源代码：

#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 << "Baxex::x" << x << endl; // 语句1

}

public:

int y;

};

int main()

{

Derived bb; // 语句2

bb.setx(16); // 语句3

bb.sety(25); // 语句4

bb.show(); // 语句5

cout << "Base.:x=" << bb.x << endl; // 语句6

cout << "Derived:y=" << bb.y << endl; // 语句7

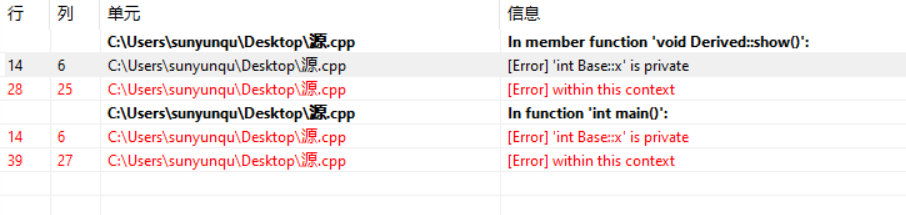
cout << "Base:x=" << bb.getx() << endl; // 语句8

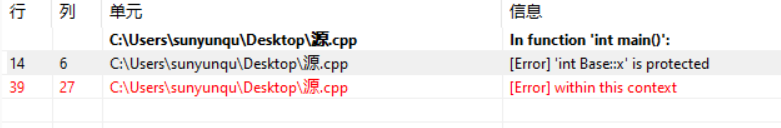
cout << "Derived:y=" << bb.gety() << endl; // 语句9

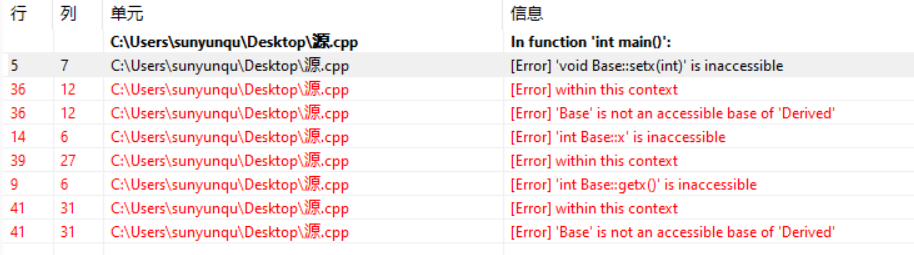
return 0;

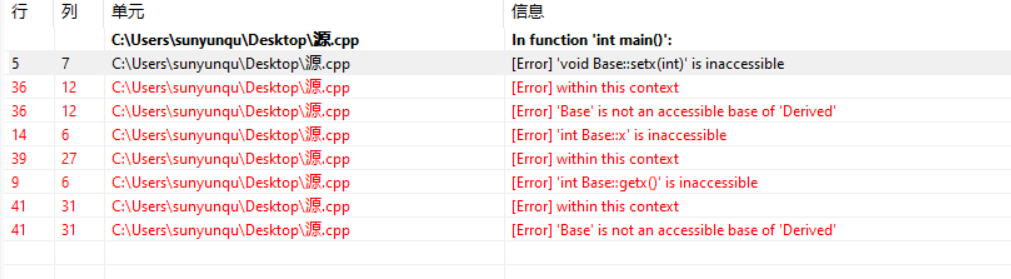
}

运行结果：









心得体会：

本次实验使我充分理解了派生类与基类之间的继承关系，掌握了在不同的继承方式下，对基类成员的访问权限也不同，因此在编程时应极为注意派生类的继承方式，避免出现不能访问的错误。

Copyright ©2021-2099 YunquSun.All rights reserved