1、接口的介绍与创建  
           关键字 ：interface  
           接口是抽象类的一种特殊情况，接口为成员属性全部为抽象或常量的特殊抽象类。  
2、接口的引用与规范  
           关键字 ：implements  
接口：一种成员属性全部为抽象的特殊抽象类，在程序中同为规范的作用  
1、类中全部为抽象方法  
2、抽象方法前不用加abstract  
3、接口抽象方法属性为public  
4、成员属性必须为常量  
抽象：  
1、类中至少有一个抽象方法  
2、抽象方法前需加abstract  
  
共同点：  
1、本身都不能被实例化，必须被继承或者引用  
  
2、继承或者引用后，需要把所有抽象方法重载方可使用  
接口有自己的创建关键字：interface 与普通类的创建类似  
interface demo {  
    const  NAME="名字";  
    function Fun1();//对象方法全部为抽象方法，前面无须加adstract  
    function Fun2();  
}  
接口可以使用关键字：implements 多个引用并用逗号  ,  分开.  
  
1、普通类引用接口例子：  
  class mypc implements demo , demo2 , demo3{  
       ……  
    }  
2、抽象类引用接口例子：  
   abstract class mypc implements demo , demo2 , demo3{  
      ……  
    }  
3、继承父类引用接口并存例子：  
  class mypc extends root implements demo , demo2 , demo3{  
       ……  
    }  
4、接口与接口继承：  
   interface demo3 extends demo{  
      ……  
    }