

二级C类与对象：静态成员和友员 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/134/2021_2022__E4_BA_8C_E7_BA_A7C___E7_c97_134470.htm

创建静态成员主要是解决数据共享问题，而创建友员的目的是为了某些函数访问类中的私有成员。静态对象独立于具体的类对象而存在，是从属于类的属性，不是属于某个对象的。静态成员变量一般用于：
用来保存该类对象的个数。 作为一个标志，指示一个特定的动作是否发生，如：可能创建几个对象，每个对象要对某个磁盘文件进行写操作，但显然在同一时间里只允许一个对象写文件，在这种情况下，用户希望说明一个静态数据成员指出文件何时正在使用，何时处于空闲状态。 用来保存指向链表第一成员或最后一个成员的指针。 用来管理该类的各个对象共享的一些资源，如各个对象都从同一空间分配，或者各个对象使用同一文件资源等。 静态成员函数则用来对这些静态成员数据操作。如返回对象个数，查找链表中的某个节点，或者判断某个共享文件是否被某个对象使用等。 静态成员函数与一般成员函数的区别：
(1) 定义方式不同：静态成员定义时前面加修饰符 `static`. 而非静态成员不用。
(2) 使用限制不同：静态成员函数只能访问静态成员变量。而非静态成员函数既可以访问静态数据成员也可以访问非静态数据成员。
(3) 调用方式不同：静态成员函数的调用格式为：`classname::static Func()`, 而非静态成员函数的调用须通过具体的对象调用，格式为：`Object.func()`。
(4) 静态成员函数没有 `this` 指针，所以它不能访问访问某个具体对象的数据成员，即静态成员函数不能访问非静态数据成员。 1

、静态数据成员：（1）静态数据成员在定义或说明时前面加关键字 static。例：..... Private: Int a,b,c. Static int s.（2）静态成员的初始化与以非数据成员的初始化不同。静态数据成员一般在类定义外部按特定的格式进行初始化。其格式为：::=. 例：class Nclass { private: static int a. }. int Nclass::a=5.（3）引用静态数据成员时，采用如下格式：:: 如果静态数据成员的访问权限允许的话（即 public 成员），可在程序中，按上述格式来引用静态数据成员。例 1：#include class Myclass { public: Myclass(int a,int b,int c). Void GetNumber(). Void GetSum(). Private: Int A,B,C. Static int Sum. }. int Myclass::Sum=0. Myclass::Myclass(int a,int b,int c) { A=a. B=b. C=c. Sum =A B C. } void Myclass::GetNumber() { cout} void Myclass::GetSum() { cout} void main() { Myclass M(1,2,3),N(4,5,6). M.GetNumber(). N.GetNumber(). M.GetSum(). N.GetSum(). } 运行结果：Number=1,2,3 Number=4,5,6 Sum=21 Sum=21 这里，M 和 N 两个对象的 Sum 值相等。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com