Subject Name: **Object Oriented Programming Using C++**

Subject Code: **BCA-301 N**

Subject Topic: **File Streams and Namespaces**

Abhishek Dwivedi

Assistant Professor

Department of Computer Application

UIET, CSJM University, Kanpur

C++ Files and Streams

- In C++ programming we are using the **iostream** standard library, it provides **cin** and **cout** methods for reading from input and writing to output respectively.
- To read and write from a file we are using the standard C++ library called **fstream**. Let us see the data types define in fstream library is:

Data Type	Description
fstream	It is used to create files, write information to files, and read information from files.
ifstream	It is used to read information from files.
ofstream	It is used to create files and write information to the files.

File Stream example: writing to a file

```
#include <iostream>
#include <fstream>
using namespace std;
int main () {
  ofstream filestream("testout.txt");
  if (filestream.is_open())
  {
    filestream << "Welcome to java.\n";
    filestream << "C++ Tutorial.\n";
    filestream.close();
  }
  else cout <<"File opening is fail.";
  return 0;
}
```

The content of a text file **testout.txt** is set with the data:

- Welcome to java.
- C++ Tutorial.

File Stream example: reading from a file

```
#include <iostream>
#include <fstream>
using namespace std;
int main () {
  string srg;
  ifstream filestream("testout.txt");
  if (filestream.is_open())
  {
    while ( getline (filestream,srg) )
    {
      cout << srg <<endl;
    }
    filestream.close();
  }
  else {
      cout << "File opening is fail."<<endl;
    }
  return 0;
}
```

C++ getline() function

- The cin is an object which is used to take input from the user but does not allow to take the input in multiple lines. To accept the multiple lines, we use the getline() function. It is a pre-defined function defined in a **<string.h>** header file used to accept a line or a string from the input stream until the delimiting character is encountered.

getline(istream &is, string &str);

Where,

- **is:** It is an object of the istream class that defines from where to read the input stream.
- **str:** It is a string object in which string is stored.

- Example where we take the user input using getline() function.

```
#include <iostream>
#include<string.h>
void main()
{
string name;                          // variable declaration.
cout << "Enter your name :" << endl;
getline(cin,name);          // implementing a getline() function
cout<<"\nHello "<<name;
getch();
}
```

C++ Namespaces

- Namespaces in C++ are used to organize too many classes so that it can be easy to handle the application.
- For accessing the class of a namespace, we need to use namespacename::classname. We can use **using** keyword so that we don't have to use complete name all the time.
- In C++, global namespace is the root namespace. The global::std will always refer to the namespace "std" of C++ Framework.

C++ namespace Example

```
#include <iostream>
using namespace std;
namespace  First
{
   void sayHello()
  {
      cout<<"Hello First Namespace"<<endl;
   }
}
namespace Second
{
     void sayHello()
    {
        cout<<"Hello Second Namespace"<<endl;
     }
}
int main()
{
 First::sayHello();
 Second::sayHello();
return 0;
}
```

C++ namespace example: by using keyword

- This is another example of namespace where we are using "using" keyword so that we don't have to use complete name for accessing a namespace program.

```
#include <iostream>
using namespace std;
namespace First{
  void sayHello(){
    cout << "Hello First Namespace" << endl;
  }
}
namespace Second{
  void sayHello(){
    cout << "Hello Second Namespace" << endl;
  }
}
using namespace First;
int main () {
  sayHello();
  return 0;
}
```

References:

- www.studytonight.com
- www.tutorialpoint.com
- www.geeksforgeeks.org
- “Object oriented programming in C++” Robert Lafore
- “Object oriented programming with C++”, E.Balagurusamy