

# 纯虚函数和抽象类

- 前面定义的类都是“具体类”，能够使用这些类定义对象。C++还提供了一种“抽象类”，抽象类不能实例化对象，它的唯一用途是为其他类提供合适的基类，其他类可从它这里继承和（或）实现接口。
- 一个类是抽象类还是具体类，主要看其是否包含纯虚函数，包含纯虚函数的类就是抽象类。纯虚函数就是在声明时初始化为0、没有函数体的虚函数，其声明形式为：

`virtual <函数类型> 纯虚函数名(<形参类型表>) = 0;`

- 例如，我们可以定义Base类：

```
class Base
{
public:
    virtual void fun()=0    // 纯虚函数
};
```

上面定义的Base类中包含纯虚函数，因此，Base类是一个抽象类。如果使用Base类创建对象，则编译程序时会报错。例如：

```
Base b; // 错误：不能定义抽象类的对象
```

- 再如，我们基于Base类创建两个派生类Derived1和Derived2：

```
class Derived1 : public Base
{
};
class Derived2 : public Base
{
public:
    void fun()          // 等价于virtual void fun()
    {
        cout<<"Derived2类的fun()函数被调用！"<<endl;
    }
};
```

则：

```
Derived1 d1;    // 错误
Derived2 d2;    // 正确
```

一个类层次结构中可以不包含任何抽象类，但是很多良好的面向对象的系统，其类层次结构的顶部是一个抽象基类。在有些情况中，类层次结构顶部有好几层都是抽象类。

