

**MODUL 7****TIPE DATA TERSTRUKTUR**

Type Data Terstruktur di dalam bahasa Pascal terdiri dari :

- Array
- Record
- Set
- File

**7.1 TIPE DATA ARRAY / LARIK**

Larik yaitu deretan data yang mempunyai tipe data sejenis.

Ex : Daftar Nomor Telpon, Tabel Pajak dll.

Tiap rinci data disebut komponen/Elemen Larik.

Tiap larik bisa dioperasikan menggunakan pemilih elemen larik yang disebut INDEX atau Subskrib. Dan Index ini diletakkan dalam tanda '[ ]', sesudah nama larik.

**7.2 DEKLARASI ARRAY**

Deklarasi perubah selalu diawali dengan kata VAR Dalam deklarasi perubah biasa, kita hanya perlu menentukan type data perubah tersebut misalnya: real, integer, boolean atau char, sedangkan pada deklarasi larik (array), kita perlu mendefinisikan type larik, juga harus menentukan banyaknya index yang akan mengisi larik tersebut.

Syntax :

VAR nmVariabel : ARRAY [index] OF typeData
--

Keterangan :

nmVariabel : Nama larik yang dideklarasikan  
index : Batasan/Jumlah Index (cacah elemen)  
typeData : Tipe data dari Larik

Larik dapat bertipe data sederhana byte, word, integer, real, boolean, char atau string dan tipe data skalar atau subrange. Index dari larik menunjukkan maksimum banyaknya elemen-elemen dari larik. Index larik ini dapat berupa tipe subrange atau skalar apapun (integer, byte, word, char ) kecuali tipe real.

Suatu larik tidak hanya dapat berupa suatu variabel yang dideklarasikan di bagian variabel, tetapi juga dapat berupa uatu konstanta yang dideklarasikan di bagian deklarasi konstanta.

**Contoh 1 :**

Var  
Nama : Array[1..10] of String[20] ;

```
I      : Byte ;
Begin
  Clrscr
  {Bagian Input Data}
  For I := 1 To 10 Do
  Begin
    Writeln('Inputkan Nama Ke ', I , ':' ) ; Readln( Nama[I] ) ;
  End;
  {Bagian Menampilkan Data yang telah diinputkan }
  For I := 1 To 10 Do
    Writeln('Nama Ke ', I , ':', Nama[I] ) ;
  Readln;
End.
Contoh 7.1 : Program dengan Penggunaan Array.
```

**Contoh 2 :**

```
Var
  Nama  : Array[1..30] of String[20]  ;
  Gaji  : Array[1..30] of Longint      ;
  i,j   : Byte ;
Begin
  Clrscr ;
  {Bagian Input Data}
  Writeln('Inputkan Jumlah Data : ') ; Readln( J ) ;
  For I := 1 To J Do
  Begin
    Writeln('Inputkan Nama Ke ', I , ':' ) ; Readln( Nama[I] ) ;
    Writeln('Inputkan Gaji Ke ', I , ':' ) ; Readln( Gaji[I] ) ;
  End;
  Clrscr ;
  {Bagian Menampilkan Data yang telah diinputkan }
  GotoXy(10, 4) ; Write('-----') ;
  GotoXy(10, 4) ; Write('No.      Nama                      Gaji');
  GotoXy(10, 6) ; Write('-----') ;
  For I := 1 To J Do
  Begin
    GotoXy(10, 6+I) ; Writel( I ) ;
    GotoXy(14, 6+I) ; Write( Nama[I] ) ;
    GotoXy(34, 6+I) ; Write( Gaji[I] ) ;
  End
  GotoX(10, J+7) ; Writeln('-----')
```

Readln

End.

Contoh 7.2 : Program untuk menginputkan Nama & Gaji menggunakan Array.

### 7.3 TIPE DATA RECORD (Rekaman)

Seperti halnya array(larik), record juga punya elemen, di sini elemennya disebut "Field". Tiap elemen bisa punya type data yang berbeda. Banyaknya field dapat bervariasi atau bisa juga tetap.

Record yang punya field yang bervariasi disebut 'Variant Record'. Deklarasi type data Record dapat ditetapkan di bagian deklarasi Var atau Type.

Syntax :

```
TYPE
    nmPengenal = RECORD
        Field1 : type1;
        Field2 : type2;
        :
        fieldn : typen;
    END;
VAR
    nmRec : nmPengenal ;
```

Keterangan :

nmPengenal : Nama Dari Record ;

Field1, Field2, Fieldn : Nama dari Variabel/Field yang terdapat dalam Record

Type1, Type2, Typen : Type Dari tiap-tiap Field yang dideklarasikan

nmRec : Nama Dari Record yang di deklarasikan

Contoh :

```
TYPE
    Mahasiswa = RECORD
        NIM : String[8] ;
        Nama : String[20] ;
        Alamat : String[30] ;
        Sex : Char ;
        Umur : Byte ;
    END;
VAR
    RecMhs : Mahasiswa ;
```

### MEMBACA & MENULIS FIELD

Dapat dilaksanakan pada seluruh record atau sebagian record.

Syntax : nmRec.Field1 ;

Tanda titik harus ditulis

Notasi ini disebut Penandaan field ('field designator').

Berbeda dengan statement pemberian (assignment), maka untuk membaca dan menulis record selalu dilakukan menurut Field yang ada, tidak dilakukan secara keseluruhan record.

Contoh : Membaca    `Readln(RecMhs.NIM );`  
                              `Readln(RecMhs>Nama );`  
          Menulis        `Writeln(RecMhs.NIM );`  
                              `Writeln(RecMhs>Nama );`

### **Contoh 3 :**

```
TYPE
    Mahasiswa = RECORD
        NIM    : String[8]    ;
        Nama   : String[20]   ;
        Umur   : Byte         ;
    END;

Var
    RecMhs : Mahasiswa      ;

Begin
    Clrscr ;
    {Bagian Input Data}
    Writeln('NIM Anda    :'); readln( RecMhs.NIM ) ;
    Writeln('Nama       :'); readln( RecMhs>Nama ) ;
    Writeln('Umur       :'); readln( RecMhs.Umur ) ;

    {Bagian Menampilkan Data yang telah diinputkan }
    Writeln('NIM Anda    :', RecMhs.NIM ) ;
    Writeln('Nama       :', RecMhs>Nama ) ;
    Writeln('Umur       :', RecMhs.Umur ) ;
    Readln;

End.
```

Contoh 7.3 : Program dengan Penggunaan Array.

## **7.4 STATEMENT WITH**

Digunakan untuk mengurangi penulisan yang berulang-ulang untuk nama recordnya. Maka dengan menggunakan statement WITH hanya cukup sekali saja menyebutkan nama recordnya.

Syntax : **WITH** nmRecord **DO**

{langsung dimasukkan/dicetak fieldnya}

**Contoh 4 :**

```
TYPE
    Mahasiswa = RECORD
        NIM      : String[8]    ;
        Nama     : String[20]   ;
        Umur     : Byte         ;
    END;

Var
    RecMhs : Mahasiswa    ;

Begin
    Clrscr ;
    With RecMhs Do
        Begin
            {Bagian Input Data}
            Writeln('NIM Anda      :'); readln(NIM ) ;
            Writeln('Nama :'); readln(Nama ) ;
            Writeln('Umur :'); readln(Umur ) ;

            {Bagian Menampilkan Data yang telah diinputkan }
            Writeln('NIM Anda      :', NIM ) ;
            Writeln('Nama          :', Nama ) ;
            Writeln('Umur          :', Umur ) ;

        End;
    Readln;

End.
```

Contoh 7.4 : Penggunaan Statemen WITH pada RECORD.

**LATIHAN**

1. Nama Program : **Latih71.Pas**

Program Latih71; {Program input data pada record}

Uses crt;

Type

Barang = record

Kode : String[5] ;

Nama : String[20] ;

Jumlah : Integer ;

Harsat : Longint ;

End;

Var

RecBrg : Array [1..40] of Barang ;

Lagi : Char;

```
I, J      : Integer;
Begin
  Clrscr ; I:=0 ; J := 0 ;
  { Untuk Menginputkan Data Barang}
  Repeat
    Inc(I)
    Write('Kode Barang   :'); readln(RecBrg[I].Kode);
    Write('Nama           :'); readln(RecBrg[I].Nama);
    Write('Jumlah         :'); readln(RecBrg[I].Jumlah);
    Write('Harga satuan   :'); readln(RecBrg[I].Harsat);
    Write('Ada lagi [Y/T]   : '); readln(lagi);
  Until lagi<>'Y' or lagi<>'y'      { bisa ditulis Until Upcase(lagi)='Y' }
  { Menampilkan Data Barang }
  Clrscr ;
  GotoXy(10, 4) ; Write('-----');
  GotoXy(10, 5) ; Write('No. Kode      Nama                Jumlah      Harga');
  GotoXy(10, 6) ; Write('-----');
  For J := 1 To I Do
    Begin
      GotoXy(10, 6+J) ; Writel( I ) ;
      GotoXy(13, 6+J) ; Write(RecBrg[J].Kode ) ;
      GotoXy(19, 6+J) ; Write(RecBrg[J].Nama ) ;
      GotoXy(39, 6+J) ; Write(RecBrg[J].Jumlah ) ;
      GotoXy(39, 6+J) ; Write(RecBrg[J].Harsat) ;
    End
    GotoX(10, J+7) ; Wrt('-----');
  Readln;
End.
```

2. Nama Program : **Latih72.Pas**

```
USES CRT ;
VAR
  NIP : Array[1..20] of string [8] ;
  NM  : Array[1..20] of string [20];
  GOL : Array[1..20] of string [3] ;
  i, j : byte ;
  Gaji, TGaji : Longint ;
  lg : Char ;
BEGIN
  i := 1 ;
  REPEAT
```

```
Clrscr;
Gotoxy(20,2); Write(' INPUT DATA GAJI PEGAWAI');
Gotoxy(10,3); Write(' _____');
Gotoxy(10,4); Write(' NIP      :');
Gotoxy(10,5); Write(' NAMA      :');
Gotoxy(10,6); Write(' GOL [I,II,III] :');
Gotoxy(10,7); Write(' _____');
Gotoxy(10,8); Write(' Isikan Data Lagi[Y/T] :');
Gotoxy(27,4); Readln( NIP[ i ] );
Gotoxy(27,5); Readln( NM[ i ] );
Gotoxy(27,6); Readln( GOL[ i ] );
Gotoxy(33,8); Readln( Lg );
inc( i );
```

**UNTIL Uppcase(lg)='T';**

```
Clrscr;
Gotoxy(3,3);write('LAPORAN          DATA          GAJI          PEGAWAI');
Gotoxy(3,4);write('_____');
Gotoxy(3,5);write(' No      NIP      NAMA          GOL      GAJI Rp ');
Gotoxy(3,6);write('_____');
```

**FOR j:=1 TO i -1 DO**

**BEGIN**

```
    If Gol [ j ]='I' then Gaji := 750000
    Else If Gol [ j ]='II' then Gaji := 1000000
    Else If Gol [ j ]='III'then Gaji := 1500000
    Else Gaji :=0;
    Gotoxy(5 ,6+j); Write( j:2, ' ',NIP[ j ]: 8, ' ', NM[ j ] );
    Gotoxy(44,6+j); Write( Gol [ j ] );
    Gotoxy(49,6+j); Write( Gaji:8 );
    TGaji := TGaji + Gaji ;
```

**END;**

```
Gotoxy(3,7+j);
Write('_____');
Gotoxy(3,8+j); Write(' TOTAL GAJI          RP.', tgaji:10);
```

END.