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; // 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

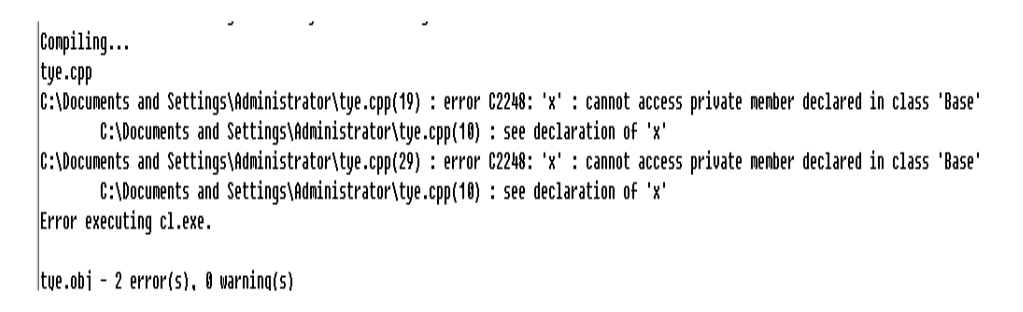
cout<<"Derived::y="<<bb.gety()<<endl; // 语句9

return 0;

}

程序结果

(1)



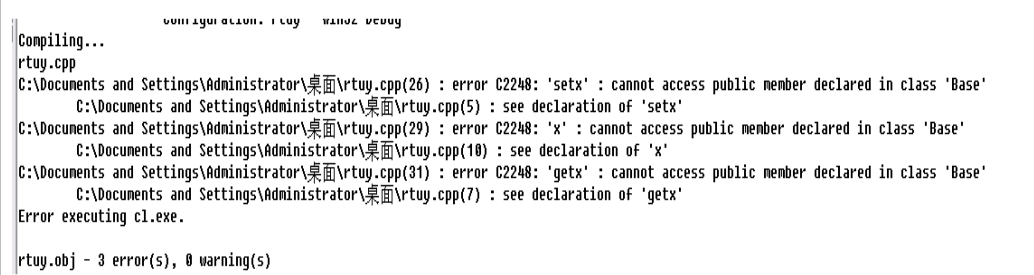
因为基类 Base 中数据成员 x 的访问权限改为 private 时,基类的私有成员在派生类中并没有成为派生类的私有成员,使得再通过派生类的对象调用 x 时,不能被系统所识别,出现了错误

(2)



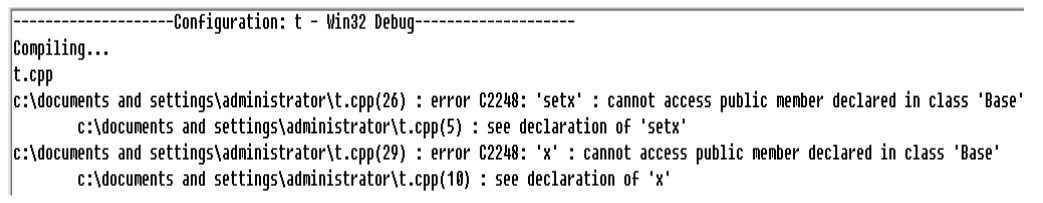
因为某类Base中数据成员 x 的访问权限改为 proteted 时,公用基类的保护成员在公用派生类中的访问属性仍然是保护,在公用派生类外就不能通过公用派生类的对象访问该成员,所以出现了错误。

(3)



将派生类 Derived 的継承方式改为 private 时,私有基类中的公用成员在私有派生类中的访问属性为私有,在类外不能通过对象调用私有成员,只能通过成员函数调用

(4)



因为私有继承时基类中的公有函数到派生类中一是私有成员,不能通过对象访问,只能通过成员函数来访问

感想心得

了解了派生类的声明方法和派生类构造函数的定义方法，以及不同方式下，基类成员在派生类中的访问属性，静态成员函数一般不访问普通数据成员，它的作用主要是访问和操作同类中的静态数据成员或全局变量。

一个类的静态成员函数与非静态成员函数不同，它不需要创建任何该类的对象就可以被调用。

Copyright IMG_2562021-2099 XiaohanYang. All rights reserved.