

Structures in C++

Structure is a compound data type that contains different variables of different types. For example, you want to store Student details like student name, student roll num, student age. You have two ways to do it, one way is to create different variables for each data, but the downfall of this approach is that if you want to store the details of multiple students, in that case it is not feasible to create separate set of variables for each student. The second and best way of doing it by creating a structure like this:

```
struct Student
{
    char stuName[30];
    int stuRollNo;
    int stuAge;
};
```

Now these three members combined will act like a separate variable and you can create structure variable like this:

```
structure_name variable_name
```

So if you want to hold the information of two students using this structure then you can do it like this:

```
Student s1, s2;
```

Then I can access the members of Student structure like this:

```
//Assigning name to first student
s1.stuName = "Ajeet";
//Assigning age to the second student
s2.stuAddr = 22;
```

Similarly I can set and get the values of other data members of the structure for every student. Lets see a complete example to put this up all together:

Structure Example in C++

```
#include <iostream>
using namespace std;
struct Student{
    char stuName[30];
    int stuRollNo;
    int stuAge;
};
int main(){
    Student s;
    cout<<"Enter Student Name: ";
    cin.getline(s.stuName, 30);
    cout<<"ENter Student Roll No: ";
    cin>>s.stuRollNo;
    cout<<"Enter Student Age: ";
    cin>>s.stuAge;
    cout<<"Student Record:"<<endl;
    cout<<"Name: "<<s.stuName<<endl;
    cout<<"Roll No: "<<s.stuRollNo<<endl;
    cout<<"Age: "<<s.stuAge;
    return 0;
}
```

Output:

```
Enter Student Name: Negan
ENter Student Roll No: 4101003
Enter Student Age: 22
Student Record:
Name: Negan
Roll No: 4101003
Age: 22
```

Structure and Function in C++

How to pass structure as an argument to function

Here we have a function `printStudentInfo()` which takes structure `Student` as an argument and prints the details of student using structure variable. The important point to note here is that you should always declare the structure before function declarations, otherwise you will get compilation error.

```
#include <iostream>
using namespace std;
struct Student{
    char stuName[30];
    int stuRollNo;
    int stuAge;
};
void printStudentInfo(Student);
int main(){
    Student s;
    cout<<"Enter Student Name: ";
    cin.getline(s.stuName, 30);
    cout<<"Enter Student Roll No: ";
    cin>>s.stuRollNo;
    cout<<"Enter Student Age: ";
    cin>>s.stuAge;
    printStudentInfo(s);
    return 0;
}
void printStudentInfo(Student s){
    cout<<"Student Record:"<<endl;
    cout<<"Name: "<<s.stuName<<endl;
    cout<<"Roll No: "<<s.stuRollNo<<endl;
    cout<<"Age: "<<s.stuAge;
}
```

Output:

```
Enter Student Name: Rick
Enter Student Roll No: 666123
Enter Student Age: 19
Student Record:
Name: Rick
Roll No: 666123
Age: 19
```

How to return the Structure from a Function

In this example we have two functions one gets the values from user, assign them to structure members and returns the structure and the other function takes that structure as argument and print the details.

```
#include <iostream>
using namespace std;
struct Student{
    char stuName[30];
    int stuRollNo;
    int stuAge;
};
Student getStudentInfo();
void printStudentInfo(Student);
int main(){
    Student s;
    s = getStudentInfo();
    printStudentInfo(s);
    return 0;
}
/* This function prompt the user to input student
 * details, stores them in structure members
 * and returns the structure
 */
Student getStudentInfo(){
    Student s;
    cout<<"Enter Student Name: ";
    cin.getline(s.stuName, 30);
    cout<<"Enter Student Roll No: ";
    cin>>s.stuRollNo;
    cout<<"Enter Student Age: ";
    cin>>s.stuAge;
    return s;
}
void printStudentInfo(Student s){
    cout<<"Student Record:"<<endl;
    cout<<"Name: "<<s.stuName<<endl;
    cout<<"Roll No: "<<s.stuRollNo<<endl;
    cout<<"Age: "<<s.stuAge;
}
```

Output:

```
Enter Student Name: Tyrion lannister
Enter Student Roll No: 333901
Enter Student Age: 39
Student Record:
Name: Tyrion lannister
Roll No: 333901
Age: 39
```