



链滴

怎么从一个 list 中筛选出另一个 list 中没有的对象

作者: [luomuren](#)

原文链接: <https://ld246.com/article/1661352838482>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

今天突然遇到这样一个需求，有两个用户对象的集合（list1, list2），现在要找出list2中和list1用户不同的元素合并到list1中，以下是实现方法。

```
list1 = [{"userId": null,"userName": "a"},
        {"userId": null,"userName": "b"}]
list2 = [{"userId": null,"userName": "c"},
        {"userId": null,"userName": "d"},
        {"userId": null,"userName": "a"}]
```

1. 双重循环比较

```
List<SysUser> list1_ = new ArrayList<>();
list1_.addAll(list1);
for (SysUser user2 : list2) {
    int i = 0;
    for (; i < list1_.size(); i++) {
        if (user2.getUserName().equals(list1_.get(i).getUserName())) {
            // 如果存在，跳出内层循环
            break;
        }
    }
    if (i >= list1_.size()) {
        list1.add(user2);
    }
}
```

2. 使用Map辅助比较

```
/**
 * @param list1
 * @param list2
 * @return
 */
private static List<SysUser> getDifferent(List<SysUser> list1, List<SysUser> list2) {
    List<SysUser> diff = new ArrayList<>();
    Map<SysUser,Integer> map = new HashMap<>(list2.size());
    // 先将list1中的元素放入map、不同元素的集合
    for (SysUser stu : list1) {
        map.put(stu, 1);
        diff.add(stu);
    }
    for (SysUser stu : list2) {
        // 如果没找到list1中的元素，放入不同元素集合
        if(map.get(stu)==null)
        {
            diff.add(stu);
        }
    }
    return diff;
}
```

3. 使用stream

```
// 从list2中筛选出list1中没有的元素
List<SysUser> diff = list2.stream().filter(
```

```
user2 -> !list1.stream().map(user1 -> user1.getUserName()).collect(Collectors.joining()).contains(user2.getUserName())  
    .collect(Collectors.toList());  
list1.addAll(diff);
```

联系方式

作者: [永夜](#)

邮箱: Evernight@aliyun.com

以上内容有不正确的地方烦请指正! ☺pray☺
ray☺pray