

MODUL 11 RECORD

1. Deskripsi Singkat

Pada modul 11 ini akan dibahas penggunaan record.

2. Tujuan Praktikum

Setelah praktikum pada modul 11 ini diharapkan:

1. Mahasiswa memahami dan menguasai struktur data record
2. Mahasiswa dapat menggunakan struktur data record dengan terampil pada program.

3. Material Praktikum

Kegiatan pada modul 11 ini memerlukan material berupa program Pascal

4. Kegiatan Praktikum

A. Deklarasi Record

Berbeda dengan array yang berfungsi untuk menyimpan sejumlah data dengan tipe yang sama, record merupakan sebuah variabel untuk menyimpan data dengan tipe yang bisa berbeda.

Record dapat juga dideklarasikan pada bagian type untuk selanjutnya kita gunakan pada bagian var.

```
Type
Identifier = Record
    Namafield-1 : Type
    Namafield-2 : Type
    .....
    Namafield-N : Type
End

Var
    Identifier : Record
    Namafield-1 : Type
    Namafield-2 : Type
    .....
    Namafield-N : Type
End
```

Contoh:

```
type
    data_mahasiswa = record
        nim : string[10];
        nama : string[25];
        alamat : string[20];
        nilai : longint;
    end;
var
    mahasiswa : data_mahasiswa;
```

atau bisa juga langsung kita deklarasikan pada bagian var

```
var
    mahasiswa : record
        nim : string[10];
        nama : string[25];
        alamat : string[20];
        nilai : longint;
    end;
```

Untuk mengakses elemen record kita menggunakan nama identifiernya diikuti dengan tanda titik (.). perhatikan contoh berikut ini. Simpan program dengan nama **Praktikum11A.pas** lalu jalankan program. Apakah output yang dihasilkan program?

```
type
    struktur_mahasiswa = record
        nim : string[10];
        nama : string[25];
        alamat : string[20];
        nilai : longint;
    end;
var
    mahasiswa : struktur_mahasiswa;
begin
```

```

mahasiswa.nim := '235121';
mahasiswa.nama := 'Upin Ipin';
mahasiswa.alamat:= 'Jl. Otista no 64c';
mahasiswa.nilai:=98;

writeln('NIM   :',mahasiswa.nim);
writeln('Nama  :',mahasiswa.nama);
writeln('Alamat :',mahasiswa.alamat);
writeln('Nilai  :',mahasiswa.nilai);

readln;

end.

```

Statement `with` digunakan untuk menyingkat penulisan dalam membaca elemen record. Sintaks umumnya adalah `with namarecord do`. Simpan ulang program **Praktikum11A.pas** dengan nama **Praktikum11B.pas**. Modifikasi kode program menjadi:

```

with mahasiswa do
begin
nim := '235121';
nama := 'Upin Ipin';
alamat:= 'Jl. Otista no 64c';
nilai:= 98;

end;

```

sehingga dengan menggunakan `with` maka kita tidak perlu untuk menuliskan nama record sebelum elemen recordnya. Jalankan program **Praktikum11B.pas**. Apakah output yang dihasilkan sama dengan program **Praktikum11A.pas**?

B. Record dalam Array

Dalam contoh sebelumnya penggunaan tipe data record hanya dapat menyimpan satu data saja. (satu data mahasiswa saja) . Untuk dapat menyimpan sejumlah data mahasiswa maka dapat digunakan array yang bertipe record. Untuk mengakses tiap data mahasiswa pada array data mahasiswa, kita bisa memanfaatkan indeks array seperti pada pelajaran array sebelumnya.

Simpan program di bawah ini dengan nama **Praktikum11C.pas**. Tambahkan potongan program untuk menampilkan nim, nama, alamat, dan nilai masing-masing mahasiswa untuk kesepuluh mahasiswa pada array. Jalankan program dan pastikan output yang dihasilkan sesuai dengan data yang telah diinputkan.

```

type

```

```

    data_mahasiswa = record
        nim : string[10];
        nama : string[25];
        alamat : string[20];
        nilai : longint;
    end;
var
    mahasiswa : array[1..10] of data_mahasiswa;
    i : integer;
begin
    for i:= 1 to 10 do
        begin
            writeln('Input mahasiswa ke-', i);
            with mahasiswa[i] do
                begin
                    write('kode='); readln(nim);
                    write('nama='); readln(nama);
                    write('alamat='); readln(alamat);
                    write('nilai='); readln(nilai);
                end;
            end;

            for i:=1 to 10 do
                begin
                    writeln('Data Mahasiswa ke-', i);
                    write('kode='); writeln(mahasiswa[i].nim);
                    write('nama='); writeln(mahasiswa[i].nama);
                    write('alamat='); writeln(mahasiswa[i].alamat);
                    write('nilai='); writeln(mahasiswa[i].nilai);
                end;
            end.

```

C. Array dalam Record

Jika dalam suatu record terdapat beberapa field yang sama tipenya maka dapat kita gunakan array.

Contoh ada data barang yang mempunyai struktur:

- Nama barang -> bertipe String
- Jumlah unit barang ke 1 -> bertipe Byte
- Jumlah unit barang ke 2 -> bertipe Byte
- Jumlah unit barang ke 3 -> bertipe Byte

Maka deklarasi recordnya dapat kita nyatakan sebagai berikut.

```
type
    data_brg = record
        namaBrg : string[15];
        unitBrg : array[1..3] of byte;
    end;
var
    Barang : array[1..10] of data_brg;
```

Untuk mengakses array unitBrg dalam record data_brg maka dapat kita gunakan indeks array seperti contoh berikut:

```
Barang[1].unitBrg[1] := 5;
```

Ketik ulang program di bawah ini dan simpan dengan nama **Praktikum11D.pas**

```
type
    data_brg = record
        namaBrg : string[15];
        unitBrg : array[1..3] of byte;
    end;
var
    Barang : array[1..10] of data_brg;
    i : integer;
```

```

Begin
    //Inisialisasi data di sini
    For i:=3 to 10 do begin
        With barang[i] do
            Begin
                Writeln('Barang ke-', i);
                Write('nama='); Readln(namaBrg);
                Write('jumlah 1= '); Readln(unitBrg[1]);
                Write('jumlah 2= '); Readln(unitBrg[2]);
                Write('jumlah 3= '); Readln(unitBrg[3]);
            End;
        End;
    End;
End.

```

Tambahkan potongan program untuk menginisialisasi data barang pada bagian komentar, dengan rincian data sebagai berikut, lalu jalankan program. Pastikan output yang ditampilkan sudah sesuai.

	namaBrg	unitBrg[1]	unitBrg[2]	unitBrg[3]
Barang 1	Scanner	4	5	3
Barang 2	Printer	1	2	4

D. Record dalam Record

Tipe record dapat menjadi elemen dari record lainnya. Sebuah record dapat memiliki elemen yang juga bertipe record.

Contoh: sebuah data pegawai mempunyai struktur sebagai berikut :

- Nama pegawai -> string
- Mulai masuk -> - Tgl
 - Bln
 - Thn
- Alamat pegawai -> - Jalan
 - Kota
- Nilai -> - Nilai pokok
 - Lembur
 - Tunjangan

Maka deklarasi record data pegawai tersebut dapat kita nyatakan sebagai berikut:

```

type
    masuk = record
        tgl : 1..31;
        bln : 1..12;
        thn : integer;
    end;
    alamat = record
        jalan : string[20];
        kota : string[10];
    end;
    nilaipeg = record
        pokok,tunjangan,lembur : real;
    end;
    pegawai = record
        nama : string[20];
        tglmasuk : masuk;
        almt : alamat;
        nilai : nilaipeg;
    end;
var
    datapegawai : array [1..10] of pegawai;

```

atau bisa juga kita deklarasikan secara langsung sebagai berikut:

```

type
    pegawai = record
        nama: string[20];
        tglmasuk: record
            tgl : 1..31;
            bln : 1..12;
            thn : integer;
        end;
        alamat : record
            jalan : string[20];
            kota : string[10];
        end;
        nilaipeg : record
            pokok,tunjangan,lembur : real;
        end;
    end;
var
    datapegawai : array [1..100] of pegawai;

```

Ketik ulang salah satu deklarasi record data pegawai di atas dan simpan dengan nama **Praktikum11E.pas**. Jalankan program dan pastikan tidak ada error. Tambahkan potongan program untuk menyimpan dan menampilkan data pegawai tertentu.

Misalnya, pengguna ingin menyimpan data pegawai dengan indeks 50. Maka, untuk mengakses misalnya nilai pokok dari pegawai nomor 50, caranya adalah seperti ini:

```
Datapegawai[50].nilaipeg.pokok := 5000000;
```

Untuk mengakses nama kota dari pegawai nomor 50 adalah sebagai berikut:

```
Datapegawai[50].alamat.kota := 'Bekasi';
```

Jalankan ulang **Praktikum11E.pas** yang telah dilengkapi dan pastikan tidak ada error.

5. Penugasan

- 1) Buatlah program statistik sederhana yang memiliki fungsi input data, menampilkan semua data, menampilkan bilangan terkecil, bilangan terbesar, dan rata-rata. Data-data disimpan dalam array berukuran 10 elemen. Fungsi-fungsi tersebut dibuat dengan menggunakan sub program sebagaimana yang telah dipelajari sebelumnya.

```
Selamat datang di aplikasi statistik sederhana
Silahkan pilih menu berikut:
1. Input data
2. Tampilkan semua data
2. Tampilkan data terkecil
3. Tampilkan data terbesar
4. Tampilkan rata-rata
5. Keluar
```

Simpan program dengan nama kelas_nim_praktikum11F.pas.

- 2) Dengan memanfaatkan array, buatlah program untuk:
 - A. Melakukan perkalian matriks berukuran 3 X 3
 - B. Menghitung determinan matriks berukuran 3 X 3
 - C. Melakukan transpose matriks berukuran 3 X 3
 - D. Melakukan invers matriks berukuran 3 X 3

Simpan program dengan nama kelas_nim_praktikum11G.pas.

3) Buatlah program konversi nilai desimal ke biner dengan menggunakan rekursi

```
Input : 7  
Output :111
```

```
Input :10  
Output :1010
```

Simpan program dengan nama kelas_nim_praktikum11H.pas.

Gabungkan ketiga file penugasan menggunakan **zip/rar** dan beri nama dengan format **kelas_nim_praktikum11.rar**. Unggah file **zip/rar** pada Google Classroom sesuai dengan batas waktu yang telah ditetapkan. Keterlambatan pengumpulan dikenakan sanksi pemotongan nilai sebesar 10 poin per jam.