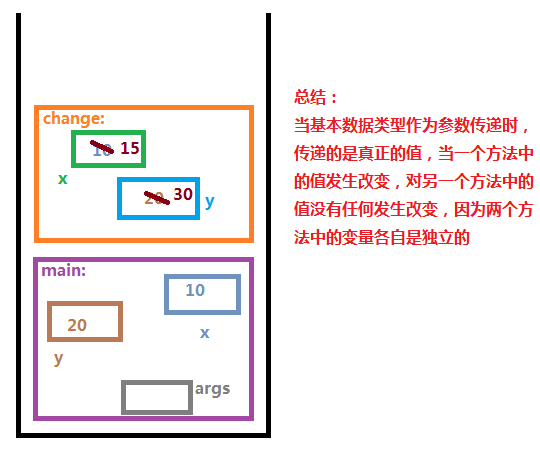
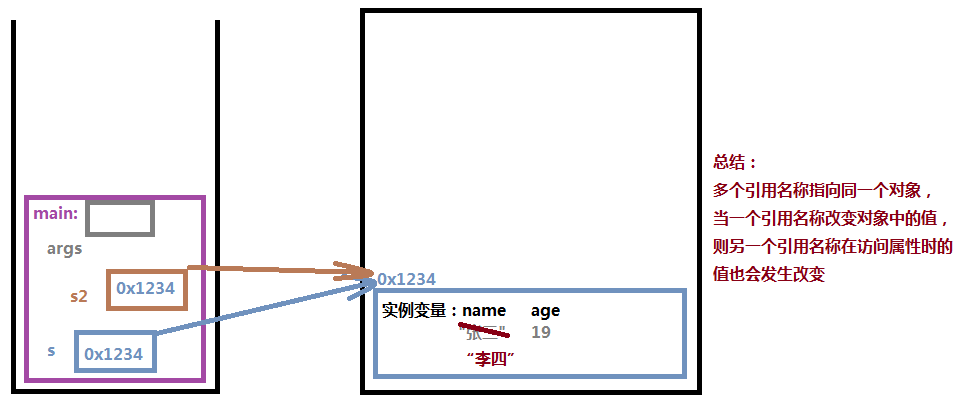
## Day09

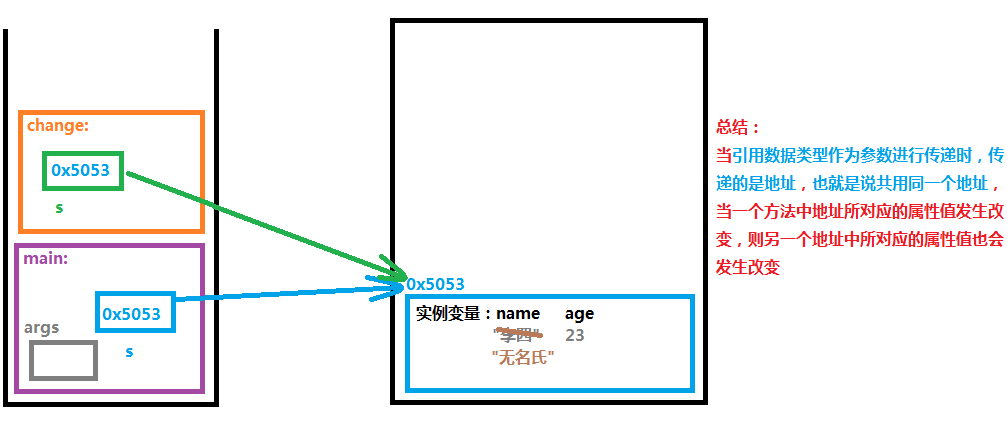
### 9-1 参数传递

* 基本数据类型作为参数传递
  + 传递的是真正的值，在一个方法中改变变量的值，对另一个方法中变量的值没有任何影响，各自变量是独立的



* 引用数据类型作为参数传递
  + 传递的是地址，也就数说多个引用名称共用同一个对象





### 9-2 static关键字

* **static关键字可以修饰属性**
  + 称为静态属性也叫做类的变量
  + 在方法区(共享区)中存储，并且所有对象都可以访问
  + 与类同生死，也就是说类存在时static修饰的属性就存在
  + 可以使用类名.属性名称，也可以使用引用名称.属性名称

注意：当引用名称的值为null时，也可以访问静态属性，不会出现空指针异常

* + 类优先于对象，也就是说类的生命周期比较长
* **static关键字可以修饰方法**
  + 称为静态方法也叫做类的方法
  + 与类同生死，也就是说类存在时static修饰的方法就存在
  + 可以使用类名.方法名称([参数列表]); ，也可以使用引用名称.方法名称([参数列表]);

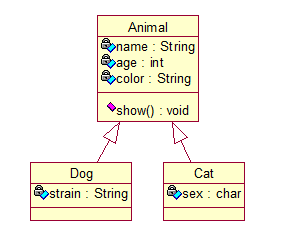
注意：当引用名称的值为null时，也可以访问静态方法，不会出现空指针异常

* 静态方法中只能访问静态属性、静态方法
* 当方法体中只访问静态属性、静态方法时，可以编写为静态方法
* static关键字可以修饰代码块
  + 主要完成对静态属性赋值
  + 静态代码块在类**第一次被载入时**则执行



### 9-3 什么是继承

* 如果多个类中有相同属性和方法，能否单独编写一个类？ ----能
* 解决办法：
  + 将多个类中相同的属性和方法编写在单独一个类中，该类称为父类
  + 在子类继承父类，然后在子类中编写独有的属性和方法



### 9-4 继承好处

* 好处：减少代码的冗余性(重复性)

### 9-5 编写继承代码步骤

第一步：编写父类

[修饰符] class 类名{

//属性和方法

}

第二步：编写子类

[修饰符] class 子类类名 extends 父类类名{

//编写独有属性和方法

}

注意：1.当一个类没有指名继承具体哪个父类时，则默认继承object类

2.Object类是所有类的父类（超类）

class Dog{

}

//等价于

class Dog extends Object{

}

### 9-6 继承中注意事项

* 子类继承父类，但是不能继承父类的构造方法
* 父类也叫做超类(基类)，子类也叫做派生类
* 继承满足的条件：
  + is a 如：猫 是 动物
* 继承具有两大特性：
  + 一个类只能继承一个直接的父类，也就是类是单根性
  + 类具有**传递性**

class A{

//编写2个属性，2个方法

}

class B extends A{

//独有属性1个，方法3个

}

class C extends A{

//独有属性1个

}

class D extends B{

}

### 9-7 重写

* override也叫做覆盖
  + 子类方法名称必须与父类方法名称一致
  + 参数列表一致
  + 返回类型一致或父类方法返回类型的子类类型
  + 修饰符不能缩小范围
* 子类不能重写：
  + 构造方法不能重写
  + 属性不能重写
  + 静态方法不能重写
  + 私有方法不能重写