

# 类的常量成员

## 常量数据成员

- 类的常量数据成员是被声明为const类型的数据成员。常量数据成员与一般的符号常量不同，在声明时不能被赋值，只能在定义对象时，通过构造函数的成员初始化列表的方式来获得初值，而且一旦一个对象被创建，其常量数据成员的值就不许再被修改。

```
class A
{
private:
    const int a;//私有常量数据成员
public:
    const int b;//公有常量数据成员
    int c;
    A(int x,int y,int z):a(x),b(y)
    {
        c=z;
    }
    .....
}
```

在类外定义A类对象时，需要：

```
A obj(10,20,30);
```

obj.a和obj.b的值分别是10和20，而且

不允许改变

## 常量成员函数

- 类的常量成员函数是被声明为const类型的成员函数。常量成员函数只有权读取对象的数据成员，但无权修改对象数据成员的值。
- 类的常量成员函数的声明形式如下：
- <类型说明符> <函数名>(<参数表>)const;
- 修饰符const要放在函数声明的尾部。在类外定义函数时，也要加上const关键字。

```
class B
```

```
{
```

```
public:
```

```
    int ReadA() const
```

```
    {    return a; }
```

```
    void WriteA(int x)
```

```
    {    a=x;    }
```

```
private:
```

```
    int a;
```

```
}
```

如果将上面的成员函数Write也声明为常量成员函数：

```
void WriteA(int x) const  
  
{    a=x;    }
```

由于WriteA函数内部对类的数据成员a进行了修改，所以，编译器在检查时就会报错。

提示：当成员函数的函数体较大、比较复杂且不需要修改数据成员的值时，通常将其定义为常量成员函数，让系统帮助避免该函数对对象数据成员的修改。