```
#include<iostream>
#include<string>

using namespace std;

//成员函数做友元

class Building;

class GoodGay{
public:
    Building *building;

    GoodGay();

    void visit(); //让visit函数可以访问Building中的私有成员
    void visit2(); //让visit2函数不可以访问Building中私有成员


};


class Building{
    //告诉编译器  GoodGay类下的visit成员函数作为本类的好朋友可以访问私有成员
    friend void GoodGay::visit();

public:
    string m_SettingRoom;
private:
    string m_BedRoom;

public:
    Building(){
        this->m_SettingRoom="客厅";
        this->m_BedRoom="卧室";
    }
};


GoodGay::GoodGay(){
    //创建建筑物对象
    building =new Building;
}

void GoodGay::visit(){
    cout<<"visit 函数正在访问:"<<building->m_SettingRoom<<endl;
    cout<<"visit 函数正在访问:"<<building->m_BedRoom<<endl;
}

void GoodGay::visit2(){
    cout<<"visit2 函数正在访问:"<<building->m_SettingRoom<<endl;
}








void test01(){
    GoodGay goodGay;
    goodGay.visit();
    goodGay.visit2();
}


int main(){
    test01();
}
```