在学习完第三章后，主要了解了什么是类与对象，知道了类与对象的基本概念，了解了构造函数与析构函数，对象数组与对象指针，以及静态成员，友元等。类是用来抽象一批对象的，类是一个抽象的概念，描述事物的共性（属性和数据）。在面向对象编程中，类的具体实例就叫做对象。构造函数用于创造一个对象，析构函数则用于销毁一个对象，c++规定每个类都必须有构造函数和析构函数。（未定义构造函数、析构函数按默认的）。析构函数和构造函数必须遵守的一些规则：构造函数和析构函数均没有返回值。一个类的构造函数可以有多个，且参数是任意的，但析构函数只能有一个，且不能带有参数的。（构造函数一般是用户成员变量的初始化）。对象数组是指每一个数组元素都是对象的数组，也就是说，若一个类有若干个对象，我们把一系列的对象用一个数组来存放。对象数组的元素是对象，不仅具有数据成员，而且还有函数成员。而对象指针就是用于存放对象地址的变量。当将类的某个数据成员声明为static时，该静态数据成员只能被定义一次，而且要被同类的所有对象共享。各个对象都拥有类中每一个普通数据成员的副本，但静态数据成员只有一个实例存在，与定义了多少类对象无关。 静态方法就是与该类相关的，是类的一种行为，而不是与该类的实例对象相关。 静态数据成员的用途之一是统计有多少个对象实际存在。 静态数据成员不能在类中初始化，实际上类定义只是在描述对象的蓝图，在其中指定初值是不允许的。也不能在类的构造函数中初始化该成员，因为静态数据成员为类的各个对象共享，否则每次创建一个类的对象则静态数据成员都要被重新初始化。静态成员不可在类体内进行赋值，因为它是被所有该类的对象所共享的。你在一个对象里给它赋值，其他对象里的该成员也会发生变化。为了避免混乱，所以不可在类体内进行赋值。静态成员的值对所有的对象是一样的。静态成员可以被初始化，但只能在类体外进行初始化。通过对这一章的让我对c++这门编程语言又有了更深的了解。

构造函数只是起初始化值的作用，但实例化一个对象的时候，可以通过实例去传递参数，从主函数传递到其他的函数里面，这样就使其他的函数里面有值了。规则，只要你一实例化对象，系统自动回调用一个构造函数，就是你不写，编译器也自动调用一次。 析构函数与构造函数的作用相反，用于撤销对象的一些特殊任务处理，可以是释放对象分配的内存空间；特点：析构函数与构造函数同名，但该函数前面加~。 析构函数没有参数，也没有返回值，而且不能重载，在一个类中只能有一个析构函数。 当撤销对象时，编译器也会自动调用析构函数。 每一个类必须有一个析构函数，用户可以自定义析构函数，也可以是编译器自动生成默认的析构函数。一般析构函数定义为类的公有成员。构函数:构造函数是在对象实例化时自动执行，而析构函数在对象被销毁时自动执行。

构造函数:当类被实例化时，会自动执行的函数

构造函数有什么作用:

我们在使用类的过程中，有时需要立即为实例化对象的多个字段父子，如果通过手工，会带来很多不可预测的问题，而如果在对象创建过程中自动执行，则会带来很多方便。

析构函数有什么作用:

默认情况下，php仅释放对象属性所占用的内存，并不销毁对象相关的资源，而利用析构函数在使用一个对象之后执行代码来清除内存，并将对象从内存中销毁

面向对象的特征有哪些？

封装、继承、多态

所谓封装，也就是把客观事物封装成抽象的类，并且类可以把自己的数据和方法只让可信的类或者对象操作，对不可信的进行信息隐藏。所谓继承是指可以让某个类型的对象获得另一个类型的对象的属性的方法,它支持按级分类的概念。所谓多态就是指一个类实例的相同方法在不同情形有不同表现形式。