

DASAR PEMROGRAMAN

File

DESKRIPSI

- ✘ **File** adalah kumpulan byte-byte yang disimpan dalam media penyimpanan.
 - ✘ File merupakan komponen yang bertipe data sama, yang jumlahnya tidak tentu, yang dapat ditambah atau dikurangi jika dibutuhkan
- ✘ **Pointer file** adalah bagian yang menunjuk ke komponen file yang akan diakses (dibaca atau direkam) untuk keperluan pengaksesan file.

✘ Selain kita akan mempelajari tentang bagaimana membuat sebuah file atau menambahkan isi suatu file, kita dapat pula melakukan manipulasi file, yaitu :

1. Menggunakan parameter
2. Mengenai Atribut File
3. Menghapus file
4. Mengubah nama file
5. Mengenai tanggal dan waktu file
6. Mencari file
7. Mengecek keberadaan file
8. Memberikan directory file

-
- ✘ File pada Pascal dikenal dalam 3 jenis, yaitu :
 - + File Text
 - + File Bertipe
 - + File Tidak Bertipe

FILE TEXT

× Karakteristik

- + Berisi data karakter ASCII
- + Tiap record boleh memiliki panjang yang bervariasi
- + Setiap record diakhiri tanda EOL (end of Line)
- + Hanya dapat diakses secara sequensial (berurutan)
- + Isi filenya dapat dilihat oleh perintah dos type atau editor text

MEMBUAT FILE TEXT

- ✘ Deklarasi Variable File

```
Var NmVar : TEXT;
```

Dengan :

NmVar : Nama variabel file text

- ✘ Menghubungkan variabel file dengan nama file

```
Assign (NmVar, namaFile);
```

Dengan :

NmVar : Nama variabel file text

namaFile : Nama file dalam bentuk string, format 8:3 penamaan dos, ditulis dalam bentuk string

- ✘ Membuat file text aktif

Rewrite(NmVar);

NmVar : Nama variabel file text yang sudah di-assign

- ✘ Menulis ke dalam file text

Write / writeln (NmVar, data item1, data item 2, ...);

NmVar : Nama variabel file text

Data item : text / string yang akan dituliskan, atau bisa juga berupa isi suatu variabel

- ✘ Menutup file

Close (NmVar);

NmVar : Nama variabel file text

CONTOH

Program membuat_file_text_namafile_HALLO_TXT;

Uses wincrt;

Var

var_teks:TEXT;

Begin

Clrscr;

Assign(var_teks,'Hallo.txt');

Rewrite(var_teks);

Writeln(var_teks,'Hallo... ini program pertamaku!');

Writeln(var_teks,'Contoh file teks');

Writeln(var_teks,'-----');

Close(var_teks);

End.

MEMBACA FILE TEXT

- ✘ Mendeklarasikan variabel file

```
Var NmVar:TEXT;
```

- ✘ Menghubungkan variabel file dengan nama file

```
Assign (NmVar, namaFile)
```

- ✘ Membuka file

```
reset (NmVar);
```

- ✘ Membaca isi file dan menampilkannya di layar

```
While not eof (NmVar) do  
  Begin  
    Read / readln ( NmVar, data item 1, data item 2, ... );  
    Write / writeln ( data item1, data item 2, ... );  
  End;
```

- ✘ Menutup file

```
Close (NmVar);
```

CONTOH

```
Program membaca_file_HALLO_TXT;
```

```
Uses wincrt;
```

```
Var
```

```
    var_teks:TEXT;
```

```
    Pesan:string;
```

```
Begin
```

```
    Clrscr;
```

```
    Assign(var_teks,'Hallo.txt');
```

```
    Reset(var_teks);
```

```
    While not eof (var_teks) do
```

```
    Begin
```

```
        Readln ( var_teks, pesan );
```

```
        Writeln (pesan);
```

```
    End;
```

```
    Close(var_teks);
```

```
End.
```

FILE BERTIPE

× Karakteristik

- + Berisi data format biner, ukurannya lebih kecil dari file teks.
- + Tiap record memiliki tipe dan panjang yang sama. Bisa saja memiliki berbagai tipe asalkan dikelompokkan dalam RECORD.
- + Dapat diakses secara random, elemen-elemennya bisa dibaca secara acak yang ke berapa saja

MEMBUAT FILE BERTIPE

1. Mendeklarasikan variabel file

```
Var NmVar:File of TypeVariabel;
```

Dengan :

```
NmVar          : Nama variabel file bertipe
```

```
TypeVariabel  : Char, variabel tipe RECORD,  
                variabel tipe array, real,  
                variabel array tipe record.
```

```
                Untuk satu file satu tipe elemen.
```

Contoh :

```
Type DaftarBarang = Array [1..100] of  
integer;
```

DataKonsumen = RECORD

Nama :string[15];

Alamat :string[30];

Kode :1..3;

DaftarKonsumen= Array [1..100] of Datakonsumen ;

Var

FileBarang : File of Daftarbarang;

FileJumlah : File of integer;

FileData : File of DataKonsumen;

FileKode : File of Char;

2. Menghubungkan variabel file dengan nama file

```
Assign (NmVar, nama file);
```

Dengan :

NmVar : Nama variabel file bertipe

nama file : Nama file dalam bentuk string,
format 8:3 penamaan dos, ditulis
dalam bentuk string.

3. Membuat /membuka file bertipe

`Rewrite (NmVar); => untuk membuat`

`Reset (NmVar); => untuk membuka`

4. Menulis / membaca file Bertipe

Write (NmVar, data item1, data 2, ...)

=> untuk menulis

Read (NmVar, data item1, data item 2,

...)=>untuk membaca

5. Menutup file

`Close (NmVar);`

FILE TIDAK BERTIPE

× Karakteristik

- + File yang mengakses langsung ke media penyimpanan tanpa adanya pengenalan record dan sebagainya.
- + Digunakan untuk tugas-tugas yang berhubungan dengan file biner yang dapat diproses tanpa mengenal jenis recordnya.

MEMBUAT FILE TIDAK BERTIPE

1. Mendeklarasikan variabel file

```
Var NmVar:FILE;
```

2. Menghubungkan variabel file dengan nama file

```
Assign (NmVar, nama file);
```

Dengan :

NmVar : Nama variabel file bertipe

nama file : Nama file dalam bentuk string,
format 8:3 penamaan dos,
ditulis dalam bentuk string.

3. Membuka file tidak bertipe

`Rewrite (NmVar [, brec]);` => untuk membuat
`Reset (NmVar [, brec]);` => untuk membuka

Dengan :

`Brec` : Menunjukkan besar file dalam byte, opsional, boleh ditulis, boleh tidak, dan besarnya kita tentukan sendiri. Defaultnya 128 Byte.

4. MENULIS / MEMBACA FILE TIDAK BERTIPE

```
Blockwrite (NmVar, Buffer, jumlah,  
            [jumptulis]); => untuk menulis
```

Dengan :

`Buffer` : daerah penyimpanan data yang akan ditulis ke dalam file.

Buffer dapat berupa suatu variabel dengan tipe apa saja sesuai dengan ukuran data yang akan ditulis di file.

`Jumlah` : jumlah data yang akan ditulis ke file dalam ukuran byte

`Jumptulis` : suatu parameter yang tidak tetap yang boleh digunakan dan boleh juga tidak digunakan, bila digunakan akan berisi jumlah byte yang dapat ditulis ke file.

4. MENULIS / MEMBACA FILE TIDAK BERTIPE

`Read (NmVar, data item1, data item 2) =>` untuk membaca

Dengan :

`Buffer` : daerah penyimpanan yang tipe variabelnya disesuaikan dengan jumlah data yang dibaca.

`Jumlah` : jumlah byte yang akan dibaca dari file, dapat merupakan suatu variabel dengan tipe word. Jumlah harus sama dengan besar buffer yang diberikan dan tidak boleh lebih dari 64 Kilobyte.

`Jumbaca` : merupakan variabel yang berisi laporan jumlah byte yang dapat dibaca dari file.

REVIEW

- ✘ Buat aplikasi yang menggunakan file sebagai media penyimpanan dan akses data!

TERIMAKASIH