通常而言，类是抽象的模板，通常包含其对应的所有对象共有的一组属性或方法。

而由类所实例化的对象都共享类的属性和行为，但是不同对象的属性的值通常是不同的。

但是在一些编程语言的设计思想认为，编程语言中的一切皆对象，例如在python中，函数，类，类型等都是对象。Python中认为class类是由元类产生的对象。

根据类似的思想，我们的确可以将类视为对象。首先类通常包含3个重要部分，分别为类名（class name）/类体名称空间：类包含的成员，方法等（class\_dict）/继承关系：父类们（class\_bases）。我们可以认为这三个部分为所有类共有的属性，而不同的类会有不同的属性值，即不同的类名，不同的成员，以及继承关系。因此在创建类的时候事实上就是利用元类实例化了一个“类“对象，并未这个“类”对象赋予类名，继承关系等属性。

C++ 原始不支持元编程，但是通过模板类来实现

<https://blog.csdn.net/banqingzi/article/details/21389367>

<https://blog.csdn.net/banqingzi/article/details/21404201>

<https://blog.csdn.net/banqingzi/article/details/21601313>

<https://blog.csdn.net/banqingzi/article/details/21648779>

<https://blog.csdn.net/banqingzi/article/details/21709803>

<https://blog.csdn.net/banqingzi/article/details/21875797>

<https://blog.csdn.net/banqingzi/article/details/21966097>

<https://blog.csdn.net/banqingzi/article/details/22105575>

<https://blog.csdn.net/banqingzi/article/details/22391117>