

C structure

STRUCTURE

- ⦿ Structures (also called structs) are a way to group several related variables into one place.
- ⦿ Each variable in the structure is known as a **member** of the structure.
- ⦿ Unlike an array, a structure can contain many different data types (int, float, char, etc.).

```
#include<stdio.h>  
#include<conio.h>
```

```
struct a
```

```
{  
int member1;  
char member2;  
float member3;  
};
```

```
Struct a var1;
```

```
void main()  
{  
clrscr();  
var1.member1=15;  
var1.member2='a';  
var1.member3=10.55;  
printf("Value of Member 1 =%d\n",var1.member1);  
printf("Value of Member 2 =%c\n",var1.member2);  
printf("Value of Member 3 =%.2f\n",var1.member3);  
getch(); }
```

```
void main()
{
    struct book
    {
        char name;
        float price;
        int pages;
    };
    struct book b1;
    clrscr();
    printf( "Enter books name, price  and no of pages of book\n");
    scanf("%c %f %d", &b1.name, &b1.price, &b1.pages);
    printf("You have Entered");
    printf("book name : %c\t price: %f\t page no ; %d ", b1.name,b1.price,b1.pages);
    getch();
}
```

Nested Structure

```
#include<stdio.h>
#include<conio.h>
struct stud_Record
{
    int rno;
    char std[10];
    struct stud_Marks
    {
        char subj_nm[30];
        int subj_mark;
    } marks;
}
result;
```

```
void main()
{
    clrscr();

    printf("\n\t Enter Roll Number:");
    scanf("%d", &result.rno);
    printf("\n\t Enter Standard:");
    scanf("%s", result.std);
    printf("\n\t Enter Subject name:");
    scanf("%s", &result.marks.subj_nm);
    printf("\n\t Enter Marks:");
    scanf("%d",&result.marks.subj_mark);
    printf("\n\n\t Roll Number: %d", result.rno);
    printf("\n\n\t Standard: %s", result.std);
    printf("\nSubject Code: %s", result.marks.subj_nm);
    printf("\n\n\t Marks: %d",result.marks.subj_mark);

    getch();
}
```

UNION

- ⦿ **The Union is a user-defined data type in C language that can contain elements of the different data types just like structure.**
- ⦿ **But unlike structures, all the members in the C union are stored in the same memory location. Due to this, only one member can store data at the given instance.**

```
union un {
    int member1;
    char member2;
    float member3;
};

int main()
{
    union un var1;
    var1.member1 = 15;
    printf("The value stored in member1 = %d",
           var1.member1);
    return 0;
}
```

```
#include<stdio.h>    void main() {
#include<conio.h>    clrscr();
union un            union un var1;
{                  var1.member1=15;
int member1;      var1.member2='a';
char member2;     var1.member3=10.55;
float member3;    printf("Value of Member 1 = %d\n",var1.member1);
};               printf(" Value of Member 2 =%c\n",var1.member2);
                 printf(" Value of Member 3 =%.2f\n",var1.member3);
                 getch(); }
```


UNION & STRUCTURE

```
#include<stdio.h>  
#include<conio.h>  
struct un  
{  
int member1;  
char member2;  
float member3;  
};  
struct un var1;
```

```
void main()  
{  
clrscr();  
//struct un var1;  
var1.member1=15;  
var1.member2='a';  
var1.member3=10.55;  
printf("the value stored in  
member1=%d\n",var1.member1);  
printf("the value stored in  
member2=%c\n",var1.member2);  
printf("the value stored in  
member3=%.2f\n",var1.member3);  
getch(); }
```