目前的做法是，在mysql建有红包接收列表和红包发送列表，用户每接收一个红包，都会在红包接收列表产生一条数据记录，同理，用户每发送一个红包，也会在红包发送列表产生一条数据记录。将红包记录存储到mysql相应的表中，再通过函数对数据集进行排序。

如果一个用户喜欢发红包，他一天发出了10次红包，每次都分为100个，mysql将为该用户的红包行为产生1010条数据记录。但是涉及到用户金额交易，只有10条数据的重要程度最高，1000个红包的领取数据只作为排序计算使用，对其永久性存储的意义并不大。

如何对红包领取数据进行高效排序计算，又不会对mysql的存储空间造成冗余数据? 我们可以利用redis来存储用户发送和接收红包记录并进行排序。

此方案的成立，基于以下条件：

1. UUID的引入

在mysql的红包接收列表和红包发送列表里，每一条红包记录都由一个id(全局唯一性的索引)来区分，如果用到redis，也需要有类似的全局唯一性值-比如UUID做key。该UUID的结构可以是：user\_id（发起者）+ linux时间戳。UUID是对用户发的红包，打上的标签，user\_id对应发红包的用户在mysql的tb\_user表中的id，Linux时间戳由PHP代码产生，在mysql中也有相应记录，代表了时空的唯一性。

1. 数据的时效性

前面说过，目前是用mysql存储红包记录。

红包操作的逻辑大家都是统一的：用户发起红包群发，提交了金额大小、红包个数的信息(红包个数应该有大小限制)，他的好友看到了红包就去抢，每个人只能领一个，每领一个红包，都会有手气排行榜，红包的个数被领完了，最终会形成红包记录排行。

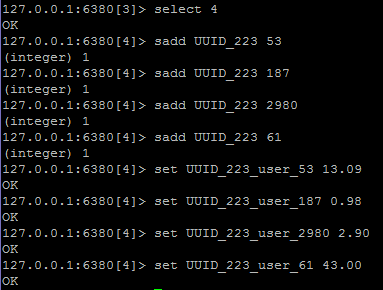
当红包记录最终排行公布时，用户只关心此排行，那么排行信息才是我们应该保存的数据。换句话说，mysql没有必要对用户发送的每一个红包，都做详细记录，也没有必要对用户接收的每一个红包做详细记录。

因此，利用redis的set和list完全可以取代mysql的红包接收列表和红包发送列表，在redis里再将排序结果另存为新的list，当红包领取完毕，记录每一个用户领取该红包的set和list就可以删除掉。

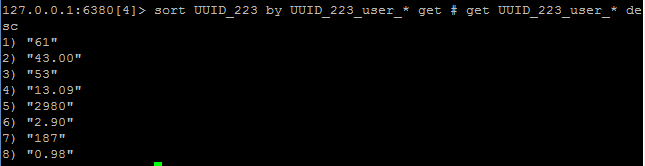
话不多说，请看实战

应用场景：一个用户发出了一定金额、数量是4个的群红包，被user\_id为53、187、2980、61这4个好友领到，他们分别领取到的金额是13.09、0.98、2.90和43.00。

首先我们对这个红包打上一个标签：UUID\_223，然后在redis里创建了名称为UUID\_223的list，对领取该红包的所有好友，做了集合。再针对每一个好友和他领取的红包金额大小，分别创建了一个set。



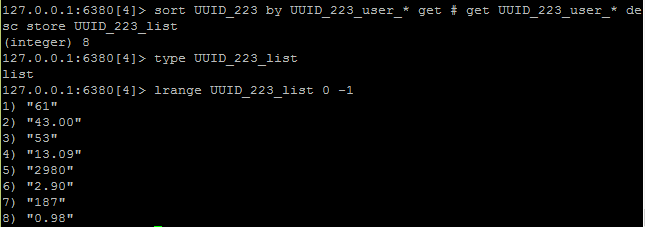
接下来，我们要对每一个好友领取的红包金额，从大到小做排序，并取出对应的用户user\_id



sort UUID\_223 by UUID\_223\_user\_\* get # get UUID\_223\_user\_\* desc此指令的结果颇似hash map的vavlue值:奇数列是领取红包手气最佳的排名，偶数列是领取到红包金额数值大小的排名。

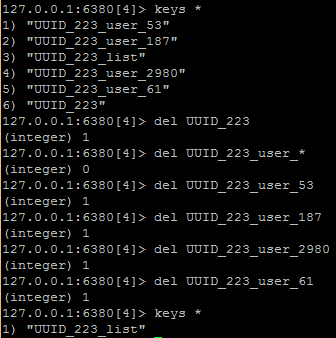
该指令中get # get UUID\_223\_user\_\*是根据sort by的条件（UUID\_223\_user\_\*），筛选出set（UUID\_223\_user\_\*）的value，再匹配list（UUID\_223）的元素。sort by默认以asc(升序)排序，使用desc(降序)将从大到小排序。

当红包被领取后，形成的最终排行如何保存？



sort UUID\_223 by UUID\_223\_user\_\* get # get UUID\_223\_user\_\* desc store UUID\_223\_list该指令使用了store方法，将上面的操作结果保存成新的list，命名为UUID\_223\_list

此时，可以将该红包涉及的其他list和set都删除，只保留最终排行的list



可以为每个set、list设置失效时间，过期自动删除；红包抢完后，手气排行list也可以设置失效时间。

扩展延伸

1. redis会不会遇到瓶颈

对set、list进行sadd、set、del操作，即使是百万条数据同时操作，redis也不会有任何压力。但是如果对大量数据进行sort by，redis的执行效率会降低。但是根据实际的业务场景，利用微信群人数可以对用户发红包的数量加以限制，或者人为限定红包数量的阈值，合理控制redis sort by操作的数据量。

1. mysql如何记录红包操作

mysql的用户列表可以存储该用户发出和接收红包的总和，例如对上面的红包UUID\_223，用户每领取一个红包就将他的接收红包总数加一，用户每发出一个群红包，就将他的发出红包总数加上群发的数量。

1. 红包记录还能再有趣

如果用户对他发的红包被哪些好友收到、他收到了哪些好友发的红包这些信息感兴趣，红包记录可以将这些信息呈现出来。