



# Tipul de date articol (record)

*Proiect realizat de: Furculiță  
Andreea, clasa a 10-a "C"*

*Profesor: Guțu Maria*

*Anul 2018*

# Objective

- Înțelegerea și însușirea structurii tipului de date record ;
- Înțelegerea modului de specificare a componentelor unei variabile de tip record;
- Cunoașterea modului în care se citesc de la tastatură datele unei variabile de acest tip;
- Cunoașterea scrierii programelor cu ajutorul tipului de date articol;
- Formularea concluziilor și înțelegerea rolului acestui tip de date.

# În ce constă tipul de date *record*?

Mulțimea de valori ale unui tip de date record este constituită din articole (înregistrări). Articolele sînt formate din componente, denumite câmpuri. Fiecare câmp are un nume (identificator de câmp). Spre deosebire de componentele unui tablou, câmpurile pot fi de tipuri diferite. Un tip de date articol se definește printr-o structură de forma:

```
type <nume tip> = record  
    <nume câmp 1> : T1;  
    <nume câmp 2> : T2;  
    . . .  
    <nume câmp n> : Tn;  
end;
```

unde T1,T2,...,Tn specifică tipul câmpurilor respective. Tipul unui nume de câmp este arbitrar, astfel un câmp poate să fie la rândul său tot de tip articol.

# Exemplu

```
type Persoana= record  
  Nume: string;  
  Prenume: string;  
  Varsta: integer; {vârsta în ani}  
end;  
var P1, P2: Persoana;
```

Structura datelor de tip **Persoana**:

## **Persoana**

Nume	Prenume	Varsta
string	string	integer

- Fiecare componentă a unei variabile de tip record poate fi specificată explicit, prin numele variabilei și denumirile de câmpuri, separate prin puncte: P1.Nume; P1.Prenume; P1.Varsta.

- De asemenea, tipul de date record poate servi ca tip de bază pentru formarea altor tipuri structurate:

**type** Persoana= **record**

Nume: **string**;

Prenume: **string**;

Varsta: integer;

**end**;

ListaPersoanelor= **array** [1..50] **of** Persoana;

**var** LP: ListaPersoanelor;

- Notăția LP[i] va specifica persoana i din listă.

# Caz în care un câmp este la rândul său de tip record

```
type Data= record
```

```
Ziua: 1..31;
```

```
Luna: 1..12;
```

```
Anul: integer;
```

```
end;
```

```
Persoana= record
```

```
Nume: string;
```

```
Prenume: string;
```

```
DataNasterii: Data;
```

```
end;
```

```
ListaPersoanelor= array [1..50] of Persoana.
```

```
var LP:ListaPersoanelor;
```

- În acest caz, componenta datei nașterii va fi specificată astfel:  
LP[i].DataNasterii.Ziua/ LP[i].DataNasterii.Luna/ LP[i].DataNasterii.Anul.

# Cum se citesc aceste date de la tastatură?

**Program P1;**

**type** Persoana= **record**

Nume: **string**;

Prenume: **string**;

Varsta: integer; {vârsta în ani}

**end**;

**var** P1, P2: Persoana;

**begin**

writeln('Dati datele primei persoane:');

write('Numele: '); readln(P1.Nume);

write('Prenumele: '); readln(P1.Prenume);

write('Varsta: '); readln(P1.Varsta);

P2:=P1; ← Două variabile de tip articol de același tip pot apărea într-o instrucțiune de atribuire.

**end.**



# Exemple de programe

- **Asupra componentelor datelor de tip articol se pot efectua toate operațiile admise de tipul câmpului respectiv.**
- Utilizatorul introduce de la tastatură un număr  $n$  de persoane,  $n \leq 50$ . Tot de la tastatură, el introduce numele, prenumele și vârsta în ani a persoanelor, dar și vârsta în ani căutată. Programul va afișa numele și prenumele personelor ce au vârsta cerută de utilizator.

**\*Afișează persoanele cu vârsta (în ani) căutată.**

**Program P2;**

**type** Persoana= **record**

Nume: **string**;

Prenume: **string**;

Varsta: integer;

**end**;

ListaPersoanelor= **array**[1..50] **of** Persoana;

**var** LP: ListaPersoanelor;

n:1..50;

i,v: integer;

**begin**

writeln('Dati nr de persoane'); readln(n);

**for** i:=1 **to** n **do begin**

writeln('Dati datele persoanei ',i);

write('Numele: '); readln(LP[i].Nume);

write('Prenamele: '); readln(LP[i].Prenume);

write('Varsta: '); readln(LP[i].Varsta);

**end**;

writeln('Dati varsta cautata'); readln(v);

writeln('Varsta de ',v, ' ani o au: ')

**for** i:= 1 **to** n **do**

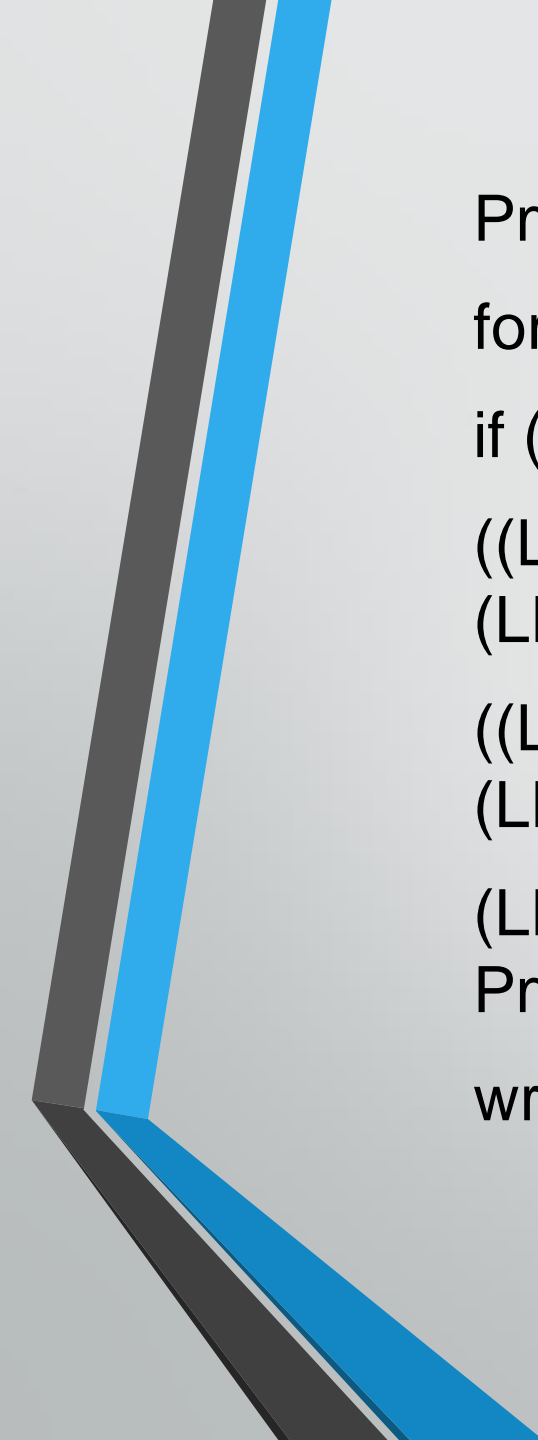
**if** LP[i].Varsta=v **then** writeln(LP[i].Nume, ' ', LP[i].Prenume);

**end.**

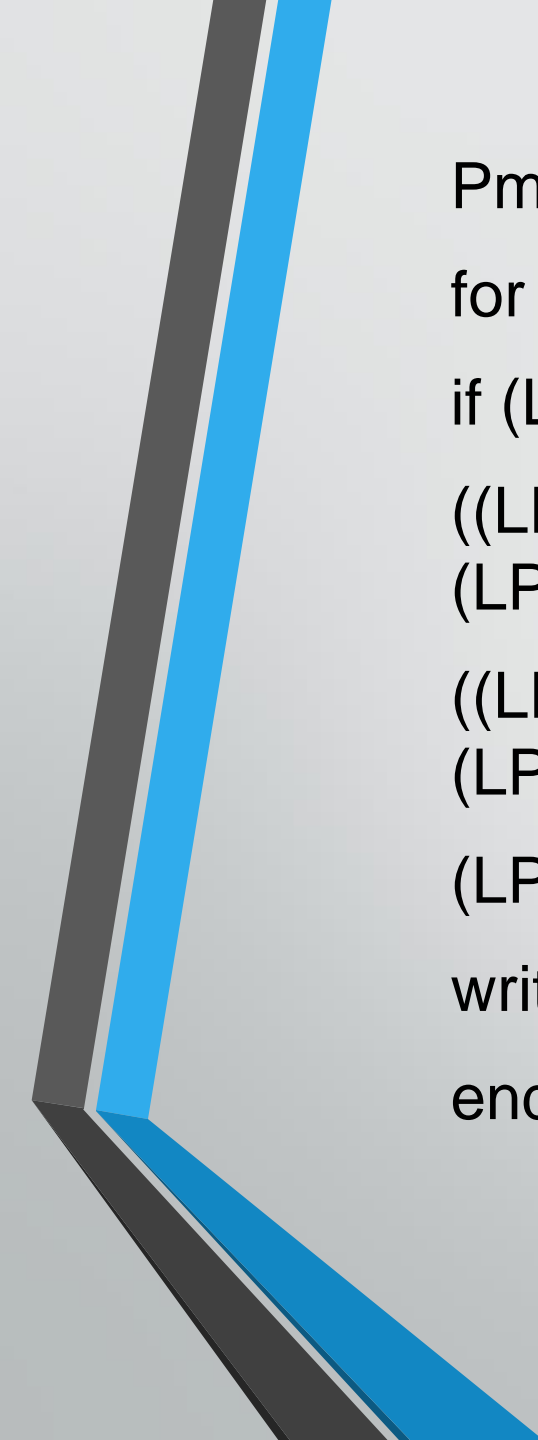
- Utilizatorul introduce de la tastatură un număr  $n$  de persoane,  $n \leq 50$ , precum și numele, prenumele și data nașterii (ziua, luna, anul) persoanelor. Programul va afișa numele și prenumele persoanei celei mai în vârstă și persoanei celei mai tinere.

**\*Găsește cea mai în vârstă și cea mai tânără persoană.**

```
Program P3;
type Data= record
  Ziua: 1..31;
  Luna: 1..12;
  Anul: integer;
end;
Persoana= record
  NumePrenume:string;
  DataNasterii:Data;
end;
ListaPersoane=array[1..50] of Persoana;
var LP:ListaPersoane;
    Pmin, Pmax:Persoana;
    i:integer;
    n:1..50;
begin
  writeln('Dati nr de persoane'); readln(n);
  for i:=1 to n do begin
    writeln('dati datele referitoare la persoana ',i);
    write('Numele si prenumele: '); readln(LP[i].NumePrenume);
    writeln('Data nasterii: ');
    write('ziua: '); readln(LP[i].DataNasterii.Ziua);
    write('luna: '); readln(LP[i].DataNasterii.Luna);
    write('anul: '); readln(LP[i].DataNasterii.Anul);
  end;
```



```
Pmax:=LP[1]; {Pmax-datele celei mai în vârstă persoane}  
for i:=1 to n do  
  if (LP[i].DataNasterii.Anul<Pmax.DataNasterii.Anul) or  
    ((LP[i].DataNasterii.Anul=Pmax.DataNasterii.Anul) and  
    (LP[i].DataNasterii.Luna<Pmax.DataNasterii.Luna)) or  
    ((LP[i].DataNasterii.Anul=Pmax.DataNasterii.Anul) and  
    (LP[i].DataNasterii.Luna=Pmax.DataNasterii.Luna) and  
    (LP[i].DataNasterii.Ziua<Pmax.DataNasterii.Ziua)) then  
    Pmax:=LP[i];  
writeln('Persoana cea mai in varsta este ', Pmax.NumePrenume);
```



```
Pmin:=LP[1]; {Pmin-datele celei mai tinere persoane}
for i:=1 to n do
if (LP[i].DataNasterii.Anul>Pmin.DataNasterii.Anul) or
((LP[i].DataNasterii.Anul=Pmin.DataNasterii.Anul) and
(LP[i].DataNasterii.Luna>Pmin.DataNasterii.Luna)) or
((LP[i].DataNasterii.Anul=Pmin.DataNasterii.Anul) and
(LP[i].DataNasterii.Luna=Pmin.DataNasterii.Luna) and
(LP[i].DataNasterii.Ziua>Pmin.DataNasterii.Ziua)) then Pmin:=LP[i];
writeln('Persoana cea mai in tanara este ', Pmin.NumePrenume);
end.
```

## Concluzie

Utilizarea tipului de date structurat articol (record) permite lucrul cu o cantitate mai mare de date în comparație cu alte tipuri de date structurate studiate până acum. Un avantaj mare pentru tipul articol reprezintă faptul că el poate fi utilizat aproximativ ca un tablou, dar câmpurile sale, spre deosebire de elementele tabloului, pot fi de tipuri total diferite.

# Bibliografie

- Manual de Informatică pentru clasa a 10-a.
- <http://wiki.freepascal.org/Record>