https://mp.weixin.qq.com/s/6T4RS7rasfmnodE2Devk1A

**Redis存储结构体信息，选hash还是string？**

**1. 存储整个对象，其中JSON序列化过的字符串作为key**

INCR id:users  
SET user:{id} '{"name":"Fred","age":25}'  
SADD users {id}

* 优势：可以认为是“最佳实践”，因为每个对象都是全特性的key，JSON解析特别快，尤其是一次性查询很多个字段的时候
* 劣势：如果只查询一个字段，速度就显得比较慢了

**2. 在hash中存储每个对象的属性**

INCR id:users  
HMSET user:{id} name "Fred" age 25  
SADD users {id}

* 优势：这也可以认为是最佳实践。每个对象都是一个全特性的key。不需要解析JSON字符串
* 劣势：如果要查询对象的全部字段会比较慢。嵌套类型的对象（即对象里面还包着对象）无法轻易存储

**3. 将对象转化为JSON字符串，用hash结构存储**

INCR id:users  
HMSET users {id} '{"name":"Fred","age":25}'

这个方案可以仅用两个key，不需要很多key。但是没法对每个用户对象设置TTL(Time to Live，剩余生存时间)，因为对象仅仅是hash中的一个字段，而不是全特性的key

* 优势：JSON解析很快，尤其是一次查询多个字段时，对主key的命名空间污染更少
* 劣势：如果要存储很多对象，那么内存使用和方案1相当。当只需要查询一个字段时，会比方案2速度慢。答者不认为这是一个“最佳实践”

**4. 存储对象的每个属性作为单独的key**

INCR id:users  
SET user:{id}:name "Fred"  
SET user:{id}:age 25  
SADD users {id}

根据上面的文章，即redis内存优化，这个方案不推荐（除非对象的属性需要专门设置TTL或者别的设置）

* 优势：对象的属性是全特征key，对于应用来说比较好处理
* 劣势：慢，内存消耗更大，不是一个“最佳实践”。对主key的命名空间有很大污染

总的来说，方案4是最不推荐的，方案1和方案2非常相似，也很常见。答者更推荐方案1，因为这个方案允许存储更复杂的对象（也就是说对象可以有很多层嵌套）。方案3通常用在对命名空间比较有要求的场景下，比如说不想要太多key，不关心TTL等参数