一、程序代码

#include<iostream>

using namespace std;

class Base {

public:

void setx(int i)

{

x = i;

}

int getx()

{

return x;

}

public:

int x;

};

class Derived :public Base {

public:

void sety(int i)

{

y = i;

}

int gety()

{

return y;

}

void show()

{

cout << "Base::x=" << x << endl; }

public:

int y;

};

int main()

{

Derived bb;

bb.setx(16);

bb.sety(25);

bb.show();

cout << "Base::x=" << bb.x << endl;

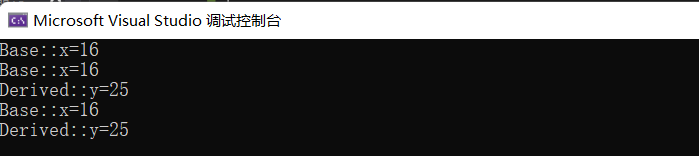
cout << "Derived::y=" << bb.y << endl;

cout << "Base::x=" << bb.getx() << endl;

cout << "Derived::y=" << bb.gety() << endl;

return 0; }

二、程序结果



三、感悟心得

1. 28和38行无法访问x，因为x的访问权限改为private时，派生类也无法直接继承和访问基类的x。

2. 38行的bb.x出现错误，因为x的访问权限改为protected时，派生类的对象无法直接访问x。

3. 35行的bb.setx()、38行的bb.x、40行的bb.gety()出现错误，因为继承方式变为私有继承后，对象无法直接访问类中的私有函数及变量；

4. 35行的bb.setx()、38行的bb.x、40行的bb.gety()出现错误，因为继承方式变为保护继承后，对象无法直接访问基类中的保护函数及变量；

Copyright ©2021-2099 ZhongkaiCui. All rights reserved