# 在任意条件查询的情况下简化封装查询对象

## 问题

基本数据模型的增、删、改、查功能中有一项功能要求按照任意条件来查询结果集，以往会将查询条件封装在pojoExample对象中的criteria属性中，如图1所示

图一

假设需要封装的查询条件是根据对象内的属性是否为空作为判断，那么判断语句和封装的次数会随着对象内属性的数量增多而增多，会出现两个问题，

1.手工判断每一个属性容易漏掉其中某些属性的封装

2.冗长的代码

## 解决

所以想解决这个问题，首先分析条件的来源（pojo）和封装到的目的（Criteria）的特点；

1.pojo的特点，它的每一个属性都有对应的get方法，用get方法可以去到属性的值；

2.Criteria的特点是拼接满足条件的方法名称都是类似

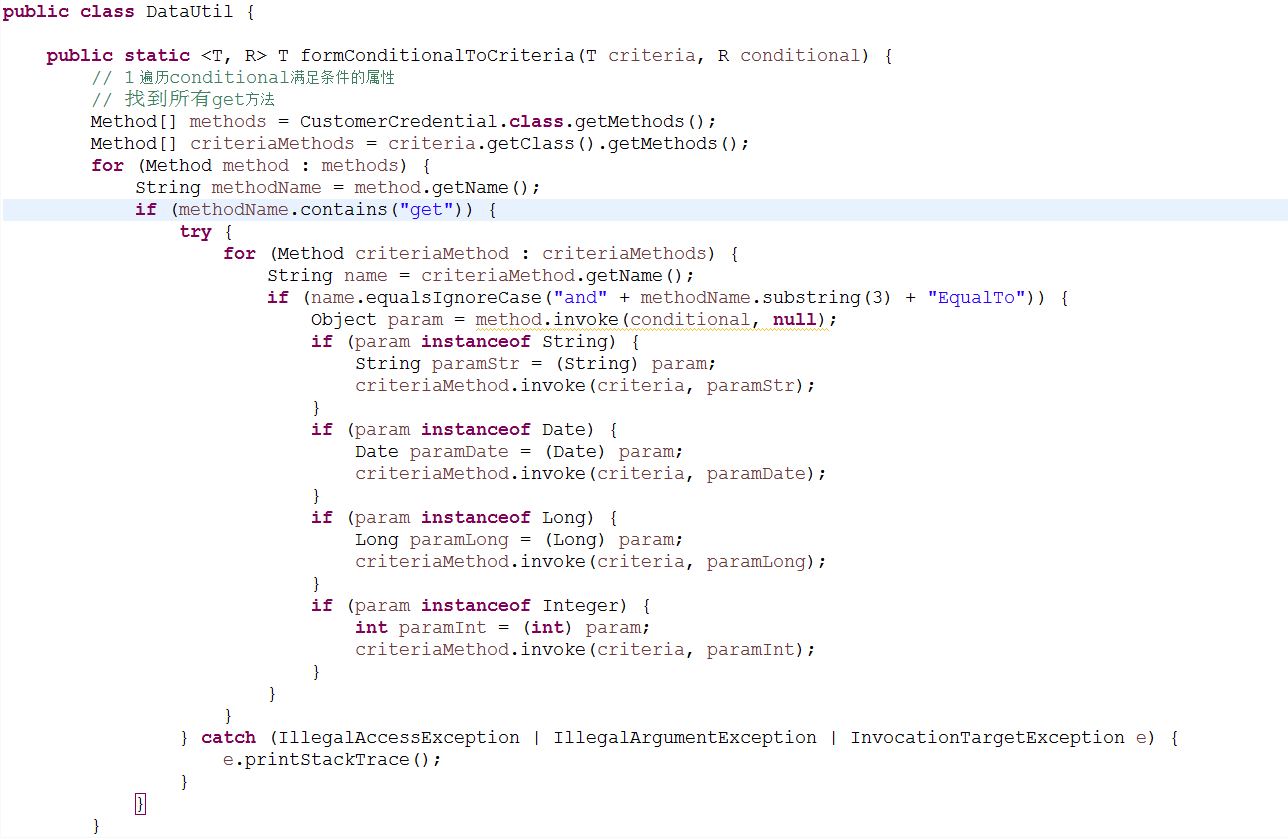
“andXXXEqualTo”，而其中的XXX就是属性的名称（首字母大写）

所以解决的思路是，（如图二所示）

1.遍历出对象所有get方法 （当然getClass()除外），拿到其属性名称（比如 从getId()拿到Id）

2.遍历Criteria所有的形如andXXXEqualTo的方法（XXX就是从第一步中获取的属性名）

3.循环的将get方法获取的属性值，作为Criteria方法的入参进而完成封装



图二

类全路径com.f.a.kobe.util.DataUtil

采用泛型方式满足任意模型的条件查询需求