Don Bosco Institute of Technology, Kurla(W)
Department of Electronics and Tele-Communication Engineering
ECL304 - Skill Lab: C++ and Java Programming
Sem III
2021-22

Lab Number:	**7**
Student Name:	Rakshita Rajeev Khantwal
Roll No :	26

Title:

1. To write a program to demonstrate friend function in C++.
2. To write a program to demonstrate friend class in C++.

Learning Objective:

- Students will be able to implement friend function and friend classes in C++.

Learning Outcome:

- To understand how to use the private members using friend function and friend class.

Course Outcome:

ECL304.6	Percept the Utility and applicability of OOP

Theory:

- Explain in details about access specifiers: public, private and protected.

Public, protected, and private inheritance have the following features:

1. public inheritance makes public members of the base class public in the derived class, and the protected members of the base class remain protected in the derived class.

2. protected inheritance makes the public and protected members of the base class protected in the derived class.

3. private inheritance makes the public and protected members of the base class private in the derived class. Private members of the base class are inaccessible to the derived class.

- Explain about friend function and friend classes in C++.

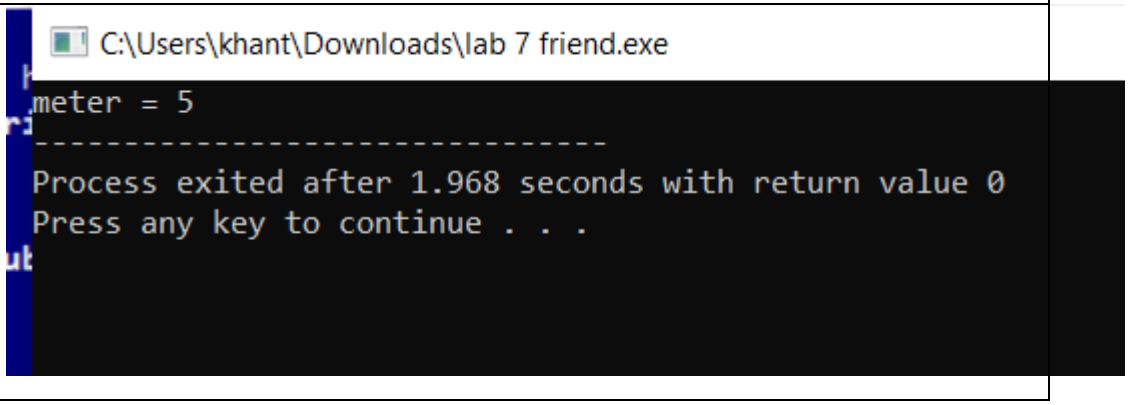
1. Friend Class A friend class can access private and protected members of other class in which it is declared as friend. It is sometimes useful to allow a particular class to access private members of other class.
2. Friend Function Like friend class, a friend function can be given a special grant to access private and protected members. A friend function can be:

a) A member of another class
b) A global function

Algorithm:	STEP 1: Start STEP 2:Create class height STEP 3:Define attributes and friend function STEP 4:Mention friend function and its attributes STEP 5:Create object in main function STEP 6:Display output STEP 7:Step

Program:	//Program to demonstrate working of friend function.
	
	#include <iostream>
	using namespace std;
	
	class height{
	private:
	int meter;
	
	public:
	height() {
	meter = 0;
	}
	
	friend int addFive(height); //friend function
	
	};
	
	int addFive(height h) {

	h.meter =h.meter +5; //accessing private members
	return h.meter;
	}
	
	int main() {
	height h;
	cout << "meter = "<<addFive(h);
	
	return 0;
	}

Input given:	-
Output Screenshot:	

Algorithm :	Step 1: start Step 2:create class one

	Step 3:define attributes and mention friend class 'class 2' Step 4:create class two Step 5:Define attributes and method void showone(one &x) Step 6:Create object in main function Step 7:Display output Step 8: Stop

Program:	#include <iostream>
	class one {
	private:
	int a;
	
	public:
	one() { a = 26; }
	friend class two; // Friend Class
	};
	
	class two {
	private:
	int b;
	
	public:
	void showone (one &x)
	{
	std::cout << "one 'a' = " << x.a;
	}
	};
	
	int main()

	{
	one a;
	two b;
	b.showone(a);
	return 0;
	}

Input given:	-
Output Screenshot: