

# STATIC MEMBERS

## Can the data members private,protected declared as static Be accessed inside the static function??

A static member function can be called even if no objects of the class exist.

The static functions are accessed using only the class name and the scope resolution operator :: as shown below :

Class\_name::static\_member\_name

A static member function can only access static data member, other static member functions and any other functions from outside the class.

## Do we need to initialise static data members ?

Other than static and global data members initialisation is must. Static and global are created in code section of main memory so they are created at loading time of program and initialised.

## Is it necessary to initialize the static variable after the class i.e outside or can we do it before the class as well?

It is **mandatory** to declare the static variable **outside** the class.

It can be done after defining the class.

## Can we implement Static function outside class just like static variable or is it mandatory to implement it inside class?

Yes it can be done **outside** the class also.

Every function can be implemented either inside or outside.

Except friend function, which is implemented outside only.

## **Can we make static variable and function private ?**

Yes you can make static members also as private.

But they will be accessible only inside the class.

# **FRIEND FUNCTION AND CLASSES**

## **Can two different classes have same friend function ?**

Yes, we can also have the **same** friend function of two **different** classes .

## **Don't we have to declare friend function in the same manner as we declare friend class ?**

No, we declare friend function inside the class and implement them outside the class.

In case of friend class, we are using the class by creating its object, so at least its name should be declared before using it. So we are declaring it before the class.

## **What is the difference b/w Friend Operator Overloading and Friend Function**

Friend functions are global functions that can access all members of an object of a class.

Operator Overloading can be done using **member functions** as well as **friend functions**.

Operator Overloading done by using **friend function** is called as **Friend Operator Overloading**.

### **Friend versus Accessor/Mutator function**

Accessor and Mutator functions can access private members **directly** where as,

Friend Functions can access member upon **object**.

Friend functions are used if a function have to access object of multiple classes, they act as connector.

Usually operator overloading is done using Friend functions.

### **If variable a and b are private in a constructor how are the initialized ?**

a and b can be initialized in a constructor, they can be of any access specifiers types public, private, or protected.

Basically you can access all type of members inside same class regardless it is private, protected or public.

## **INNER CLASSES**

### **Can we create outer class object in inner class and access the data members of the outer class inside the inner class ?**

**Inner** class is **useful** only for **outer** class.

If a class is having lots of code then we can reduce its complexity by defining inner class.

So inner class is useful only inside.

### **What is flush and where do we use it?**

The output is sent to the screen using cout, it does not go immediately to the terminal.

First it will be sent to output buffer and then displayed on the screen.

Sometimes in very large programs, it remains in the buffer and we don't get the output in proper order.

Flush is used to confirm that the buffered is cleared and the output is sent to the screen.