Adam Mukharil Bachtiar
English Class
Informatics Engineering 2011

Algorithms and Programming

Record

Steps of the Day

Let's Start

Definition of Record

Background of Record

I need a program that **similar with array program** but can be composed with **different data types**.

What is Record

Data structure that contains of **several fields**

(more than one) which has **different data types**.

Illustration of Record

Records were named **Lecturer 1 and Lecture 2**,

consists of **3 fields** each of its.

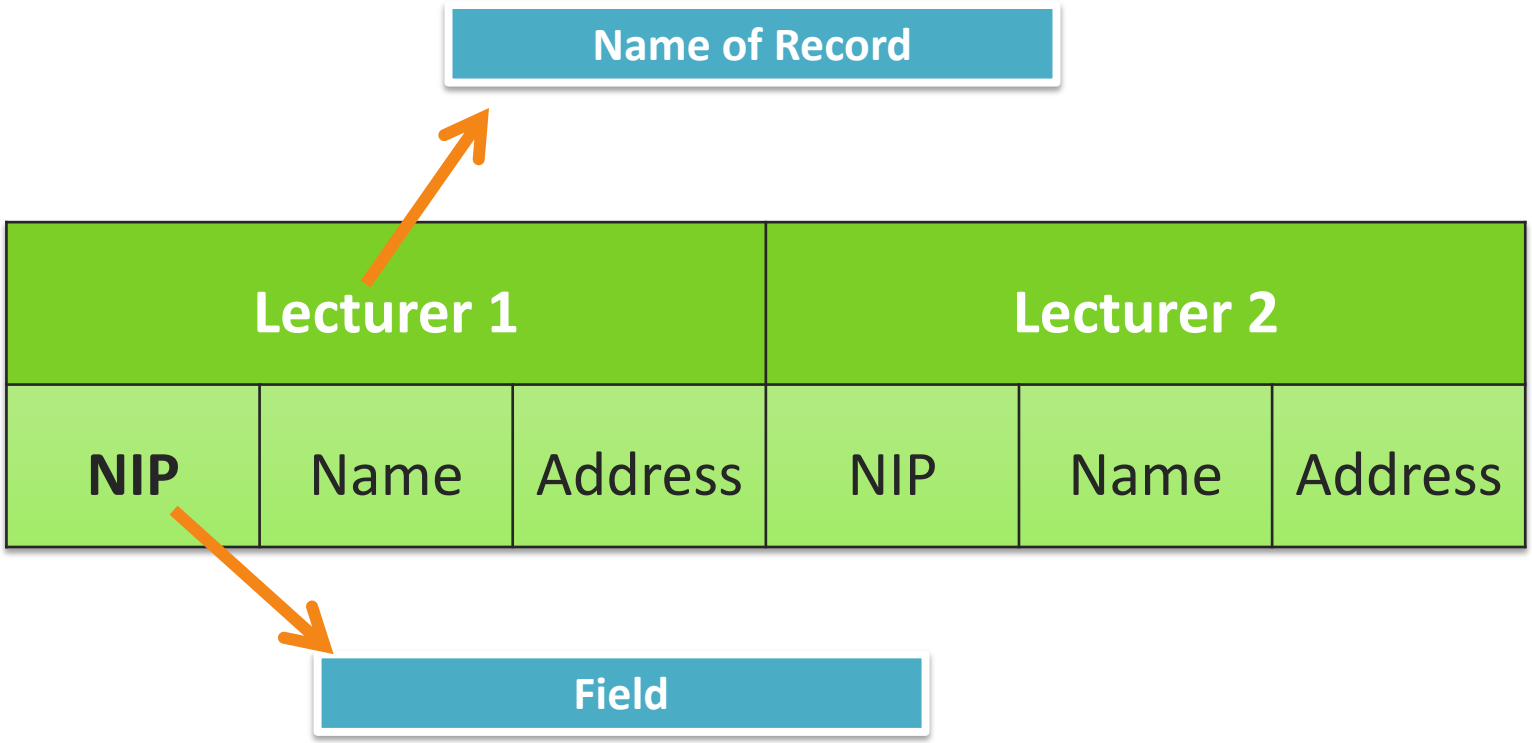
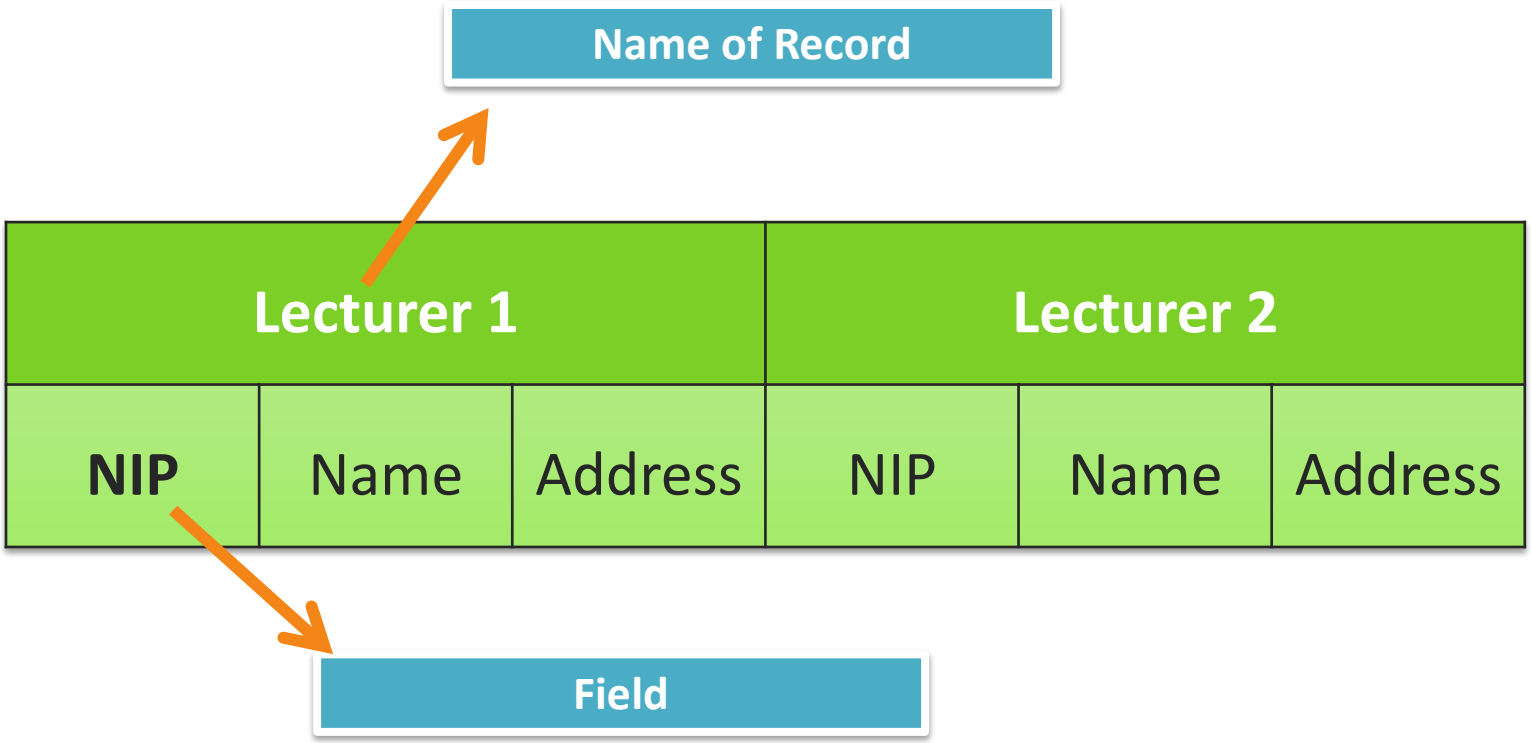


Illustration of Record



If you want to access NIP from Lecturer 1, yu can do with **Lecturer1.NIP**

Application of Record

Definition and Structures of Record

Steps in Record

- Declare record
- Initialize record
- Accessing record (input, operate, and output)

Record Declaration (Algorithm)

```
Kamus:
   type
      TipeRecord = record
      < field_1 : TipeData_1,
        field_2 : TipeData_2,
        ..
        field_n :TipeData_n >
      endrecord

   NamaRecord : TipeRecord
```

Example of Record Declaration (Algorithm)

```
Kamus :

   type

      RecordDosen = record

      < NIP   : integer,

        Nama  : string,

        Gaji  : real >

      endrecord


Dosen : RecordDosen
```

Record Declaration (PASCAL)

```
type
    TipeRecord = record
      field_1 : TipeData_1;
      field_2 : TipeData_2;
      ..
      field_n :TipeData_n;
    end;


var
    NamaRecord : TipeRecord;
```

Example of Record Declaration (PASCAL)

```
type
    RecordDosen = record
      NIP   : longint;
      Nama  : string;
      Gaji  : double;
    end;


var
   Dosen: RecordDosen;
```

Record Initialization (Algorithm)

Format:

```
NamaRecord.NamaField ← DefaultValue
```

Example:

```
Dosen.NIP  ← 0
Dosen.Nama ← ''
Dosen.Gaji ← 0
```

Record Initialization (Pascal)

Format:

```
NamaRecord.NamaField := DefaultValue;
```

Example:

```
Dosen.NIP  := 0;
Dosen.Nama := '';
Dosen.Gaji := 0;
```

Input Value to Record (Algorithm)

Format:

```
input(NamaRecord.NamaField)
```

Example:

```
input(Dosen.NIP)
input(Dosen.Nama)
input(Dosen.Gaji)
```

Input Value to Record (Pascal)

Format:

```
readln(NamaRecord.NamaField);
```

Example:

```
readln(Dosen.NIP);
readln(Dosen.Nama);
readln(Dosen.Gaji);
```

Output Value from Record (Algorithm)

Format:

```
output(NamaRecord.NamaField)
```

Example:

```
output(Dosen.NIP)

output(Dosen.Nama)

output(Dosen.Gaji)
```

Output Value from Record (Pascal)

Format:

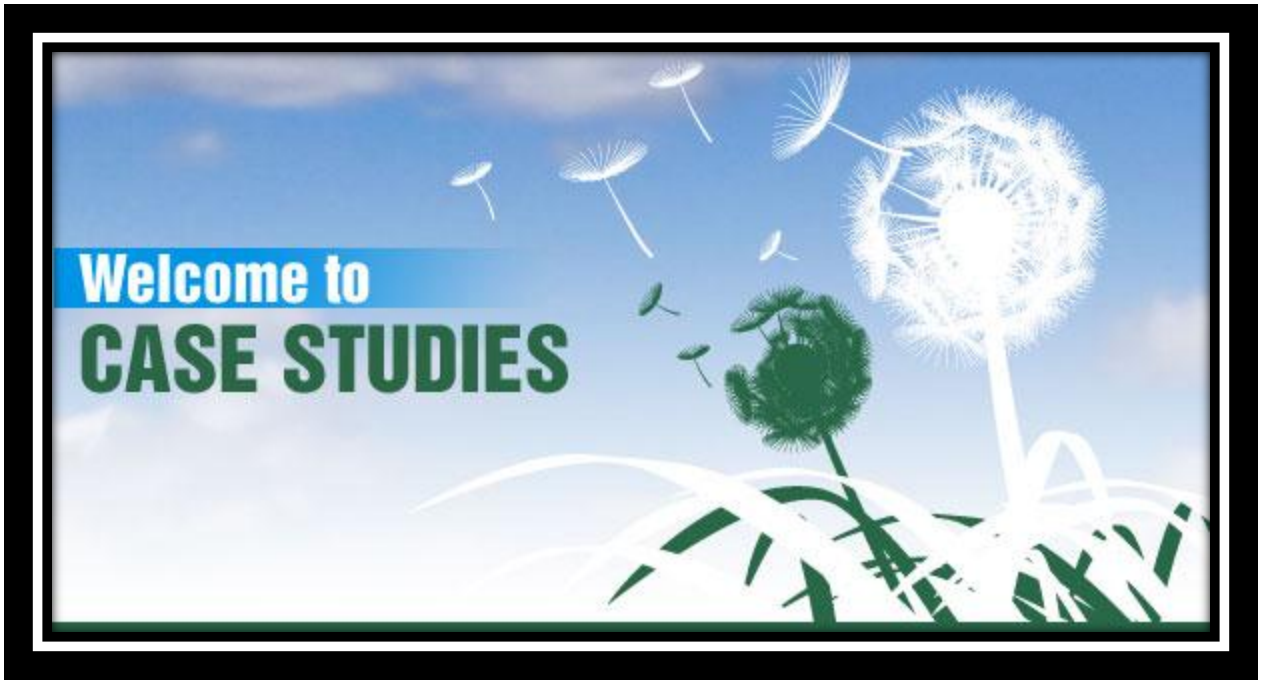
```
writeln(NamaRecord.NamaField);
```

Example:

```
writeln(Dosen.NIP);

writeln(Dosen.Nama);

writeln(Dosen.Gaji);
```


Welcome to
CASE STUDIES

Example of Record (Algorithm)

```
1   Algoritma RecordDosen
2   {I.S.: Dideklarasikan dua buah record dosen}
3   {F.S.: Menampilkan isi record}
4
5   Kamus:
6   type
7         RecordDosen = record
8         < NIP   : integer,
9           Nama  : string,
10          Gaji  : real >
11        endrecord
12
13  Dosen1,Dosen2 : RecordDosen
```

Example of Record (Algorithm)

```
14  Algoritma:
15        {input record}
16        input(Dosen1.NIP)
17        input(Dosen1.Nama)
18        input(Dosen1.Gaji)
19
20        input(Dosen2.NIP)
21        input(Dosen2.Nama)
22        input(Dosen2.Gaji)
23
24        {Operasi field record}
25        Dosen1.Gaji ← Dosen1.Gaji + 1000000 {Tambah THR}
26        Dosen2.Gaji ← Dosen2.Gaji - 100000 (Karena telat}
27
```

Example of Record (Algorithm)

```
28      {Output record}
29      output(Dosen1.NIP)
30      output(Dosen1.Nama)
31      output(Dosen1.Gaji)
32
33      output(Dosen2.NIP)
34      output(Dosen2.Nama)
35      output(Dosen2.Gaji)
```

Example of Record (Pascal)

```
1   program RecordDosenIF;
2   uses crt;
3
4   type
5       RecordDosen=record
6         NIP:longint;
7         Nama:string;
8         Gaji:double;
9       end;
10
11  var
12      Dosen1,Dosen2:RecordDosen;
13
```

Example of Record (Pascal)

```
14  {input record}
15      write('Masukkan NIP dosen pertama  : ');
16      readln(Dosen1.NIP);
17      write('Masukkan Nama dosen pertama : ');
18      readln(Dosen1.Nama);
19      write('Masukkan Gaji dosen pertama : ');
20      readln(Dosen1.Gaji);
21
22      writeln();
23      write('Masukkan NIP dosen kedua    : ');
24      readln(Dosen2.NIP);
25      write('Masukkan Nama dosen kedua   : ');
26      readln(Dosen2.Nama);
27      write('Masukkan Gaji dosen kedua   : ');
```

Example of Record (Pascal)

```
28      readln(Dosen2.Gaji);
29
30      {Operasi pada field record}
31      Dosen1.Gaji:=Dosen1.Gaji+1000000; {karena THR}
32      Dosen2.Gaji:=Dosen2.Gaji-100000;  {karena telat}
33
34      {output record}
35      writeln();
37      writeln('NIP dosen pertama  = ',Dosen1.NIP);
38      writeln('Nama dosen pertama = ',Dosen1.Nama);
39      writeln('Gaji dosen pertama = ',Dosen1.Gaji:0:2);
40
```

Example of Record (Pascal)

```
41      writeln();
42      writeln('NIP dosen kedua    = ',Dosen2.NIP);
43      writeln('Nama dosen kedua   = ',Dosen2.Nama);
44      writeln('Gaji dosen kedua   = ',Dosen2.Gaji:0:2);
45
46      writeln();
47      write('Tekan sembarag tombol untuk menutup...');
48      readkey();
49  end.
```

Example of Record (Pascal)

```
54              jumlah2:=jumlah2+bil2[i];
55          end;
56          writeln('Jumlah elemen array bil 2 = ',jumlah2);
57
58          writeln();
59          write('Tekan sembarang tombol untuk menutup...');
60          readkey();
61  end.
```

Array of Record

Definition and Structures of Array of Record

Background of Array of Record

I have **lecturer's record** but i need

lots of variables to declare

lecturers in program.

What is Array of Record

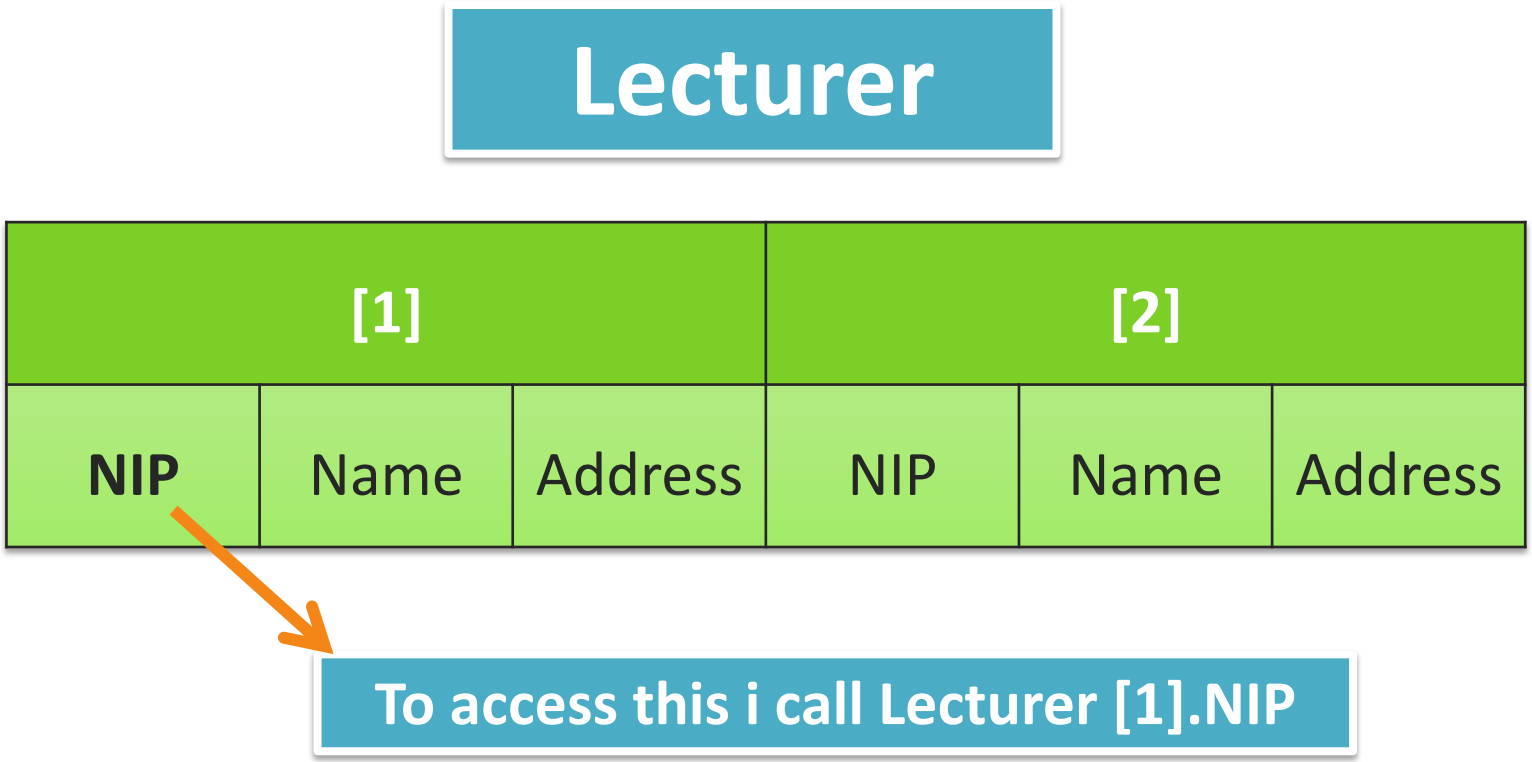
Record that **declare** using **array's form**.

It can be made using **all ways of array's declaration** (three ways).

Illustration of Array of Record

Had been declared an array that had **Lecturer**

type consists of **3 fields** each of element.



Array of Record Declaration (Algorithm)

```
Kamus:
  const
    maks = value
  type
    TipeRecord = record
    < field_1 : TipeData_1,
      field_2 : TipeData_2,
      ..
      field_n : TipeData_n >
    endrecord
    NamaArrayofRecord = array [1..maks] of TipeRecord

  NamaRecord : NamaArrayofRecord
```

Example of Array of Record Declaration (Algorithm)

```
Kamus:
   const
     maks = 20
   type
     DosenIF = record
     < NIP   : integer,
       Nama  : string,
       Gaji  : real >
     endrecord
     ArrayDosenIF = array [1..maks] of DosenIF

   Dosen: ArrayDosenIF
```

Array of Record Declaration (Pascal)

```
const
   maks = value;
type
   TipeRecord = record
      field_1 : TipeData_1;
      field_2 : TipeData_2;
      ..
      field_n : TipeData_n;
   end;
   NamaArrayofRecord = array [1..maks] of TipeRecord;

var
   NamaRecord : NamaArrayofRecord;
```

Example of Array of Record Declaration (Pascal)

```
const
    maks = 20;
type
    DosenIF = record
        NIP  : longint;
        Nama : string;
        Gaji : double;
    end;
    ArrayDosenIF = array [1..maks] of DosenIF;
var
    Dosen: ArrayDosenIF;
```

Record Initialization (Algorithm)

Format:

```
NamaRecord[indeks].NamaField ← DefaultValue
```

Example:

```
Dosen[1].NIP  ← 0
Dosen[1].Nama ← ''
Dosen[1].Gaji ← 0
```

Record Initialization (Pascal)

Format:

```
NamaRecord[indeks].NamaField := DefaultValue;
```

Example:

```
Dosen[1].NIP  := 0;
Dosen[1].Nama := '';
Dosen[1].Gaji := 0;
```

Input Value to Array of Record (Algorithm)

Format:

```
input(NamaRecord[indeks].NamaField)
```

Example:

```
input(Dosen[1].NIP)

input(Dosen[1].Nama)

input(Dosen[1].Gaji)
```

Input Value to Array of Record (Pascal)

Format:

```
readln(NamaRecord[indeks].NamaField);
```

Example:

```
readln(Dosen[1].NIP);
readln(Dosen[1].Nama);
readln(Dosen[1].Gaji);
```

Output Value from Array from Record (Algorithm)

Format:

```
output(NamaRecord[indeks].NamaField)
```

Example:

```
output(Dosen[1].NIP)

output(Dosen[1].Nama)

output(Dosen[1].Gaji)
```

Output Value from Array from Record (Pascal)

Format:

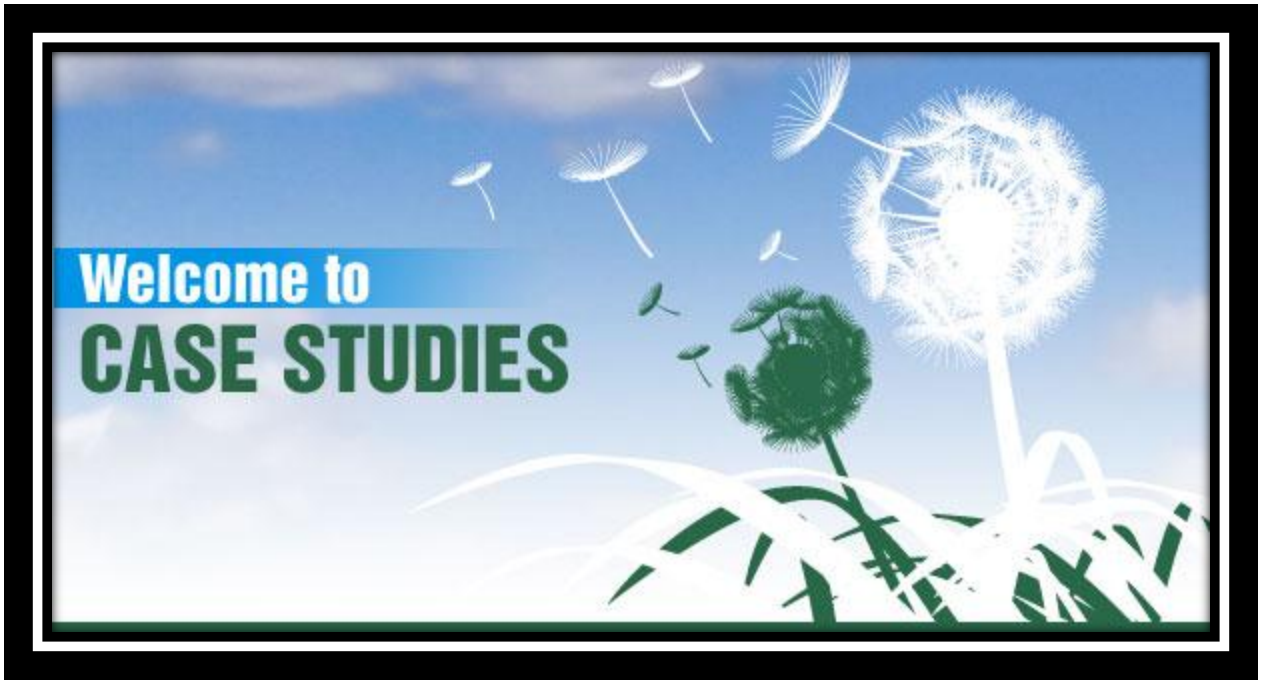
```
writeln(NamaRecord[indeks].NamaField);
```

Example:

```
writeln(Dosen[1].NIP);

writeln(Dosen[1].Nama);

writeln(Dosen[1].Gaji);
```


Welcome to
CASE STUDIES

Example of Array of Record (Algorithm)

```
1   Algoritma ArrayRecordMakananMinuman
2   {I.S : didefinisikan dua array of record food and drink}
3   {F.S : menampilkan array of record beserta operasinya}
4
5   const
6       maks=3;
7   type
8       RecordMakanan = record
9       < KodeMakanan:integer,
10        NamaMakanan:string,
11        HargaMakanan:real,
12        DiskonMakanan:real >
13      endrecord
```

Example of Array of Record (Algorithm)

```
14      RecordMinuman = record
15      < KodeMinuman:integer,
16        NamaMinuman:string,
17        HargaMinuman:real,
18        DiskonMinuman:real >
19      endrecord
20      {array of record}
21      ArrayMakanan = array [1..maks] of RecordMakanan;
22      ArrayMinuman = array [1..maks] of RecordMinuman;
23
24  Makanan:ArrayMakanan;
25  Minuman:ArrayMinuman;
26  TotalHarga:real;
27  i:integer;
```

Example of Array of Record (Algorithm)

```
28  Algoritma:
29      {input record}
30      for i ← 1 to maks do
31        input(Makanan[i].KodeMakanan)
32        input(Makanan[i].NamaMakanan);
33        input(Makanan[i].HargaMakanan)
34        input(Makanan[i].DiskonMakanan)
35      endfor
37      for i ← 1 to maks do
38        input(Minuman[i].KodeMinuman)
39        input(Minuman[i].NamaMinuman)
40        input(Minuman[i].HargaMinuman)
41        input(Minuman[i].DiskonMinuman)
42      endfor
```

Example of Array of Record (Algorithm)

```
43    {perhitungan total harga}
44    TotalHarga ← 0
45    for i ← 1 to maks do
         TotalHarga ← TotalHarga+(Makanan[i].HargaMakanan
                      (Makanan[i].HargaMakanan*Makanan[i].DiskonMakanan))
                      +(Minuman[i].HargaMinuman-
                      (Minuman[i].HargaMinuman*Minuman[i].DiskonMinuman))
46    endfor
47    {output record}
48    for i ← 1 to maks do
49        output(Makanan[i].KodeMakanan)
50        output(Makanan[i].NamaMakanan)
51        output(Makanan[i].HargaMakanan)
52        output(Makanan[i].DiskonMakanan)
53    endfor
```

Example of Array of Record (Algorithm)

```
54      for i ← 1 to maks do
55          output(Minuman[i].KodeMinuman)
56          output(Minuman[i].NamaMinuman)
57          output(Minuman[i].HargaMinuman)
58          output(Minuman[i].DiskonMinuman)
59      endfor
60
61      output(TotalHarga);
```

Example of Array of Record (Pascal)

```
1   program MenuMakananMinuman;
2   uses crt;
3
4   const
5       maks=3;
6   type
7       RecordMakanan = record
8           KodeMakanan:integer;
9           NamaMakanan:string;
10          HargaMakanan:real;
11          DiskonMakanan:real;
12      end;
13
```

Example of Array of Record (Pascal)

```
14      RecordMinuman = record
15         KodeMinuman:integer;
16         NamaMinuman:string;
17         HargaMinuman:real;
18         DiskonMinuman:real;
19      end;
20      {array of record}
21      ArrayMakanan=array [1..maks] of RecordMakanan;
22      ArrayMinuman=array [1..maks] of RecordMinuman;
23  var
24     Makanan:ArrayMakanan;
25     Minuman:ArrayMinuman;
26     TotalHarga:real;
27     i:integer;
```

Example of Array of Record (Pascal)

```
28      begin
29     {input record}
30     for i:=1 to maks do
31     begin
32       write('Masukkan kode makanan ',i,'    : ');
33       readln(Makanan[i].KodeMakanan);
34       write('Masukkan nama makanan ',i,'    : ');
35       readln(Makanan[i].NamaMakanan);
37       write('Masukkan harga makanan ',i,'   : ');
38       readln(Makanan[i].HargaMakanan:0:2);
39       write('Masukkan diskon makanan ',i,' : ');
40       readln(Makanan[i].DiskonMakanan:0:2);
41     end;
```

Example of Array of Record (Pascal)

```
42      writeln();
43      for i:=1 to maks do
44      begin
45        write('Masukkan kode Minuman ',i,'    : ');
46        readln(Minuman[i].KodeMinuman);
47        write('Masukkan nama Minuman ',i,'    : ');
48        readln(Minuman[i].NamaMinuman);
49        write('Masukkan harga Minuman ',i,'   : ');
50        readln(Minuman[i].HargaMinuman:0:2);
51        write('Masukkan diskon Minuman ',i,' : ');
52        readln(Minuman[i].DiskonMinuman:0:2);
53      end;
54
```

Example of Array of Record (Pascal)

```
55      {perhitungan total harga}
56    TotalHarga:=0;
57    for i:=1 to maks do
         TotalHarga:=TotalHarga+(Makanan[i].HargaMakanan
                     (Makanan[i].HargaMakanan*Makanan[i].DiskonMakanan))
                     +(Minuman[i].HargaMinuman-
                     (Minuman[i].HargaMinuman*Minuman[i].DiskonMinuman));
58    {output record}
59    clrscr();
60    for i:=1 to maks do
61    begin
62        writeln('Kode makanan ',i,' adalah ',Makanan[i].KodeMakanan);
63        writeln('Nama makanan ',i,' adalah ',Makanan[i].NamaMakanan);
```

Example of Array of Record (Pascal)

```
64      writeln('Harga makanan ',i,' adalah ',Makanan[i].HargaMakanan:0:2);
65      writeln('Diskon makanan ',i,' adalah ',Makanan[i].DiskonMakanan:0:2);
66    end;
67
68    writeln();
69    for i:=1 to maks do
70    begin
71      writeln('Kode Minuman ',i,' adalah ',Minuman[i].KodeMinuman);
72      writeln('Nama Minuman ',i,' adalah ',Minuman[i].NamaMinuman);
73      writeln('Harga Minuman ',i,' adalah ',Minuman[i].HargaMinuman);
74      writeln('Diskon Minuman ',i,' adalah ',Minuman[i].DiskonMinuman);
75    end;
76  writeln();
77    writeln('Total harga yang harus dibayar adalah : Rp. ',TotalHarga:0:2);
78    writeln();
79    write('Tekan sembarang tombol untuk menutup...');
80    readkey();
81  end.
```

THANK YOU

GRACIAS

Contact Person:
Adam Mukharil Bachtiar
Informatics Engineering UNIKOM
Jalan Dipati Ukur Nomor. 112-114 Bandung 40132
Email: adfbipotter@gmail.com
Blog: http://adfbipotter.wordpress.com

Copyright © Adam Mukharil Bachtiar 2011