析构虚函数

子类继承了父类，但是普通析构函数是不会调用子类的析构的，所以可能会导致内存释放地不干净。我们需要使用虚析构函数来解决。

写法：

Virtual ~Animal(){}

虚析构函数是为了解决[基类](http://baike.baidu.com/view/535539.htm)的[指针](http://baike.baidu.com/view/159417.htm)指向派生类对象，并用基类的指针删除派生类对象。

纯虚析构函数

写法：

声明：Virtual ~Animal() = 0

实现：Animal::~Animal(){…}

纯虚析构，需要声明，也需要实现。类内声明，类外实现。

若类中有纯虚函数，则类也是抽象类，不可实例化。

代码：  
class People**{**

public**:**

People**(){**

cout **<<** "构造函数 People!" **<<** endl**;**

**}**

virtual void showName**()** **=** 0**;**

virtual **~**People**(){**

cout **<<** "析构函数 People!" **<<** endl**;**

**}**

**};**

class Worker **:** public People**{**

public**:**

Worker**(){**

cout **<<** "构造函数 Worker!" **<<** endl**;**

pName **=** **new** char**[**10**];**

**}**

virtual void showName**(){**

cout **<<** "打印子类的名字!" **<<** endl**;**

**}**

**~**Worker**(){**

cout **<<** "析构函数 Worker!" **<<** endl**;**

**if** **(**pName **!=** **NULL){**

**delete** pName**;**

**}**

**}**

private**:**

char**\*** pName**;**

**};**

void test**(){**

People**\*** people **=** **new** Worker**;**

people**->~**People**();**

**}**