

PART 16

PENGERTIAN DAN CARA PENGGUNAAN TIPE DATA BOOLEAN PASCAL



-
- Apabila anda baru pertama kali mempelajari algoritma dan pemrograman, **tipe data boolean** mungkin terasa asing, apalagi jika dibandingkan dengan tipe data 'biasa' yang telah kita pelajari seperti angka (**integer** dan **real**) atau huruf (**char** dan **string**).

PENGERTIAN TIPE DATA BOOLEAN PASCAL

Tipe data **boolean** sebenarnya sangat sederhana. Tipe data ini hanya bisa diisi dengan salah satu dari 2 nilai: **TRUE** atau **FALSE**. Tipe data **boolean** banyak digunakan untuk percabangan kode program, atau untuk memutuskan apa yang mesti dijalankan ketika sebuah kondisi terjadi.

Sebagai contoh, kita bisa membuat kode program untuk menentukan apakah sebuah angka genap atau ganjil berdasarkan input dari pengguna. Untuk keperluan ini kita harus mengecek terlebih dahulu apakah angka itu bisa dibagi 2 (untuk angka genap), atau tidak bisa dibagi 2 (untuk angka ganjil). Tipe data boolean bisa digunakan untuk menampung kondisi seperti ini, benar atau salah (*True* atau *False*).

Contoh penggunaan tipe data boolean akan lebih jelas jika anda telah mempelajari kondisi percabangan program seperti IF (yang akan saya jelaskan pada tutorial terpisah). Untuk sementara anda hanya perlu memahami bahwa tipe data boolean adalah tipe data yang hanya memiliki 2 isi: **TRUE** atau **FALSE**.

Cara Penggunaan Tipe Data Boolean Pascal

Langsung saja kita lihat contoh program untuk mendefenisikan dan mengisi variabel dengan tipe data **boolean** di pascal:

```
1  program tipe_boolean;  
2  uses crt;  
3  var  
4      a:boolean;  
5  begin  
6      clrscr;  
7  
8      a:= true;  
9      writeln('a: ',a);  
10  
11     a:= false;  
12     writeln('a: ',a);  
13  
14     a:= not true;  
15     writeln('a: ',a);  
16  
17     a:= not false;  
18     writeln('a: ',a);  
19  
20     readln;  
21 end.
```



Dalam contoh diatas, saya membuat variabel **a** sebagai **boolean**, kemudian mengisinya dengan **true** dan **false**. Pada contoh ketiga dan keempat, saya menggunakan perintah **not** untuk membalik nilai boolean. Perintah '**not true**' akan menghasilkan **false**, sedangkan '**not false**' akan menghasilkan **true**.

Perhatikan bahwa nilai untuk boolean ini (**true** atau **false**) harus ditulis **tanpa tanda kutip**. Jika anda menambahkan tanda kutip, ini menjadi **string**. Dengan kata lain '**true**' tidak sama dengan **true** (perhatikan tanda kutipnya).

Selain menulis langsung nilai boolean, kita juga bisa menginput hasilnya dari sebuah operasi perbandingan. Berikut contohnya:

```
1  program tipe_boolean;
2  uses crt;
3  var
4      a:boolean;
5  begin
6      clrscr;
7
8      a:= 1=1;
9      writeln('a: ',a);
10
11     a:= 9<8;
12     writeln('a: ',a);
13
14     a:= 999<>998;
15     writeln('a: ',a);
16
17     a:= 'duniailkom' = 'duniailkom.com';
18     writeln('a: ',a);
19
20     readln;
21 end.
```



Salah satu konsep yang harus anda pahami di dalam pemrograman adalah penggunaan tanda '='. Didalam pascal, tanda ini digunakan untuk operasi perbandingan, bukan pengisian ke dalam sebuah variabel (operasi *assignment*).

Dengan kata lain, "**a:= 1=1**" berarti kita mengisi nilai variabel **a** dengan hasil dari perbandingan "**1=1**". Jadi apakah **1=1**? Yup **benar**, yang di dalam bahasa inggris menjadi: **true**. Oleh karena itulah variabel **a** akan diisi dengan nilai boolean **true**.

Pada contoh kedua, apakah "**9<8**"? **salah!**, karena 9 bukan lebih kecil dari delapan, sehingga hasilnya: **false**.

Berikutnya, apakah "**999<>998**"? Tanda "<>" di dalam pascal bisa dibaca sebagai "*tidak sama dengan*", sehingga hasilnya adalah: **true**.

Contoh terakhir, apakah '**duniailkom**' = '**duniailkom.com**'? **salah!**, karena untuk '*benar*', kedua string harus mengandung karakter yang sama persis. Sehingga hasilnya adalah **false**.

Tipe data **boolean** yang dipelajari disini akan lebih jelas penggunaannya jika anda telah mempelajari struktur logika percabangan kode program seperti **IF**. Saya akan menjelaskan struktur alur kode program ini setelah pembahasan mengenai tipe data dan operator pascal.