1、属性：通过方法访问和操作字段，一方面可以保护字段，同时还可以像访问公共字段一样访问数据。

2、OOP：Object-oriented programming

3、创建OOP

4、纯php代码文件结尾不加?>免得被其他问价包含是报错。

5、\_\_set() and \_\_get() can execute是因为程序已经在类内了，而类内可以调用私有的方法。

6、常量调用 类::常量

**继承：**

1、通过重写调用父类方法，语法格式：父类名::方()或者parent::方法()调用。

2、final关键子防止被其他类继承，建议只要是单独的类都用final声明（保证不会被其他类继承）。

3、抽象类：

1、抽象类不能被实例化。

2、抽象方法必须被子类方法重写。

4、接口

1、定义：接口定义了实现某种服务的一般规范，声明了所需的函数和常量，但是不指定如何实现。

关键是建立必须实现的一般规则，只有满足了这些规则才能说是实现了这个接口。

2、规则：a、类全部为抽象方法。b、接口抽象方法必须是public。c、成员必须是常量。

多态

5、多态

定义：多态是指OOP能够根据使用类的上下文来重新定义或改变类的性质或者行为，或者说接口的多种不同的实现方式即为多态。把不同的子类对象都看作父类，可以屏蔽不同对象之间的差异，写出通用的代码。

6、面向对象的工具

OOP魔术方法

1、\_\_autoload()

2、\_\_call()

3、\_\_toString()

4、\_\_clone()

5、class\_exists();

6、get\_class();

7.get\_class\_methods();

8.get\_class\_vars();

9.get\_parents\_class();

10.interface\_exists();

11.is\_a(object,class\_name);

12.is\_subclass\_of(object,class\_name);

13.method\_exists(object,method\_name);

OOP反射API

1.PHP5的类和对象函数并没有告诉我们类内部的所有一切，而只是报告了它们的公共成员。要充分了解类，需要指导其所有的私有成员和保护成员，还要指导其方法所期望的参数。对此使用反射API。（了解）

Reflection::*export*(new ReflectionClass('DateTime'));

getConstants()、getFileName()、getFileName();