<https://www.w3cschool.cn/python/python-object.html>

**一、面向对象**

1. 三大基本特性
   * 封装：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。
   * 继承：它可以使用现有类的所有功能，并在无需重新编写原来的类的情况下对这些功能进行扩展。
   * 多态：虽然针对不同对象的具体操作不同，但通过一个公共的类，它们（那些操作）可以通过相同的方式予以调用。
2. 五大基本原则
   * 单一职责原则：一个类的功能要单一
   * 开放封闭原则：扩展性方面是开放的而在更改性方面是封闭的，就是说在不修改原模块的功能代码的前提下，增加新模块的功能，这就要求设计之初要将公共部分抽象出来便于扩展。
   * 替换原则：子类应当可以替换父类并出现在父类能够出现的任何地方，也就是说父类的功能要包含在子类功能里。
   * 依赖原则：假设B是较A低的模块，但B需要使用A的功能，这时不要直接使用A中的具体类，而应该由B定义一个抽象接口，并由A来实现抽象接口，B只使用这个抽象接口，也就达到了依赖倒置的目的。
   * 接口分离原则：模块间要通过抽象接口隔离开，而不是通过具体的类强耦合起来
3. 类：用来描述具有相同的属性和方法的对象的集合，对象是类的实例。
4. 类变量：定义在类中且在函数体之外，通常不作为实例变量使用。
5. 数据成员：类变量或实例变量
6. 方法重载：如果父类继承的方法不能满足子类的需求，可以对其进行改写，这个过程叫方法重载。
7. 实例变量：定义在方法中的变量，只作用于当前实例的类
8. 对象：通过类定义的数据结构实例，包括两个数据成员（类变量和实例变量）和方法。