

# HashMap

特点:

HashMap是List特点 与 Set特点 结合为一体的一个存储容器

Map的主要特点就是键值对，键是唯一，值可以相同.

键就相当于Set只能存储一个null，值就相当于List可以多个null

底层存储:

数组，哈希表，链表，红黑树

增删查改总体性能比其他集合略高

最坏时间复杂度为 $O(n^2)$ ，平均为 $O(1)$ .

hashMap的扩容与增加是搭配使用的，扩容只会按<扩容

如: 1 2 4 8 16 32 64 128 ..

当增加时会按照 数组容量-1的方式计算hashCode

这样就可以保证计算完毕的hashCode可以保证在数组容量范围.

常用方法:

[V](#) put([K](#) key, [V](#) value)

增加

boolean remove([Object](#) key, [Object](#) value)

删除

boolean replace([K](#) key, [V](#) oldValue, [V](#) newValue)

替换

[Set](#)<[K](#)> keySet()

将该HashMap中所有的key返回

[Collection](#)<[V](#)> values()

将该HashMap中所有的value返回

[Set](#)<[Map.Entry](#)<[K](#), [V](#)>> entrySet()

返回该HashMap的一个Set集合

Set底层就是HashMap 所以拥有key value

迭代方式:

1. 通过keySet()调用迭代器遍历

2. forEach

