
C++

类与对象

创客学院 荣荣老师

图示: (非主干知识想到, 或看笔记资料, 网上搜).

1 / 类的申明

2 / 成员函数

3 / 常成员、常对象

4 / 静态成员

5 / 友元

数据 函数 对象

↑
↓

数据、函数 能上, 类可用

↑
静态、
↑

`A(const A &x)`

`~A()`

`A(x)`

`A()`

`protected`

`public`

`private`

类名

`class C {`

↑

类 成员 函数 函数

↑
友元

→

类的申明

```
class 类名
{
    private:
        私有的数据和成员函数;
    public:
        公用的数据和成员函数;
    protected:
        保护的数据和成员函数
};
```

结构体?

类的成员函数(类内/类外实现)

- 构造函数（类的初始化，创建类时自动调用；初始化表,this指针）

 - 默认构造函数 `A();`

 - 拷贝构造函数 `A(const A &x);` *//深浅拷贝问题*

 - 赋值构造函数 `operator=;`

- 析构函数（类的解构，销毁类时自动调用）

 - `~A();`

- 普通成员函数

常成员、常对象（C++推荐const而不用#define, mutable）

!!! *const* 数据成员只在某个对象生存期内是常量，而对于整个类而言却是可变的(*static* 例外)。

■ 常数据成员（构造函数初始化表赋值）

```
class A{  
public: A():x(100) { }  
    const int x;  
}
```

■ 常成员函数

```
void func() const;
```

■ 常对象

```
const A a;
```

静态成员(属于类不属于对象)

静态成员的申明

`static int x;`

`static const int x = 10;`

静态数据成员初始化

类外: `static int A::x = 10;`

静态成员函数

`static void func();` //能访问静态成员

调用方法 `A::func();`

友元(破坏封装)

- 友元类

```
friend class B;
```

- 友元函数

```
friend void func();
```

- 友元成员函数

```
friend void B::func();
```

扫一扫，获取更多信息



THANK YOU