



MODUL #04

PROGRAMMING STRUCTURE

KOMENTAR DAN HEADER PADA C++

NUR ALAMSYAH

KOMENTAR DALAM C++

Tidak hanya pada bahasa pemrograman C++ yang menyediakan fitur Comment (Komentar), hampir di setiap bahasa pemrograman menyediakan fitur Comment. Semua karakter yang tergolong dalam komentar tidak akan dibaca oleh compiler, dan tidak dilaksanakan sebagai perintah oleh compiler atau dengan nama lain akan diabaikan.

Komentar dalam kode program biasanya digunakan oleh para programmer untuk memberi tanda atau keterangan pada kode program yang dia kembangkan, agar mudah dibaca, dipahami dan dimengerti terutama untuk programmer lainnya jika kita sedang bekerja dalam suatu team, karena tidak semua programmer mempunyai jalan pikiran yang sama dalam pembuatan algoritma, hal ini sangat beguna untuk memberitahu dengan menggunakan bahasa manusia dan mempermudahkan pada orang untuk mempelajari apa yang ada di dalam kode program yang telah kita tulis.

C++ menyediakan **dua macam fitur komentar** yaitu **single-line** dan **multi-line**. kedua cara tersebut bisa kita gunakan dan tidak akan mempengaruhi jalanya program, atau dengan kata lain yaitu compiler akan mengabaikan semua karakter dalam bentuk komentar yang anda buat.

Single-line : dengan tanda //

Cara komentar diatas akan membuat semua karakter setelah tanda // dan hanya di baris itu akan berlaku menjadi komentar. Biasanya cara ini digunakan untuk menandai atau memberi keterangan pada kode di baris itu.

Contoh Penulisan:

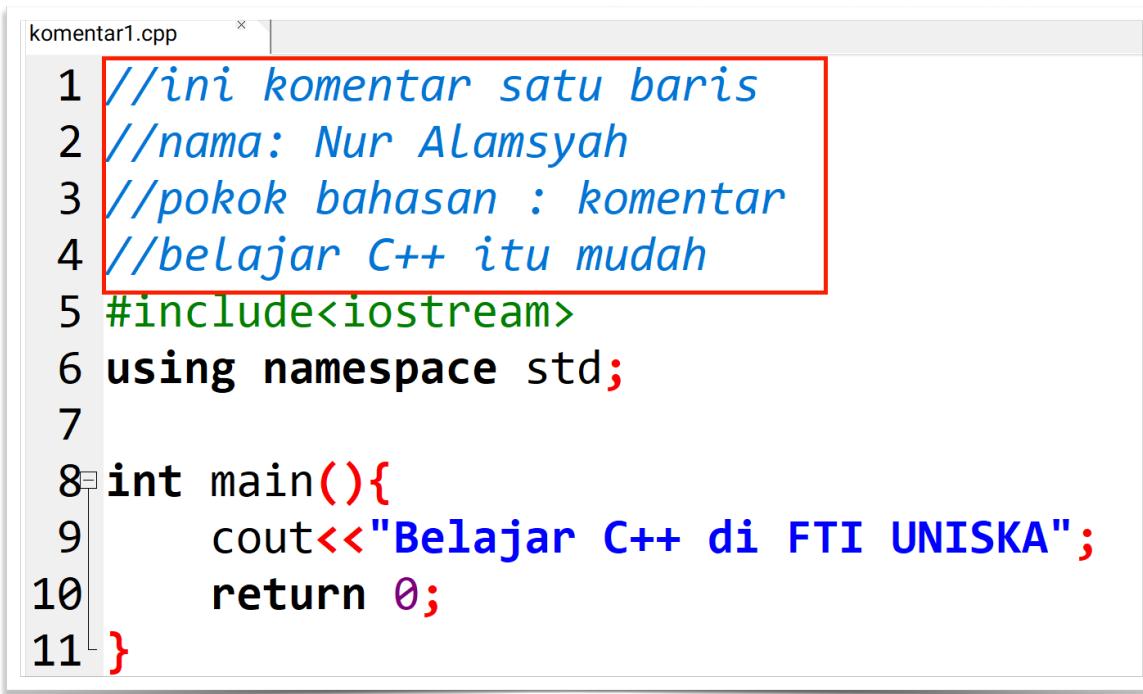
Praktikan: Buat Folder baru beri nama **Pertemuan 4 Komentar dan dasar IO. Buka Aplikasi Dev C++ kemudian buat file baru** beri nama **komentar1.cpp**, ketikkan kode berikut:

```

komentar1.cpp
1 #include<iostream>
2 using namespace std; //ini komentar satu baris
3
4 int main(){
5     cout<<"Belajar C++ di FTI UNISKA"; //ini juga komentar
6     return 0;
7 }

```

Atau tambakan komentar diatas `#include <iostream>`



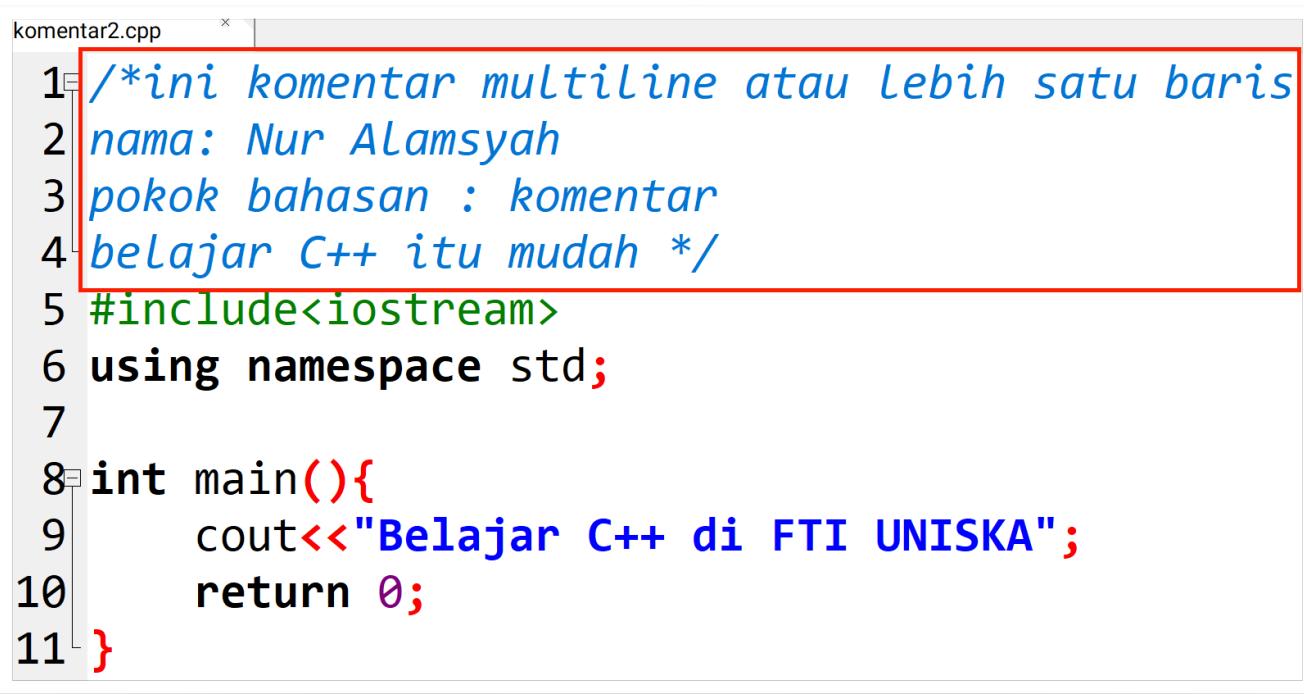
```
1 //ini komentar satu baris
2 //nama: Nur Alamsyah
3 //pokok bahasan : komentar
4 //belajar C++ itu mudah
5 #include<iostream>
6 using namespace std;
7
8 int main(){
9     cout<<"Belajar C++ di FTI UNISKA";
10    return 0;
11 }
```

Multi-line : diapit oleh /* dan */

Merupakan fitur komentar yang diturunkan oleh Bahasa Pemrograman C, cara di atas akan membuat semua karakter di antara tanda `/*` dan `*/` akan berlaku sebagai komentar. Jika cara *single line* tidak cukup untuk memberi keterangan (komentar) yang membutuhkan berbaris-baris atau membuat komentar menggunakan break line agar mudah untuk di baca, maka kita dapat menggunakan cara komentar *multi-line*.

Contoh Penulisan :

Praktikkan: Buat file baru bernama “komentar2.cpp” dan ketikkan kode berikut:



```

komentar2.cpp
1 /*ini komentar multiline atau lebih satu baris
2 nama: Nur Alamsyah
3 pokok bahasan : komentar
4 belajar C++ itu mudah */
5 #include<iostream>
6 using namespace std;
7
8 int main(){
9     cout<<"Belajar C++ di FTI UNISKA";
10    return 0;
11 }

```

Fitur komentar biasanya juga digunakan untuk menandai pernyataan dari algoritma yang bermasalah, Menghilangkan sementara pernyataan dari dalam baris eksekusi kompilator agar tidak dibaca oleh kompilator. Kadang hal ini berguna untuk menghilangkan sementara pernyataan bermasalah, agar dapat diperbaiki lain waktu dan membuat program masih tetap berjalan dengan baik tanpa pernyataan itu.

FILE HEADER DAN STRUKTUR DASAR C++

File Header

Compiler akan membaca program dari atas ke bawah, jadi kita harus meletakkan file header di paling atas, atau setidaknya tepat di atas program utama.

Karena jika meletakkan file header di bawah program utama yang terjadi adalah error karena compiler tidak bisa mendeklarasikan fungsi yang terdapat pada program utama.

Struktur Dasar C++

```
1 #include <iostream>
2
3 using namespace std;
4
5 main(){
6     cout<<"Hai, Saya Belajar C++";
7 }
```

File Header

Instruksi agar compiler dapat menggunakan semua fungsi yang terdapat pada file header

Program Utama

FILE HEADER YANG PALING SERING DIGUNAKAN

A. Header stdio.h

1. printf
2. puts
3. gets
4. scanf

B. Header iostream

1. cout
2. cin
3. endl;

C. Header conio.h

1. getch
2. getche

D. Header string

1. strcpy
2. strcmp
3. strlen

Praktikan: Buat Folder baru beri nama **stdio** kemudian didalamnya **buat file** baru beri nama **printf.cpp**, ketikkan kode berikut:

Header : stdio.h

Function : printf()

Fungsi output yang digunakan untuk menampilkan pesan ke layar secara terformat

```
1 #include <stdio.h>
2
3 main(){
4     printf("Hai, saya warga negara
5         Indonesia");
6 }
```

Praktikan: Buat file baru beri nama **puts.cpp**

Header : stdio.h

Function : **puts()**

Fungsi output yang digunakan untuk menampilkan pesan yang bertipe data string

```
1 #include <stdio.h>
2
3 main(){
4     puts("Indonesia");
5 }
```

Praktikan: Buat file baru beri nama **gets.cpp**

Header : stdio.h

Function : **gets()**

Fungsi input yang secara khusus untuk masukan dengan tipe data string

```
1 #include <stdio.h>
2
3 main(){
4     char negara[15];
5     puts("Saya tinggal di ");
6     gets(negara);
7 }
```

Praktikan: Buat file baru beri nama **scanf.cpp**

Header : stdio.h

Function : **scanf()**

Fungsi input untuk menginputkan data numerik, karakter, dan string secara terformat

```
1 #include <stdio.h>
2
3 main(){
4     char negara[15];
5     printf("Saya tinggal di ");
6     scanf("%s", negara);
7 }
```

Praktikan: Buat Folder baru beri nama **iostream** kemudian didalamnya **buat file** baru dan beri nama **cout.cpp**

Header : iostream

Function : **cout**

Fungsi output yang menampilkan pesan dengan tipe data apapun ke layar

```
1 #include <iostream>
2
3 using namespace std;
4
5 main(){
6     cout<<"Hai, Saya warga negara Indonesia";
7 }
```

Praktikan: Buat file baru beri nama **cin.cpp**

Header : iostream

Function : **cin**

Fungsi input untuk masukan berupa numerik dan karakter

```
1 #include <iostream>
2
3 using namespace std;
4
5 main(){
6     char jawab[10];
7     cout<<"Ibukota Negara Indonesia adalah";
8     cin>>jawab;
9 }
```

Praktikan: Buat file baru beri nama **endl.cpp**

Header : iostream

Function : **endl**

Fungsi yang digunakan untuk perintah pindah baris

```
1 #include <iostream>
2
3 using namespace std;
4
5 main(){
6     char jawab[10];
7     cout<<"Ibukota Negara Indonesia adalah"<<endl;
8     cin>>jawab;
9 }
```

Praktikan: Buat Folder baru beri nama **conio** kemudian didalamnya **buat file** baru dan beri nama **getch.cpp**

Header : conio.h

Function : getch()

GET CHARACTER, berfungsi untuk mempause output suatu program, dan kembali running setelah kita melakukan interaksi dg memberikan inputan

```

1 #include <conio.h>
2 #include <iostream>
3
4 using namespace std;
5
6 main() {
7     char tombol;
8     cout<<"Tekan tombol apapun pada keyboard = ";
9     cin>>tombol;
10    cout<<endl;
11    cout<<"Saya menekan tombol = "<<tombol<<endl;
12    getch();
13 }
```

Praktikan: Buat file baru beri nama **getche.cpp**

Header : conio.h

Function : getche()

GET CHARACTER ECHO, perbedaannya dg **getch()** adalah inputan yang diberikan akan tampil dalam window secara langsung

```

1 #include <conio.h>
2 #include <iostream>
3
4 using namespace std;
5
6 main() {
7     char tombol;
8     cout<<"Tekan tombol apapun pada keyboard = ";
9     tombol=getche();
10    cout<<endl;
11    cout<<"Saya menekan tombol = "<<tombol<<endl;
12    getch();
13 }
```

Praktikan: Buat Folder baru beri nama **string** kemudian didalamnya **buat file** baru dan beri nama **strcpy.cpp**

Header : string

Function : strcpy()

Digunakan untuk menyalin suatu string ke variabel tujuan

```

1 #include <iostream>
2 #include <string>
3
4 using namespace std;
5
6 main(){
7     char negara[20];
8     strcpy(negara, "Indonesia");
9     cout<<"Saya warga negara "<<negara;
10 }
```

Praktikan: Buat file baru beri nama **strcmp.cpp**

Header : string

Function : strcmp()

Digunakan untuk membandingkan dua buah string

```

1 #include <iostream>
2 #include <string>
3
4 using namespace std;
5
6 main (){
7     char string1[10]; char string2[20]; int hasil;
8
9     cout<<"Masukkan kata pertama = ";cin>>string1;
10    cout<<"Masukkan kata kedua = ";cin>>string2;
11
12    hasil=strcmp(string1,string2);
13
14    if(hasil==0){
15        cout<<"Kata pertama dan kedua sama";
16    }else{
17        cout<<"kata pertama dan kedua berbeda";
18    }
19 }
```

Praktikan: Buat file baru beri nama **strlen.cpp**

Header : **string**

Function : **strlen()**

Digunakan untuk menghitung jumlah karakter yang ada dalam suatu string

```
1 #include <iostream>
2 #include <string>
3
4 using namespace std;
5
6 main (){
7     char nama[10];
8     int jumlah_karakter;
9
10    cout<<"Masukkan sebuah kata ";cin>>nama;
11    jumlah_karakter = strlen(nama);
12    cout<<"Kata "<<nama;
13    cout<<" memiliki "<<jumlah_karakter<<" karakter ";
14 }
```