Ministry of Higher Education and Scientific Research
Djilali BOUNAAMA University - Khemis Miliana(UDBKM)
Faculty of Science and Technology
Department of Mathematics and Computer Science

Chapter 5

Structured Data Types: 2 : Records

MI-L1-UEF121 : Algorithms and Data Structures I

Noureddine AZZOUZA

n.azzouza@univ-dbkm.dz

Course Topics

1. Records
    - 2.1 Definition
    - 2.2 Declaration
    - 2.3 Manipulation

Records

Definition

- ✓ A **record** is a set of data that can be of different types, called **Fields**, and grouped under a single user-defined **object** name.

- ✓ A **record** is a **type** of data describing composite information, made up of **several values** which may themselves be of **different types**

Representation

a Record

Field1	Field2	Field3	Field4	Field5
Type1	Type2	Type3	Type4	Type5

- ***Examples***

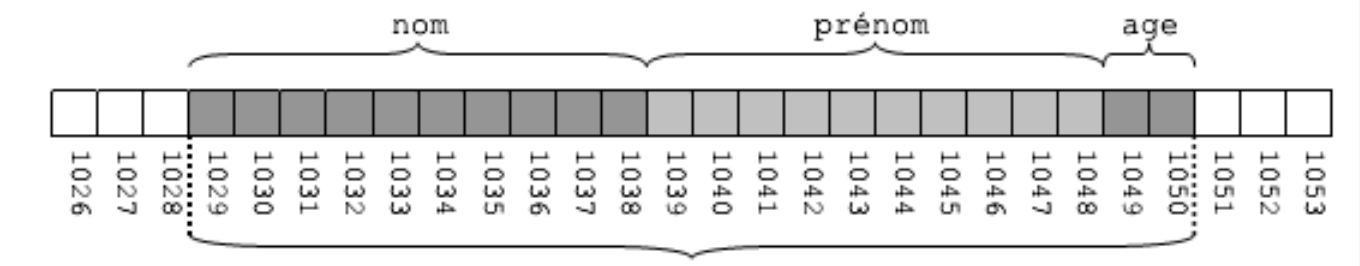
Person

first_name	last_name	age
String	String	integer

Product

Reference	Designation	Quantity	Purchase_price	Selling_price
String	String	integer	Real	Real

- ***Memory Representation***



Declaration

✓ The declaration of a record is carried out by specifying the keyword "**record**", followed by the description of the fields and finished with the keyword "**End**".

```
Type    nom_type = Record
                   Field1 : type 1
                   ...
                   Field n : type n
                End nom_type
```

- Field types can be ***predefined*** or ***defined by the user***
- A Field has exactly the same properties as a variable of the same type
- The Field of a record variable can itself be a record

Declaration : *Examples*

```
Type date = record;
        day: integer;
        month: integer;
        year: integer;
    End date

Var date_entree: date;
```

```
Type person = record;
        nom : String [20];
        prenom : String [20];
        date_naissance: date ;
    End personne

Var P1, P2 : person;
```

```
Type product = record;
        ref: String [10];
        design: String [20];
        qte: integer;
        BP, SP: Real;
    End product

Var  prod: produit;
     T: Array[1..100] of product;
```

Déclaration : *Implementation*

PASCAL

```
Type date = record
        jour: integer;
        mois: integer;
        annee: integer;
        end;

    personne = record
        nom: String;
        prenom: String;
        date_naissance: date;
        end;

    produit = record
        ref: String[10];
        design: String[25];
        qte: integer;
        PA, PV: Real;
        end;
```

C

```
struct date
{   int jour;
    int mois;
    int annee;
};

struct
{   char nom[10];
    char prenom[10];
    struct date date_naissance;
};

struct produit
{   char ref[10];
    char design[25];
    int qte;
    float PA, PV;
};
```

Manipulation

✓ Access to a Field is done by specifying the **name of the record** followed by the **name of the Field** separated by a **dot**.

❑ ***Reading from a Field***

valeur := **Objet.Field** *;*

```
m := date_entree.month;
nom2 := P2.nom;
annee1 := P1. date_naissance.year;
Ecrire('le prix de vente=', prod.SP);
pr1 := T[i].design;
```

❑ ***Writing on the Field***

Objet.Field *:= valeur;*

```
date_entree.day := jj;
Lire (P2.prenom);
P1. date_naissance.annee = 1992;
Prod.qte := 1500 – qte_vente;
T[12].SP := 652.50;
```

Ministry of Higher Education and Scientific Research
Djilali BOUNAAMA University - Khemis Miliana(UDBKM)
Faculty of Science and Technology
Department of Mathematics and Computer Science

Chapter 5

Structured Data Types: 2 : Records

MI-L1-UEF121 : Algorithms and Data Structures I

Noureddine AZZOUZA

n.azzouza@univ-dbkm.dz