

Le type « enregistrement »

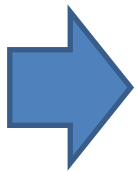
Le type « record » sous DELPHI

Ricco Rakotomalala
Université Lumière Lyon 2

Pourquoi le type enregistrement ?

Idée : regrouper dans une seule **structure** les informations (**champs**, propriétés, attributs) relative à une entité (ex. personne, voiture, maison, etc.).

- Ces champs peuvent être de types (élémentaires) différents.
- Ces champs peuvent faire référence également à un autre type structuré.



Premier pas vers la modélisation des programmes.



Delphi

```
TYPE NOM_ENREGISTREMENT = RECORD
    nom_champ_1 : type données;
    nom_champ_2 : type données;
    ...
END;
```

Exemple : définir le type TPersonne

```
UPersonne.pas
ProjectPersonne UPersonne

unit UPersonne;

interface

//définition du type "personne"
TYPE   TPersonne = record
        //propriétés d'une personne
        nom : shortstring;
        age : integer;
        poids: double;
    end;

//procédures et fonctions
procedure saisie(var p: TPersonne);
procedure affichage(p: TPersonne);

implementation

procedure saisie(var p: TPersonne);
begin
    write('nom = '); readln(p.nom);
    write('age = '); readln(p.age);
    write('poids = '); readln(p.poids);
end;

procedure affichage(p: TPersonne);
begin
    writeln('nom = ', p.nom);
    writeln('age = ', p.age);
    writeln('poids = ', p.poids);
end;

end.
```

Toujours mieux de définir un enregistrement dans une unité dédiée.

Comme TPersonne est défini dans la partie interface de l'unité, elle sera visible à l'extérieur.

Mieux vaut utiliser un type statique pour les chaînes de caractères (pour le stockage dans les fichiers typés).

Saisie() et Affichage() sont décrits dans la partie interface, donc seront visibles à l'extérieur

Notez l'écriture lors de la manipulation de l'instance : `nom_variable.nom_champ`

Exemple : exploiter le type TPersonne

```
ProjectPersonne.dpr
ProjectPersonne | UPersonne
program ProjectPersonne;

{$APPTYPE CONSOLE}

uses
  SysUtils,
  UPersonne;

var m: TPersonne;
begin
  writeln('saisie des informations :');
  saisie(m);
  writeln('affichage des informations :');
  affichage(m);
  readln;
end.
```

Importation du module

Le type TPersonne est référencé

Les procédures saisie() et affichage() peuvent être utilisées.

A la place de l'appel à la procédure « saisie », on aurait pu faire directement :

```
readln(m.nom);
readln(m.age);
readln(m.poids);
```

```
C:\Users\Maison\Desktop\test\ProjectP...
saisie des informations :
nom = pierre
age = 39
poids = 67
affichage des informations :
nom = pierre
age = 39
poids = 6.7000000000000000E+0001
```

Collection d'objets : tableaux d'enregistrements

Un **tableau** est une collection indicée d'**éléments homogènes**, il est donc possible de définir des tableaux d'enregistrements.

Indispensable si on veut manipuler le type Tpersonne dans ce module.

Chaque cellule du tableau contient une personne (avec son nom, âge, poids). **Ce type tableau est exporté** puisque décrit dans la partie interface de ce module.

Procédures et fonctions qui seront exportées.

La variable « tp » est utilisable dans tout module ou prog. principal qui fait référence à l'unité « UTabPersonne » (avec « uses UTabPersonne »).

Accès au poids de n'individu n°i

```
unit UTabPersonne;

interface

//importation du module
uses UPersonne;

//définition du type tableau
Type TTabPersonne = array of TPersonne;

//procédures et fonctions
procedure saisieTab(var t: TTabPersonne);
function poidsMoyenTab(const t: TTabPersonne): double;

//variable globale "tableau de personnes"
var tp: TTabPersonne;

implementation

procedure saisieTab(var t: TTabPersonne);
var i: integer;
begin
  for i:= 0 to length(t)-1 do
    saisie(t[i]);
  end;

function poidsMoyenTab(const t: TTabPersonne): double;
var s: double; i: integer;
begin
  s:= 0.0;
  for i:= 0 to length(t)-1 do
    s:= s + t[i].poids;
  result:= s/length(t);
end;

end. |
```

Prog. principal : tableaux d'enregistrements

Indispensable si on veut manipuler le type TTabPersonne

```
ProjectPersonne.dpr
ProjectPersonne | UPersonne | UTabPersonne
program ProjectPersonne;

{$APPTYPE CONSOLE}

uses
  SysUtils,
  UPersonne,
  UTabPersonne;

var n: integer;
begin
  //nombre de personnes à gérer
  writeln('combien de personnes :');
  readln(n);
  //dimensionner le tableau "tp"
  SetLength(tp, n);
  //saisir les informations
  saisieTab(tp);
  //afficher la moyenne
  writeln('moyenne poids = ', poidsMoyenTab(tp));
  //détruire (libérer) le tableau
  Finalize(tp);
  readln;
end.
```

Parce qu'elle est déclarée dans la partie INTERFACE de UTabPersonne, la variable « **tp** » est directement utilisable ici !!!

```
C:\Users\Maison\Desktop\test\ProjectPerson...
combien de personnes :
2
nom = pierre
age = 54
poids = 68
nom = paul
age = 43
poids = 89
moyenne poids = 7.850000000000000E+0001
```

FIN...

Les mêmes concepts sont – à peu de choses près – présents dans tous les langages de programmation...