

# Les structures

Les structures en C permettent de créer des types personnalisés, un peu comme les classes en Java mais sans méthodes (askip c'est possible de faire des méthodes mais c'est très peu commun et donc pas expliqué dans ce cours).

Il existe deux manière de faire une structure en C mais il vaut mieux toujours rester consistant sur la manière utilisée.

Voici une première manière de faire une structure en C :

```
struct etudiant {  
    int matricule;  
    char nom[50+1];  
    char prenom[50+1];  
    char adresse[200+1];  
    char telephone[15+1];  
    float moyenne;  
};
```

Ensuite pour déclarer une variable :

```
struct etudiant un_etudiant;
```

Et voici la seconde manière de faire :

```
typedef struct {  
    int matricule;  
    char nom[50+1];  
    char prenom[50+1];  
    char adresse[200+1];  
    char telephone[15+1];  
    float moyenne;  
} Etudiant;
```

Et ensuite pour déclarer une variable :

```
Etudiant un_etudiant;
```

# Lecture et écriture des données

Etant donné qu'il n'y a pas de méthodes aux structures, il n'y a pas de modificateur `private` sur les attributs, les attributs peuvent donc être accédé et modifié sans limite.

```
/* Lecture d'un struct */
printf("%s %s\n", un_etudiant.nom, un_etudiant.prenom);

/* Ecriture d'un struct */
un_etudiant.matricule = 123456;
```

## Partage des structures entre programmes

Pour partager les structures entre plusieurs programmes on peut simplement mettre la déclaration du struct dans un fichier en-tête (`.h`).

---

Revision #1

Created 2 October 2023 18:31:59 by SnowCode

Updated 6 January 2024 19:14:47 by SnowCode