-masing bilangan, yaitu :

• Octal = basis 8

• Decimal = basis 10

• Hexadecimal = basis 16

Contoh proram :

#include <iostream>

#include <iomanip>

using namespace std;

int main( )

{

int a = 250;

cout<<"Penggunan Manipulator setbase()"<<"\n\n";

cout<<"Decimal Octal Hexadecimal"<<endl;

cout<<"----------------------------"<<"\n";

for(a=250;a<=260;a++)

{

cout<<setbase(10);

cout<<a<<endl;

cout<<setbase(8);

cout<<a<<endl;

cout<<setbase(16);

cout<<a<<endl;

}

return 0;

}

setw() & cout.width(n)

setw() & cout.width(n) merupakan suatu fungsi manipulator yang digunakan untuk mengatur lebar tampilan dilayar dari suatu nilai variabel. Untuk setw() dibutuhkan deklarasi File header <iomanip> dan untuk cout.width() cukup menggunakan library <iostream>.

Bentuk penulisannya :

setw(n);

//atau

cout.width(n) ;

keterangan :

n = merupakan nilai lebar tampilan data, integer.

Contoh Program :

// Penggunaan Manipulator setw()

#include <iostream>

#include <iomanip>

using namespace std;

int main( )

{

int a;

cout<<"Penggunan Manipulator setw()"<<endl<<endl;

cout<<"-------------------------------"<<endl;

for(a=1;a<=15;a++)

{

cout<<setw(a)<<a<<endl;

}

cout<<"-------------------------------"<<endl;

return 0;

}

setfill()

setfill() merupakan suatu fungsi manipulator yang digunakan untuk manampilkan suatu karakter yang diletakan didepan nilai yang diatur oleh fungsi setw(). File header yang harus disertakan adalah file header <iomanip>.

Bentuk penulisannya :

setfill(charakter);

Contoh Program :

// penggunaan setfill dan setw( )

# include <iostream>

# include <iomanip>

using namespace std;

int main(){

int a;

cout<<"Penggunan Manipulator setfill()"<<"\n\n";

cout<<"-------------------------------"<<"\n";

for(a=1;a<=15;a++)

{

cout<<setfill('-');

cout<<setw(a)<<a<<endl;

}

return 0;

}

setiosflags() dan setf()

Fungsi setiosflags() dan setf() merupakan suatu fungsi manipulator yang digunakan untuk mengatur sejumlah format keluaran data. Fungsi ini biasa pada fungsi cout(), untuk setiosflags() menggunakan library <iomanip> dan untuk setf() bisa hanya menggunakan library <iostream>.

Dan untuk mengembalikan sejumlah pengaturan fungsi anda dapat menggunakan resetiosflags() dan unsetf(). Untuk file headernya sama seperti diatas. untuk resetiosflags() menggunakan library <iomanip> dan untuk unsetf() bisa hanya menggunakan library <iostream>.

Ada beberapa format keluaran untuk fungsi setiosflags( ), antara lain.

1. Tanda Format Perataan Kiri dan Kanan, Tedapat dua buah tanda format yang digunakan untuk perataan kiri dan kanan,pengaturan terhadap lebar variabel untuk perataan kiri dan kanan ini melalui

fungsi setw( ).

- ios::left digunakan untuk mengatur perataan sebelah kiri

- ios::right digunakan untuk mengatur perataan sebelah kanan

Contoh Program:

//tanda format ios::left dan ios::right

#include <iostream>

#include <iomanip>

using namespace std;

int main( )

{

int a = 75, b = 56;

cout<<"Penggunaan ios::left dan ios::right\n\n";

cout<<"Rata Sebelah Kiri = ";

cout<<setiosflags(ios::left)<<setw(10)<<a;

cout<<setiosflags(ios::left)<<setw(10)<<b;

cout<<endl;

cout<<"Rata Sebelah Kanan = ";

cout<<setiosflags(ios::right)<<setw(10)<<a;

cout<<setiosflags(ios::right)<<setw(10)<<b;

return 0;

}

2. Tanda Format Keluaran Notasi Konversi, Tanda format yang digunakan untuk keluaran Notasi, yaitu:

- ios::scientific digunakan untuk mengatur keluaran dalam bentuk notasi eksponensial.

- ios::fixed digunakan untuk mengatur keluaran dalam bentuk notasi desimal.

Contoh Program :

//tanda format ios::scientific dan ios::fixed

#include <iostream>

#include <iomanip>

using namespace std;

int main( ){

cout<<"Penggunaan ios::scientific dan ios::fixed\n";

cout<<"\nHasil ios::scientific dari 75.45 adalah ";

cout<<setiosflags(ios::scientific)<<75.45<<endl;

cout<<"Hasil ios::fixed dari 56.65 adalah ";

cout<<setiosflags(ios::fixed)<<56.65;

return 0;

}

3. Tanda Format Konversi Dec, Oct dan Hex, Terdapat tiga macam tanda format yang digunakan untuk konversi keluaran dalam basis Decimal, Octal dan Hexadecimal, yaitu:

- ios::dec digunakan untuk mengatur keluaran dalam konversi basis desimal.

- ios::oct digunakan untuk mengatur keluaran dalam konversi basis oktal.

- ios::hex digunakan untuk mengatur keluaran dalam konversi basis heksadesimal.

Contoh Program :

//tanda format ios::dec, ios::oct, ios::hex

#include <iostream>

#include <iomanip>

using namespace std;

int main(){

cout<<"Bilangan Decimal dar 75 = ";

cout<<setiosflags(ios::dec)<<75<<endl;

cout<<"Bilangan Octal dari 10 = ";

cout<<setiosflags(ios::oct)<<10<<endl;

cout<<"Bilangan Hexadecimal dari 15 = ";

cout<<setiosflags(ios::hex)<<15;

return 0;

}

4. Tanda Format Manipulasi Huruf Hexadecimal, untuk keperluan memanipulasi atau mengubah huruf pada notasi hexadecimaldengan menggunakan tanda format:

- ios::uppercase digunakan untuk mengubah huruf pada notasi huruf hexadecimal.

Contoh Program :

//tanda format ios::uppercase

#include <windows.h>

#include <iostream>

#include <iomanip>

using namespace std;

void gotoxy(int x, int y){

COORD coord;

coord.X=x;

coord.Y=y;

SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE),coord);

}

int main(){

int a;

cout<<"Penggunaan ios::uppercase\n";

cout<<"-------------------------\n";

cout<<"Tanpa Dengan \n";

cout<<"Konversi Konversi \n";

cout<<"-------------------------\n";

for (a=1; a<=15; a++)

cout<<hex<<a<<endl;

for (a=1; a<=15; a++)

{

gotoxy(15,a+5);

cout<<setiosflags(ios::uppercase)<<hex<<a<<endl;

}

return 0;

}

5. Tanda Format Keluaran Dasar Bilangan Hexadecimal dan Octal, untuk keperluan menampilkan dasar bilangan Hexadecimal dan Oktal dengan menggunakan tanda format:

- ios::showbase digunakan untuk menampilkan tanda 0x (nol-x) diawal pada tampilan bilangan hexadecimal dan 0 (nol) diawal pada tampilan bilangan decimal.

Contoh Program :

#include <windows.h>

#include <iostream>

#include <iomanip>

using namespace std;

void gotoxy(int x, int y){

COORD coord;

coord.X=x;

coord.Y=y;

SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE),coord);

}

int main(){

int a;

cout<<"Penggunaan ios::showbase\n";

cout<<"----------------------------\n";

cout<<"Decimal Hexadecimal Oktal \n";

cout<<"----------------------------\n";

cout<<setiosflags(ios::showbase);

for (a=1; a<=15; a++)

{

gotoxy(4,a+5);

cout<<dec<<a<<endl;

}

for (a=1; a<=15; a++)

{

gotoxy(15,a+5);

cout<<hex<<a<<endl;

}

for (a=1; a<=15; a++)

{

gotoxy(25,a+5);

cout<<oct<<a<<endl;

}

cout<<"----------------------------\n";

return 0;

}

6. Tanda Format Menampilkan Titik Desimal, untuk keperluan menampilkan titik desimal dengan menggunakan tanda format:

- ios::showpoint digunakan untuk menampilkan titik desimal pada bilangan

yang tidak mempunyai titik desimal pada tipe data float atau double.

Contoh Program :

//tanda format ios::showpoint

# include <iostream>

# include <iomanip>

using namespace std;

int main()

{

double a = 78;

//-> tanpa tanda format ios::showpoint

cout<<"Tanpa tanda format ios::showpoint"<<endl;

cout<<"Variabel a = "<<a<<"\n\n";

//-> dengan tanda format ios::showpoint

cout<<"Dengan tanda format ios::showpoint"<<endl;

cout<<setiosflags(ios::showpoint);

cout<<"Variabel a = "<<a<<endl;

return 0;

}

7. Tanda Format Menampilkan Simbol Plus ( + ), berfungsi untuk keperluan menampilkan simbol Plus ( + ) pada bilangan genap dengan

menggunakan tanda format:

- ios::showpos digunakan untuk menampilkan simbol plus (+) pada variabel yang memiliki nilai bilangan positif.

Contoh Program :

//tanda format ios::showpos

# include <iostream>

# include <iomanip>

using namespace std;

int main()

{

int a = 8, b = -9;

cout<<"Tanpa Menggunakan ios::showpos"<<"\n\n";

cout<<"Nilai a = "<<a<<" Nilai b = "<<b<<endl;

cout<<"\n\n";

cout<<setiosflags(ios::showpos);

cout<<"Dengan Menggunakan ios::showpos"<<"\n\n";

cout<<"Nilai a = "<<a<<" Nilai b = "<<b<<endl;

return 0;

}

Contoh Program :

#include <iostream>

using namespace std;

int main ( )

{

double y = 1234.56789 ;

cout << "menuliskan bilangan real dengan presisi berbeda" ;

cout << " \n Presisi 3 = " ;

cout.precision (3) ;

cout.width (15) ;

cout << y ;

cout << " \n Presisi 7 = " ;

cout.precision (7) ;

cout.width (15) ;

cout << y ;

cout << " \n \n Menggunakan notasi scientific / fixed " ;

cout.setf (ios :: scientific | ios :: showpos) ;

cout << " \n Scientific = " << y ;

cout.setf (ios :: fixed) ;

cout << " \n Fixed = " << y ;

cout.unsetf (ios :: scientific | ios :: fixed | ios :: showpos) ;

cout << " \n \n Menggunakan shompoint " ;

double z = 123 ;

cout.setf (ios :: showpoint) ;

cout << " \n Showpoint aktif = " << z ;

cout.unsetf (ios :: showpoint) ;

cout << "\n Showpoint non aktif : " << z ;

return 0;

}

Cukup sekian dari belajarcpp.com semoga artikel ini bisa bermanfaat untuk kita semua. Mohon maaf atas kekuranganya dan Terima Kasih atas dukungan dan kunjungan anda ke BelajarCPP. Have a nice day.

Last update : 3 Februari 2016