

Penjelasan Code C++

```
1      #include <iostream>;  
2      using namespace std;
```

LINE 1 : #include <iostream>;

Bertujuan untuk mengimpor dari pustaka iostream yang bertujuan agar kita bisa menggunakan COUT dan CIN.

LINE 2 : using namespace std;

Code ini memudahkan kita untuk menulis COUT dan CIN dimana sebelum menggunakan code ini kita harus terlebih dahulu menulis std ::

```
3  
4  ✓  int main() {
```

LINE 4 : int main () { }

Code ini berfungsi untuk menampung fungsi utama di sebuah code. Dimana di setiap int main akan diringi symbol () dan {}. Symbol () berfungsi untuk menampung argument, sedangkan {} berfungsi untuk menampung isi utama dari int main.

```
5  
6      int n;  
7      int jumgenap = 0;  
8      int jumganjil = 0;
```

LINE 6 : int n;

LINE 7 : int jumgenap = 0;

LINE 8 : int jumganjil = 0;

3 Code diatas menggunakan data type tipe int atau juga bisa disebut interger, data type int membutuhkan yaitu variable , dan nilai contoh variable n, jumgenap, dan jumganjil, dan 0 merupakan nilai variable tersebut. interger adalah semua bilangan bulat kecuali bilangan desimal.

```
9  
10     cout << "Masukan Jumlah Data : ";  
11     cin >> n;
```

LINE 10 : cout << "Masukan Jumlah Data : ";

LINE 11 : cin >> n;

Fungsi COUT dan CIN, Fungsi cout atau juga bisa disebut character output untuk mencetak atau menampilkan sebuah data, contoh data "Masukan Jumlah Data: " .

Sedangkan cin atau juga bisa disebut character input berfungsi sebagai menerima input dari pengguna dan akan dimasukan ke data n dan kita bisa lihat di LINE 6 : int n; dimana variable n tidak mempunyai nilai sehingga ketika kita input nilai seperti 10 maka int n; akan berubah menjadi int n = 10 ; dan kita bisa lihat bahwa cin berfungsi memberi nilai ke suatu variable.

```

12
13         for (int i = 0; i < n; i++) {
14             if (i % 2 == 0) {
15                 jumgenap += i;
16             }
17             else {
18                 jumganjil += i;

```

LINE 13: for (int i = 0; i < n; i++)

LINE 14: if (i % 2 == 0)

LINE 15: jumgenap += i;

LINE 17: else { jumganjil += i; }

Perulangan for adalah salah satu struktur kendali dalam kode yang digunakan untuk mengulang suatu blok kode. Kode tersebut memiliki simbol () dan {}. Simbol () digunakan untuk menampung initialization, condition, dan increment/decrement. Sedangkan simbol {} digunakan untuk menampung beberapa pernyataan yang akan dieksekusi berulang kali selama kondisi pada perulangan for bernilai true.

Kondisi if adalah bagian dari kontrol aliran yang memungkinkan program untuk mengeksekusi blok kode tertentu jika suatu kondisi terpenuhi, dalam hal ini adalah $i \% 2 == 0$. Kondisi tersebut dapat dibaca sebagai "jika i modulo 2 sama dengan 0." Nilai i akan terus bertambah dengan satu karena adanya pernyataan i++ dalam perulangan for. Saat $i \% 2 == 0$, kondisi akan bernilai true. Jika tidak, kondisi akan bernilai false.

Ketika kondisi bernilai true, kode akan dilanjutkan ke pernyataan jumgenap += i; yang akan menambah nilai jumgenap dengan nilai i.

Sebaliknya, ketika kondisi bernilai false, kode akan dilanjutkan ke blok else dengan pernyataan jumganjil += i; yang akan menambah nilai jumganjil dengan nilai i.

Proses ini akan terus berlangsung hingga nilai i kurang satu dari nilai n, karena adanya kondisi $i < n$ pada perulangan for.

```

24         cout << "Jumlah total bilangan ganjil : " << jumganjil << "\n";
25         cout << "Jumlah total bilangan genap : " << jumgenap << "\n";

```

LINE 24 : cout << "Jumlah total bilangan ganjil : " << jumganjil << "\n";

LINE 25 : cout << "Jumlah total bilangan genap : " << jumgenap << "\n"

Kedua code ini menggunakan cout yang berfungsi untuk menampilkan sebuah data dan data tersebut "Jumlah total bilangan ganjil : " << jumganjil << "\n"; dimana "Jumlah total bilangan ganjil" adalah sebuah teks sedangkan jumganjil adalah variable dan fungsi "\n" adalah sebagai tanda untuk mengakhiri baris.

