# 12. Structures in C

**THEORY EXERCISE**:

- Explain the concept of structures in C. Describe how to declare, initialize, and access structure members.

In C a Structure is a user-defined data type that allows grouping different types of data under a single name each data element in a structure is called a member, different types such as integers, floats or arrays. They define using the 'struct' keyword can be accessed using the dot operators (.).

## 1. Declaration of a Structure

To declare a structure, use the struct keyword.
**Syntax:**

struct StructureName {

    dataType member1;

    dataType member2;

    ...

};

**Example:**

struct Student {

    int rollNo;

    char name[50];

    float marks;

};

This defines a new structure type named Student.

---

## 2. Structure Variable Declaration

You can declare structure variables in two ways:

**Method 1: After the structure definition**

struct Student s1;

**Method 2: With the structure definition**

```c
struct Student {

    int rollNo;

    char name[50];

    float marks;

} s1, s2;
```

---

## 3. Initialization of Structure Members

You can initialize structure members at the time of declaration:

```c
struct Student s1 = {1, "Rahul", 85.5};
```

Note: All members should be initialized in order.

---

## 4. Accessing Structure Members

Use the dot (.) operator to access members of a structure variable.

**Example:**

```c
#include <stdio.h>
struct Student {

    int rollNo;

    char name[50];

    float marks;

};
int main() {

    struct Student s1 = {1, "Rahul", 85.5};

    printf("Roll No: %d\n", s1.rollNo);

    printf("Name: %s\n", s1.name);

    printf("Marks: %.2f\n", s1.marks);

}
```

---

**Summary**

| Task | Syntax/Example |
| --- | --- |
| Declare structure | struct Student {int id; char name[20];}; |
| Declare variable | struct Student s1; |
| Initialize structure | struct Student s1 = {1, "Rahul"}; |
| Access member | s1.id, s1.name |