Insertion Operator - FAQ

:: not required for friend functions?

No. It is not required.

friend functions are global functions.

Friend functions

if are using two or more objects in the parameter. Then we have to make use of friend

when both are of same class then we have 2 options.

1. make operator as a member of class

2. make it as friend

when 2 arguments are from different classes then there is only 1 option that is friend function

Insertion Operator Overloading

- Prototype of a function is standard.

Prototype:

- **ostream & operator<<(ostream &os, Complex &c);**
- It is a global function.
- It must be declared as fired inside class Complex.
- **It must take 2 parameters.**
    1. ostream
    2. Object of your class.
- ostream parameter must be reference
- Return type should be reference of ostream

How cout works?

- If Complex comp; is a complex object, then
- cout<<comp;
- Here operator<< is called and 2 parameters are passed

  1. cout
  2. comp

Why return by reference?

- usually if you write cout<<x<<y; then x is printed using cout and after printing x it should become cout<<y, means y should also get cout.
- Similarly if we write

cout<<comp<<endl;

- Operator<< is called by passing cout and comp then it returns ostream by reference.
- Result of return by reference comes on left side so it becomes
- **cout<<endl;** Then endl is printed.