Static Variables

Static variables in a Function:

When a variable is declared as static, space for **it gets allocated for the lifetime of the program**. Even if the function is called multiple times, space for the static variable is allocated only once and the value of variable in the previous call gets carried through the next function call. It will be initialized once in the first call to the function.

Static variables in a class:

As the variables declared as static are initialized only once as they are allocated space in separate static storage so, the static variables **in a class are shared by the objects.** There can not be multiple copies of same static variables for different objects. Also because of this reason static variables can not be initialized using constructors.

A static variable inside a class should be initialized explicitly by the user using the class name and scope resolution operator outside the class.

Class objects as static:

Just like variables, objects also when declared as static have a scope till the lifetime of program. Consider the below program where the object is non-static.

Static functions in a class:

Just like the static data members or static variables inside the class, static member functions also does not depend on object of class. We are allowed to invoke a static member function using the object and the '.' operator but it is recommended to invoke the static members using the class name and the scope resolution operator.

Static member functions are allowed to access only the static data members or other static member functions, they can not access the non-static data members or member functions of the class.