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;

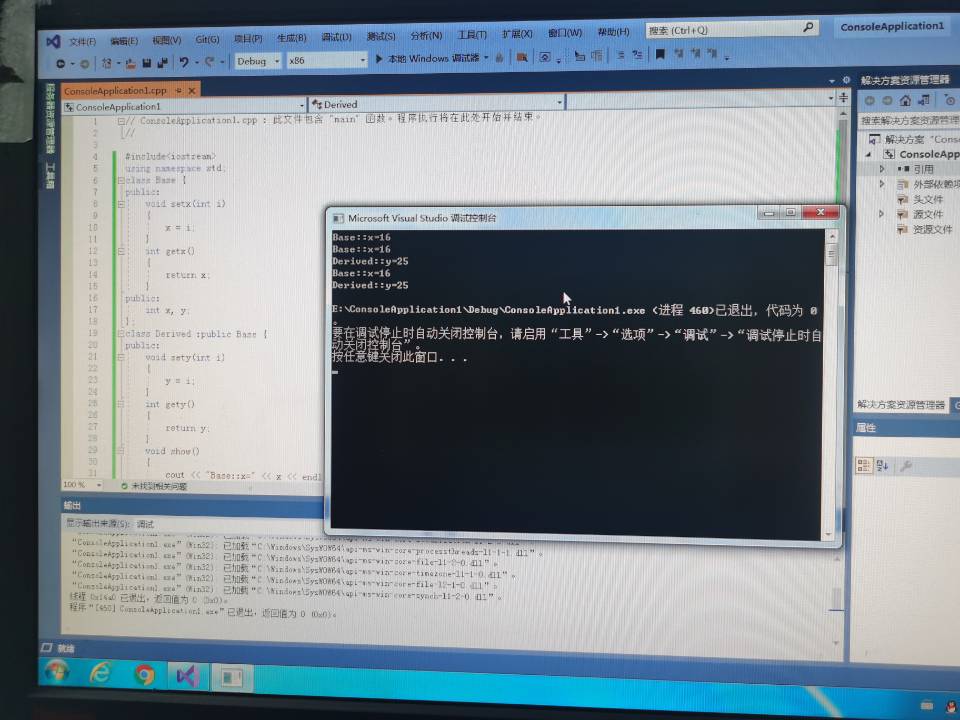
cout<<"Derived::y="<<bb.y<<endl;

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

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

return 0;

}



感悟

通过本次实验，我掌握了派生类的声明方法和派生类构造函数的定义方法

也掌握了不同方式下，基类成员在派生类中的访问属性