**Content 45**

### Pointers to Derived Class in C++

The main thing is that the base class pointer will only point the base class not the other class or derived class.

But If there is an pointer which is pointing inherited class then it will support all the classes derived in it.

**Code in which I had created base class pointer and accessed the methods and function of base class.**

#include <iostream>

using namespace std;

class Base{

    public:

    int var\_base;

    void display(void){

        cout<<"~~~~~~~Base class here~~~~~~~~~"<<endl;

        cout<<"The value of var\_base is: "<<var\_base<<endl;

    }

};

class  Derived: public Base{

    public:

    int var\_derived;

    void display(void){

        cout<<"\n\n ~~~~~~Derived class Here~~~~~~~"<<endl;

        cout<<"The Value of var\_base: "<<var\_base<<endl;

        cout<<"The Value of var\_dervied: "<<var\_derived<<endl;

    }

};

int main()

{

    Base \* base\_pointer;        //Base class pointer will only access base class not the derived class

    Derived derived\_obj;

    base\_pointer = &derived\_obj; //pointer pointing derived class.

    base\_pointer->var\_base=45;

    // base\_pointer->var\_derived=34; //this will throw an error

    base\_pointer->display();

    Derived \*pointer\_derived;

    pointer\_derived = & derived\_obj;

    pointer\_derived->var\_derived=10;

    pointer\_derived->var\_base=7;

    pointer\_derived->display();

    return 0;

}

**Output:**

~~~~~~~Base class here~~~~~~~~~

The value of var\_base is: 45

~~~~~~Derived class Here~~~~~~~

The Value of var\_base: 7

The Value of var\_dervied: 10