#### 继承

1、重点：**面向对象三大特征：**

* 1. 封装
  2. 继承
  3. 多态

1. 继承：
   1. 作用：减少代码冗余，方便代码修改
   2. 如何实现：
      1. 将所有子类中共同的属性和方法提炼出来，放在一个单独的类中即父类
      2. 子类中只写自己所独有的属性和方法
      3. 然后所有的子类去继承这个父类
   3. 继承关键字是： 子类 extends 父类
   4. 注意：java的继承是 单继承的，即只能继承一个父类。
   5. 子类中如何调用父类的属性和方法
      1. 构造方法：
         1. super(); 调用父类无参构造
         2. super(参数); 调用父类的有参构造
         3. 注意： 子类调用父类构造方法，一定要放在子类构造方法中的第一句。

属性：

super.属性名；

前提： 属性的访问修饰符 不能是private，可以是public protected 默认

方法：

super.方法名();

前提 方法的访问修饰符 不能是private，可以是public protected 默认

* 1. 子类不能继承父类的情况：
     1. 父类中被private修饰的成员
     2. 子类和父类不在同一个包中，父类中被默认访问修饰符修饰的成员
     3. 父类的构造方法
  2. 多重继承关系的初始化顺序：

先初始化父类属性------再初始化父类构造方法---再初始化子类属性--再初始化子类构造方法

1. 方法重写和方法重载
   1. 方法重写特点：
      1. 三同一不同：
         1. 方法的返回值类型相同
         2. 方法名相同
         3. 参数列表相同
         4. 方法的具体实现是不同的
         5. 访问权限不能严于父类
   2. 方法重载特点：
      1. 一同三不同
         1. 方法名必须相同
         2. 参数列表不同(参数个数和参数类型不同)
2. 抽象类
   1. 关键字 abstract

public abstract class 类名{

}

* 1. 特点： 抽象类不能够被实例化，所以我们一般将父类定义为抽象类

1. 抽象方法
   1. 关键字 abstract

public abstract 返回值类型 方法名(){

}

* 1. 特点：
     1. 抽象方法被abstract修饰
     2. 抽象方法是没有方法体的（没有具体的实现）
     3. 抽象方法所在的类必须是抽象类， 抽象类中的方法可以不是抽象方法
     4. 抽象方法必须被子类实现，除非子类也是一个抽象类

#### 多态

1. 多态：

同一种操作，由于条件的不同，产生的结果不一样。

1. 多态的实现有两种方式：
   1. 方法重写：同一个方法在不同类中重写 （三同一不同）
      1. 方法名相同，返回值类型相同，参数列表相同
      2. 方法的具体实现不同
      3. 不能发生在同一个类。
      4. 重写的方法的访问修饰符不能低于原方法
   2. 方法重载：（一同三不同）
      1. 方法名相同
      2. 参数列表不同（参数类型不同，参数个数不同）
      3. 方法重载发生在同一个类中
      4. 方法重载和访问修饰符以及返回值类型没有关系。
2. 实现多态的三要素是
   1. 创建具有继承关系的父类和子类
   2. 子类需要重写父类的方法
   3. 使用父类的引用指向子类对象（由父类去new子类对象） （自动类型转换）
3. 同一个方法实现多态的两种形式
   1. 以父类类型作为参数
   2. 以父类类型作为方法的返回值
4. 由子类得到父类 ---》 自动类型转换 向上转型 父类 a = new 子类();

由父类得到子类 ---》 强制类型转换 向下转型 子类 b = (子类) a；

# 问题

1. 继承的作用是什么，如何实现的，关键字是什么
   1. 作用：减少代码冗余，方便代码修改
   2. 如何实现：
      1. 将所有子类中共同的属性和方法提炼出来，放在一个单独的类中即父类
      2. 子类中只写自己所独有的属性和方法
      3. 然后所有的子类去继承这个父类
   3. 继承关键字是： 子类 extends 父类

2、java继承的特点是什么

单一性和传递性

3、子类不能继承父类的情况：

* 1. 父类中被private修饰的成员
  2. 子类和父类不在同一个包中，父类中被默认访问修饰符修饰的成员
  3. 父类的构造方法

4、方法重写和方法重载的区别是什么

* 1. 方法重写特点：
     1. 三同一不同：
        1. 方法的返回值类型相同
        2. 方法名相同
        3. 参数列表相同
        4. 方法的具体实现是不同的
        5. 访问权限不能严于父类
  2. 方法重载特点：
     1. 一同三不同
        1. 方法名必须相同
        2. 参数列表不同(参数个数和参数类型不同)

5、抽象类的特点是什么

抽象类不能够被实例化，所以我们一般将父类定义为抽象类

1. 抽象方法的特点是什么
   1. 抽象方法被abstract修饰
   2. 抽象方法是没有方法体的（没有具体的实现）
   3. 抽象方法所在的类必须是抽象类， 抽象类中的方法可以不是抽象方法
   4. 抽象方法必须被子类实现，除非子类也是一个抽象类
2. 同一个方法实现多态的两种形式
   1. 以父类类型作为参数
   2. 以父类类型作为方法的返回值