

Rangkuman Materi C# Fundamental Pertemuan 3

Operator adalah sebuah simbol yang digunakan untuk melakukan operasi tertentu.

1. Operator Aritmatika

Operator aritmatika adalah operator untuk melakukan operasi aritmatika seperti kali, bagi, tambah, kurang.

Operator ini terdiri dari:

Nama Operator	Simbol
Penjumlahan	+
Pengurangan	-
Perkalian	*
Pembagian	/
Sisa Bagi	%
Increment	++
Decrement	--

2. Operator Penugasan

Operator penugasan (Assignment Operator) merupakan operator untuk memberikan tugas pada variabel. Biasanya digunakan untuk mengisi nilai.

Operator Penugasan terdiri dari:

Nama Operator	Sombol
Pengisian Nilai	=
Pengisian dan Penambahan	+=
Pengisian dan Pengurangan	-=
Pengisian dan Perkalian	*=
Pengisian dan Pembagian	/=
Pengisian dan Sisa bagi	%=
Pengisian dan shift left	<<=
Pengisian dan shift right	>>=
Pengisian dan bitwise AND	&=
Pengisian dan bitwise OR	=
Pengisian dan bitwise XOR	^=

3. Operator Increment dan Decrement

Increment adalah penambahan nilai +1 sedangkan Decrement adalah pengurangan nilai -1. Biasanya, kita melakukannya seperti ini:

```
int mangga = 2;  
mangga = mangga + 1; // increment
```

Nah, dengan operator Increment/Decrement.. kita bisa buat lebih singkat.

```
int mangga = 2;  
mangga++; // increment
```

Operator increment dan decrement dapat ditulis di depan maupun belakang operan/variabel. Penulisan di depan disebut pre-increment atau pra-increment, sedangkan penulisan di belakang disebut post-increment atau pasca-increment.

```
mangga++; // post-increment
++apel; // pre-increment
```

4. Operator Perbandingan

Operator pembanding adalah operator untuk memabndingkan dua buah nilai. Operator ini juga dikenal dengan operator relasi. Nilai yang dihasilkan dari operasi pembanding akan berupa boolean True dan False.

Operator pembanding terdiri dari:

Nama Operator	Simbol
Lebih Besar	>
Lebih Kecil	<
Sama Dengan	==
Tidak Sama dengan	!=
Lebih Besar Sama dengan	>=
Lebih Kecil Sama dengan	<=

5. Operator Pembanding

Operator perbandingan merupakan operator yang mengembalikan nilai boolean (true / false). Nilai yang dihasilkan dari operator perbandingan adalah True dan False. Dalam bahasa pemrograman C# terdapat beberapa operator perbandingan yang dapat digunakan yang dapat dilihat pada tabel di bawah ini.

Operator	Keterangan	Contoh
==	Sama dengan	a == b
!=	Tidak sama dengan	a != b
<	Lebih kecil	a < b
>	Lebih besar	a > b
<=	Lebih kecil atau sama dengan	a <= b
>=	Lebih besar atau sama dengan	a >= b

6. Operator Logika

Digunakan untuk melakukan operasi logika pada nilai-nilai boolean dan menghasilkan output yang juga Boolean true dan false.

Nama Operator	Simbol
Logika AND	&&
Logika OR	
Negasi/kebalikan	!

Namun perlu diingat: Operan harus bertipe boolean.. Karena jika tidak, maka akan terjadi error. Operator Logika digunakan untuk membuat operasi logika.

Contoh:

- Pernyataan 1: Mangga rasanya manis

- Pernyataan 2: Jeruk rasanya manis

Jika kita coba buktikan dengan tabel kebenaran logika, maka bisa kita buat seperti ini.

Mangga dan Jeruk:

Pernyataan 1	Pernyataan 2	Hasil Logika AND
true	true	true
true	false	false
false	true	false
false	false	false

Mangga atau Jeruk:

Pernyataan 1	Pernyataan 2	Hasil Logika OR
true	true	true
true	false	true
false	true	true
false	false	false

Bukan Mangga:

Pernyataan 1	Hasil Logika NOT
true	false
false	true

7. Array

- Array dalam C# digunakan untuk menyimpan koleksi elemen dengan tipe data yang sama dalam urutan terindeks.
- Indeks array dimulai dari 0, yang berarti elemen pertama memiliki indeks 0, elemen kedua memiliki indeks 1, dan seterusnya.
- Array dapat memiliki dimensi tunggal, 2 dimensi, dan multidimensi yang lain.
- Panjang setiap dimensi ditetapkan saat instance array dibuat.
- Kita bisa memanggil elemen array berdasar indexnya. yang nantinya bisa menggunakan perulangan (materi selanjutnya).
- Cara mendeklarasikan array adalah sebagai berikut
 - tipeData[] namaVariable; atau tipeData[] namaVariable = new int[5];
 - Untuk memasukkan nilai ke dalamnya, kita dapat menempatkan nilai dalam daftar yang dipisahkan dengan koma, di dalam kurung kurawal
 - tipeData[] namaVariable = {"Volvo", "BMW", "Ford", "Mazda"};

8. List

List merupakan sebuah objek yang dapat menampung berbagai variabel dengan urutan tertentu. List memiliki ukuran yang dinamis, sesuai dengan kemampuan maksimal memory computer.

9. Pengkondisian

- If- Else

If-else adalah salah satu pengendalian alur program yang digunakan untuk mengambil keputusan berdasarkan kondisi tertentu. Ini memungkinkan program untuk menjalankan blok kode tertentu jika kondisi tertentu terpenuhi, dan menjalankan blok kode lain jika kondisi tersebut tidak terpenuhi. Blok kode dalam else if statement akan dijalankan jika kondisi yang diuji dalam **if** sebelumnya adalah **salah**, dan kondisi dalam **else if** adalah **benar**.

- Switch

Switch adalah salah satu struktur kontrol yang digunakan dalam pemrograman untuk membuat keputusan berdasarkan nilai ekspresi yang berbeda. Ini memungkinkan program untuk menjalankan blok kode tertentu berdasarkan nilai ekspresi yang cocok dengan salah satu dari beberapa kasus yang telah ditentukan.

Case adalah nilai-nilai atau ekspresi yang dibandingkan dengan ekspresi switch. case default adalah case yang dijalankan jika tidak ada case yang cocok dengan ekspresi switch. Ini bersifat opsional dan tidak harus ada dalam setiap struktur switch.

10. Ternary Operator

Operator ternary adalah cara singkat untuk mengevaluasi ekspresi berdasarkan kondisi tertentu. Sama seperti if else namun ini versi singkatnya. penulisan ternary operator adalah sebagai berikut

kondisi ? nilaiJikaBenar : nilaiJikaSalah;

11. Perulangan

Perulangan atau looping adalah konsep yang sangat penting dalam pemrograman yang digunakan untuk mengulangi serangkaian perintah atau tugas tertentu berulang kali selama kondisi tertentu terpenuhi.

12. Perulangan-for

Dari pada kita menuliskan blok kode yang berulang - ulang secara manual, alangkah baiknya kita menggunakan perulangan agar lebih efisien dan cepat dalam penulisan blok kode. penulisan perulangan for adalah sebagai berikut:

```
for(int = 0 ; i <5 ; i++){  
    //blok kode
```

```
}
```

- Perulangan for memiliki 3 bagian yang perlu diperhatikan pertama (int i = 0) adalah sebuah “inisialisasi” yang menetapkan nilai awalnya ke 0.
- kedua (i < 5) adalah “kondisi” dimana perulangan akan terus berjalan hingga kondisi terpenuhi
- ketiga (i++) adalah “iterator” menentukan bagaimana memodifikasi variable perulangan setelah mengeksekusi blok setelah pernyataan for. Di sini, ini menentukan bahwa indeks harus ditambah dengan 1 setiap kali blok dieksekusi.

13. Perulangan While

Pernyataan while adalah pernyataan yang berguna untuk memproses suatu pernyataan atau memproses pernyataan beberapa kali. Pernyataan atau aksi akan di ulang jika kondisi bernilai benar dan jika salah maka keluar dari blok perulangan (loop).

Penulisan perulangan while dalam C# adalah sebagai berikut,

```
int i = 0;

while( i < 5){
//kode program i++;

}
```

- int i adalah sebuah inisialisasi
- (i < 5) adalah kondisinya
- dan i++ adalah iterasi / iteratornya

14. Perulangan Do While

Perulangan akan dilakukan minimal 1x terlebih dahulu, kemudian baru dilakukan pengecekan terhadap kondisi, jika kondisi benar maka perulangan masih akan tetap dilakukan. Perulangan dengan do...while() akan dilakukan sampai kondisi false.

15. Perulangan Foreach

Perulangan Foreach adalah alat yang ampuh untuk melakukan iterasi melalui koleksi, seperti array atau daftar. penulisan foreach dalam c# adalah sebagai berikut,

```
foreach(var a in angka){
```

```
//blok kode
```

```
}
```

16. Break

Break adalah pernyataan yang biasanya digunakan dalam loop (for, while, atau do-while) atau dalam struktur switch untuk menghentikan eksekusi loop atau keluar dari struktur switch lebih awal.

Jika terdapat kode program setelah break, maka kode tersebut tidak akan dikerjakan / akan diabaikan. **Pernyataan break digunakan dengan pernyataan pengambilan keputusan seperti if...else.**

17. Continue

Continue adalah pernyataan yang digunakan dalam loop untuk melanjutkan ke iterasi berikutnya tanpa menjalankan kode di bawahnya dalam iterasi saat ini.

Ini berguna ketika Anda ingin mengabaikan beberapa iterasi loop berdasarkan kondisi tertentu.

18. Function

- Fungsi (juga dikenal sebagai metode dalam konteks C#) adalah blok kode yang digunakan untuk menjalankan tugas tertentu.
- Fungsi memungkinkan untuk mengelompokkan sejumlah pernyataan ke dalam satu unit yang dapat digunakan kembali.
- Fungsi memiliki nama dan dapat menerima argumen (data yang diteruskan ke dalamnya) dan mengembalikan nilai (hasil dari operasinya).
- Penulisan fungsi dalam c# adalah sebagai berikut

```
[modifiers] returnType namaFungsi([parameter1, parameter2, .....])
```

```
{
```

```
//blok kode
```

```
return nilaiKembali; (opsional, tergantung return type  
fungsinya)
```

```
}
```