# * Different forms of member functions

C++ offers different types of member functions.
C++ offers various functions as compared to some other languages.

The following types of member functions:-

- Simple member function
- Inline member function
- Static member function
- Friend member function
- Const member function.

## (i) Simple Member function :-

- These are the basic member function, which dont have any special keyword.

- If we define a member function in a class without specifying any particular type then it is called Simple member function.

- It doesn't have any additional requirement therefore it is called simple member function.

```
Syntax:- return_type function_name (parameter)
         {
             function body;
         }
```

# (2) Inline functions :-

- All the member functions defined inside the class defination are by default declared as Inline

- Inline function is a function which gets expanded inline at each point in the program when it is called

- functions are made inline normally when they are small enough to be defined in one or two lines.

- The use of Inline function improves the execution speed of the program.

- User can make a function inline by typing the keyword Inline before

the start of the function.

- The Inline function define before main() fun.

<u>Syntax</u>

Inline data.type function name (argument list)
{
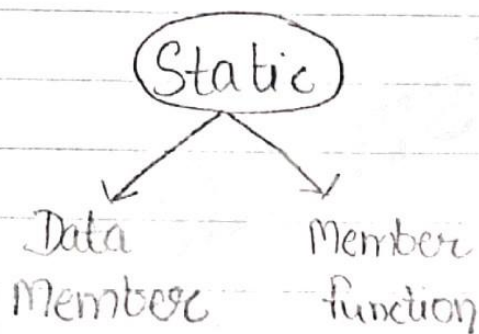   function body;
}

<u>Example:-</u>
```
#include<iostream.h>
#include<conio.h>
inline   int addnum (int x, int y)
{
    return (x+y);
}

void main()
{
    int a = 10;
    int b = 5;
    cout << addnum(a,b);
    getch();
}
```

## (3) Static function

- A function is said to be static, which can only access static member of the class.

- The keyword static is used to before the member function to make a member function static.

```
        (Static)
        /      \
       ↓        ↓
    Data      Member
   Member    function
```

- **Static data member :-** Static data members are class variables whenever we declare a data member as a static either inside or outside of a class called static data member.

- declaration of static data member within class defination is similar to any other variable declaration.

- It starts with keyword static.

```
class abc
{
    static int a;     // within class
}
```

It can be defined outside the class

→

```
int abc :: a = 0;
```

## Static Member function :-

- If we create a member function of a class as a static is called static member function.

- Static member function is independent of any object of the class.

- Static function, can be called even there is no object of class.

- Static function can be called using class name.

- Syntax :-
      static return type function name (arg list);

## Example:-

```cpp
#include <iostream.h>
#include <conio.h>
class demo
{
    Private:
        static int a;   // Static data member
        static int b;
    public:
        static void print() //Static member fun
        {
            cout<<"value of a"<< a << endl;
            cout<<"value of b"<< b << endl;
        }
};
int demo:: a = 10;
int demo:: b = 20;

void main()
{
    demo Ob;
    cout<<"printing through object name";  // Object name
    Ob.print();
    cout<<"Printing through class name";   // class name
    demo:: print();
    getch();  }
```

# (4) Friend Function

- A friend function in C++ is defined as a function that can access private, protected and a public member of class.

- The friend function is declared using the friend keyword.

- A friend can be function, member function, or a class.

- It has the right to access all private and protected members of the class.

- For accessing the data, the declaration of a friend function should be done inside the body of a class starting with keyword friend.

Syntax:-

friend return_type function_name (argument_list);

**Example:-**

```cpp
#include <iostream.h>
#include <conio.h>
class Add
{
    int a,b;        // Private
    public:
    void input()
    {
        cout<<" enter value of a,b";
        cin >> a >> b;
    }
    friend void sum (Add Obj);
};
void sum (Add Obj)
{
    int c;
    c= Obj.a + Obj.b;
    cout << "Sum is = "<<c;
}
void main()
{
    Add A;
    A.input();
    sum (A);
    getch();
}
```

## Characteristics of friend function

- It can be declared either in the private or the public part.

- It can be invoked like a normal function without using the object.

- It cannot access the member names directly and has to use dot membership operator and use an object name with the member name.

## (5) Const member function

The const member functions are the functions which are declared as constant in the program

The object called by these functions cannot be modified
The keyword const is used.

Syntax :- Const data_type functionname (arg list);

example :- Const int sum (void);