

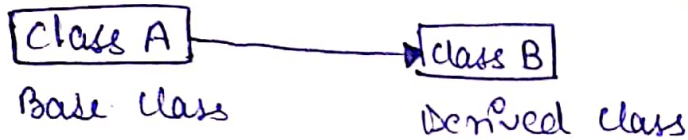
SDF-II
Tutorial-12

Shubham Garg
9919103057
Batch: F2

Q. The Various Types of Inheritance are →

(i) Single Inheritance

In this type, a class is allowed to inherit from only one class i.e. one sub class is inherited by one base class only.

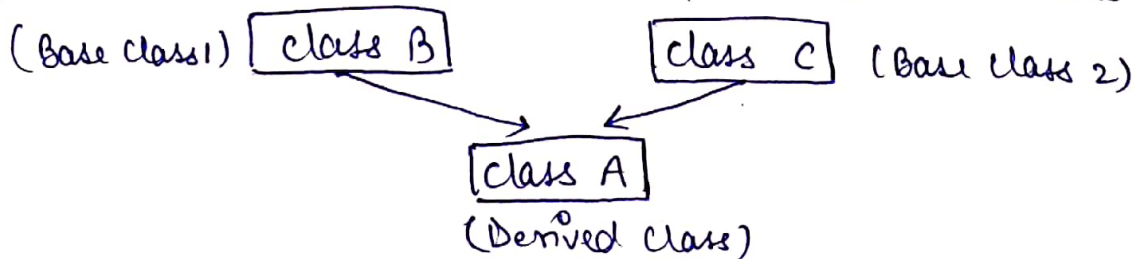


Syntax:

```
class subclass_name: access_mode  
base_class  
{  
    // body of subclass  
};
```

(ii) Multiple Inheritance

In this type, a class can inherit from more than one classes i.e. one sub class is inherited from more than one base classes.



Syntax:

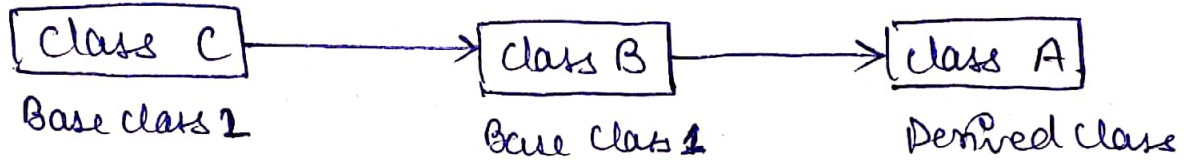
```
class subclass_name:  
access_mode base_class1,  
access_mode base_class2,  
.....  
{  
    // body of subclass  
};
```

Note

Here, no. of base classes will be separated by comma (,) and access mode for every base class must be specified.

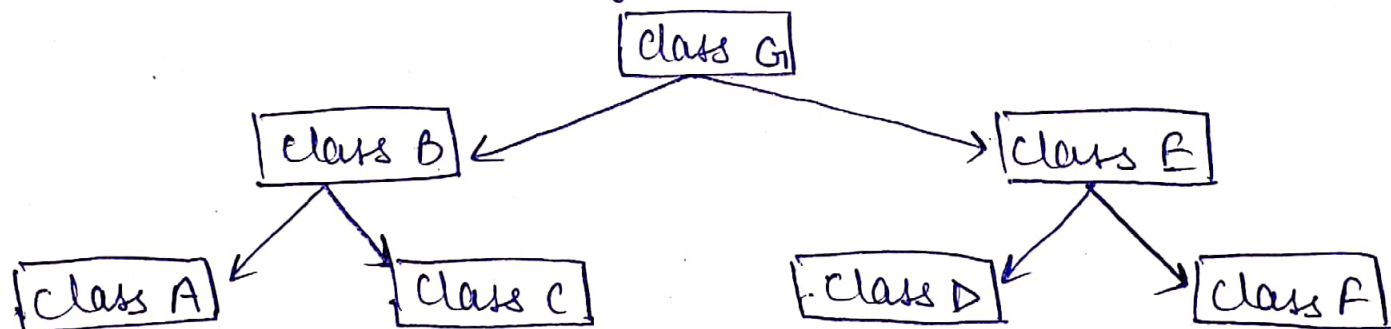
3. Multilevel Inheritance

In this type, a derived class is created from another derived class.



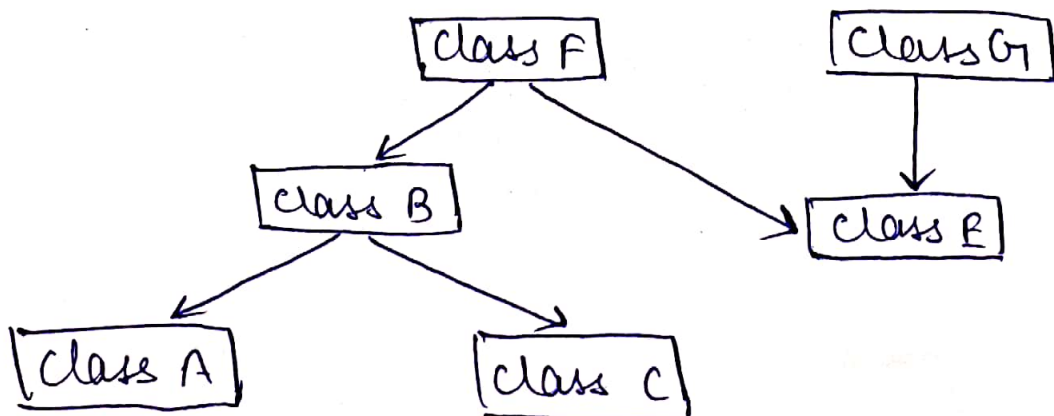
4. Hierarchical Inheritance

In this type, more than one sub class is inherited from a single base class i.e. more than one derived class is created from a single base class.



5. Hybrid (Virtual) Inheritance

It is implemented by combining more than one type of inheritance. For example! Combining Hierarchical and multiple inheritance.



Shubham Gang 9919103057

Answer-2 Protected

Answer-3 41

Answer-4 No, Base Class and its object do not have any knowledge about any class derived from base class.

Answer-5 Yes, whenever we create derived class object default constructor is executed and then the derived class constructor.

Shubham Garg 9919103057

Q.6 Write a C++ program to display the skills of a person according to his/her profession using inheritance.

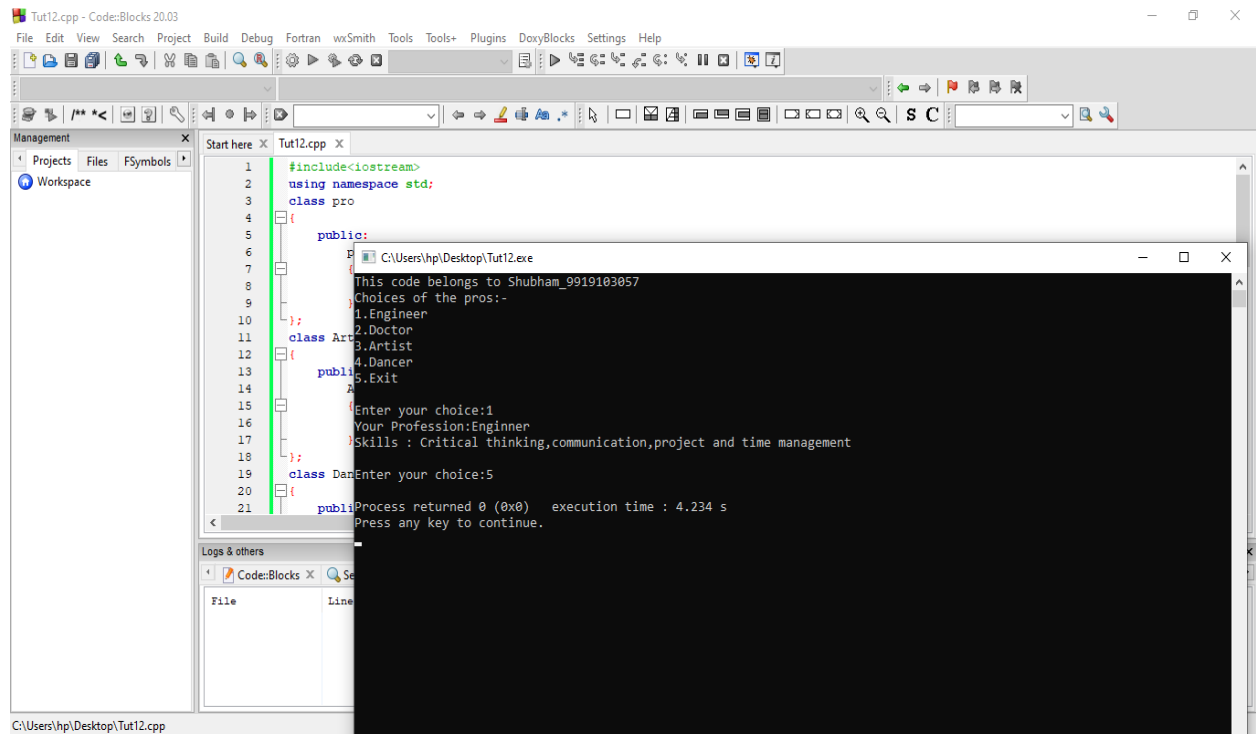
```
#include<iostream>
using namespace std;
class pro
{
    public:
        pro()
        {
            cout<<"Your Profession:";
        }
};
class Artist : public pro
{
    public:
        Artist():pro()
        {
            cout<<"Artist\nSkills : Persistence,Patience,Passion,A sense of adventure and Discipline.";
        }
};
class Dancer : public pro
{
    public:
        Dancer():pro()
        {
            cout<<"Dancer\nSkills : Goal-directed actions that are observable as small units of engagement in daily life occupations";
        }
};
class Engineer : public pro
{
    public:
        Engineer():pro()
        {
            cout<<"Enginner\nSkills : Critical thinking,communication,project and time management";
        }
};
class Doctor : public pro
{
    public:
        Doctor():pro()
        {
```

```

        cout<<"Doctor\nSkills :
Compassion,Understanding,Empathy,Honesty,Competence,Commitment,Humanity and
Courage";
    }
};
int main()
{
    cout<<"This code belongs to Shubham_9919103057\n";
    int ch;
    cout<<"Choices of the pros:-"<<endl;
    cout<<"1.Engineer"<<endl;
    cout<<"2.Doctor"<<endl;
    cout<<"3.Artist"<<endl;
    cout<<"4.Dancer"<<endl;
    cout<<"5.Exit";
    while(1)
    {
        cout<<endl<<endl<<"Enter your choice:";
        cin>>ch;
        if(ch==1)
            Engineer e;
        else if(ch==2)
            Doctor d;
        else if(ch==3)
            Artist a;
        else if(ch==4)
            Dancer d;
        else if(ch==5)
            break;
    }
    return 0;
}

```

Output :



Q.7 Write a C++ program to read and print employee information using multiple inheritance.

```
#include<iostream>
```

```
#include<cstdio>
```

```
using namespace std;
```

```
class a1
```

```
{
```

```
private:
```

```
    string name,address;
```

```
protected:
```

```
    void get()
```

```
{
```

```
    cout<<"Enter Name: "<<endl;
```

```
    fflush(stdin);
```

```
    getline(cin,name);
```

```
    cout<<"Enter Address: "<<endl;
```

```
    fflush(stdin);
```

```
    getline(cin,address);
```

```
}
```

```
void show()
```

```
{
```

```

        cout<<"Name : "<<name<<endl;
        cout<<"Address : "<<address<<endl;
    }
};

class a2
{
private:
    string occ;
    int salary;
protected:
    void get()
    {
        cout<<"Enter Occupation:"<<endl;
        fflush(stdin);
        getline(cin,occ);
        cout<<"Enter salary: "<<endl;
        fflush(stdin);
        cin>>salary;
    }
    void show()
    {
        cout<<"Occupation : "<<occ<<endl;
        cout<<"Salary: "<<salary<<endl;
    }
};

class b:public a1,public a2
{
public:
    b()
    {
        a1::get();
        a2::get();
    }
    void showdata()
    {
        a1::show();
        a2::show();
    }
};

int main()
{
    cout<<"This code belongs to Shubham_9919103057\n";
    b emp;

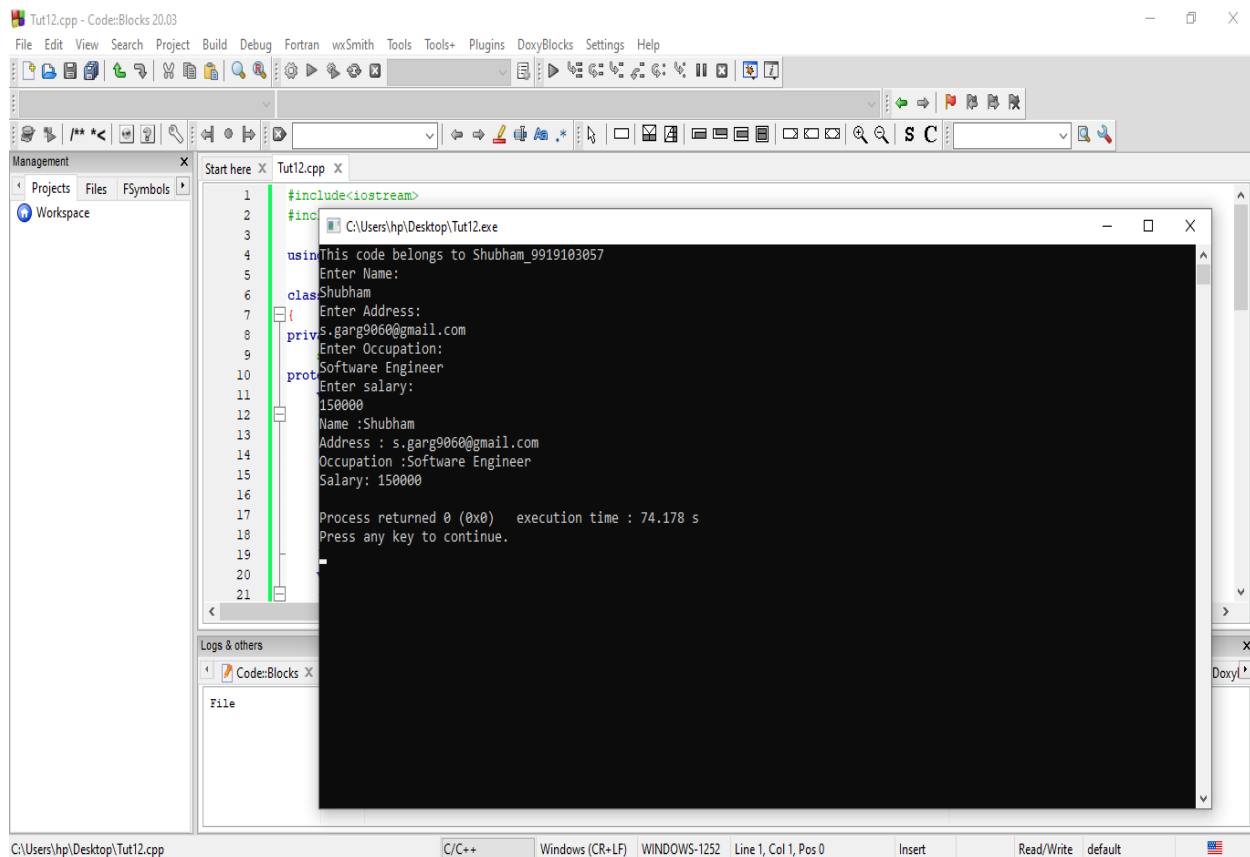
```

```

emp.showdata();
return 0;
}

```

Output :



Q. 8: Write a C++ program to calculate cube, root and square of a number using hierarchical inheritance.

```
#include<iostream>
```

```
#include<math.h>
```

```
using namespace std;
```

```
class operation
```

```
{
```

```
protected:
```

```
int n;
```

```
public:
```



```
void setdata()
{
    cout<<"\nEnter the value:";
    cin>>n;
}

};

class cube:public operation
{
    public:
    void getdata()
    {
        cout<<"Cube:"<<pow(n,3)<<endl;
    }
};

class root:public operation
{
    public:
    void getdata()
    {
        cout<<"Root:"<<pow(n,0.5)<<endl;
    }
};

class square:public operation
{
    public:
    void getdata()
    {
        cout<<"Square:"<<pow(n,2)<<endl;
```

```

    }
};

int main()
{
    cout<<"This code belongs to Shubham_9919103057\n";
    int ch;
    cout<<"Choices:-\n";
    cout<<"1.Cube\n";
    cout<<"2.Root\n";
    cout<<"3.Square\n";
    cout<<"4.Exit\n";
    while(1)
    {
        cout<<"\nEnter Your choice:";
        cin>>ch;
        if(ch==1)
        {
            cube c;
            c.setdata();
            c.getdata();
        }
        else if(ch==2)
        {
            root r;
            r.setdata();
            r.getdata();
        }
        else if(ch==3)

```

```

{
    square s;

    s.setdata();

    s.getdata();

}

else if(ch==4) break;

}

cout<<"Exit!!!";

return 0;

}

```

Output :

The screenshot shows the Code::Blocks IDE with the following components:

- Source Code (Tut12.cpp):**

```

1 #include<iostream>
2 #include<math.h>
3 using namespace std;
4 class opert
5 {
6     protected:
7     inChoices:-
8     public:
9     1.Cube
10    2.Root
11    3.Square
12    4.Exit
13
14    Enter Your choice:1
15
16    };
17
18    class cube
19    {
20    Cube:8000
21
22    public:
23    vo
24
25    {
26    Enter the value:20
27    Cube:8000
28
29    Enter Your choice:2
30
31    vo
32
33    {
34    Enter the value:400
35    Root:20
36
37    }
38
39    Enter Your choice:4
40
41    Exit!!!
42
43    Process returned 0 (0x0)   execution time : 49.537 s
44    Press any key to continue.

```
- Console Output:**

```

Enter Your choice:1
Enter the value:20
Cube:8000
Enter Your choice:2
Enter the value:400
Root:20
Enter Your choice:4
Exit!!!
Process returned 0 (0x0)   execution time : 49.537 s
Press any key to continue.

```
- Status Bar:** C:\Users\hpl\Desktop\Tut12.cpp

Q. 9: Create two classes named Mammals and MarineAnimals. Create another class named BlueWhale which inherits both the above classes. Now, create a function in each of these classes which prints "I am mammal", "I am a marine animal" and "I belong to both the categories: Mammals as well as Marine Animals" respectively. Now, create an object for each of the above

class and try calling

1 - function of Mammals by the object of Mammal

2 - function of MarineAnimal by the object of MarineAnimal

3 - function of BlueWhale by the object of BlueWhale

4 - function of each of its parent by the object of BlueWhale

```
#include<iostream>
```

```
using namespace std;
```

```
class Mammals
```

```
{
```

```
    public:
```

```
        void disp1()
```

```
        {
```

```
            cout<<"I am mammal\n";
```

```
        }
```

```
};
```

```
class MarineAnimal
```

```
{
```

```
    public:
```

```
        void disp2()
```

```
        {
```

```
            cout<<"I am a marine animal\n";
```

```
        }
```

```
};
```

```
class BlueWhale:public Mammals,public MarineAnimal
```

```
{
```

```
    public:
```

```
        void disp3()
```

```
        {
```

```
            cout<<"I belong to both the categories: Mammals as well as Marine Animals\n";
```

```
        }
```

```
};
```

```
int main()
{
    cout<<"This code belongs to Shubham_9919103057\n";
    Mammals Mammal;
    MarineAnimal MarineAnimal;
    BlueWhale BlueWhale;
    cout<<"Calling function of Mammals by the object of Mammal:-\n";
    Mammal.disp1();
    cout<<"\nCalling function of MarineAnimal by the object of MarineAnimal:-\n";
    MarineAnimal.disp2();
    cout<<"\nCalling function of BlueWhale by the object of BlueWhale:-\n";
    BlueWhale.disp3();
    cout<<"\nCalling function of each of its parent by the object of BlueWhale:-\n";
    BlueWhale.disp1();
    BlueWhale.disp2();
    return 0;
}
```

Output :

