

1. 源码比较

```
//两者都实现了Map接口
//HashMap的源码
public class HashMap<K,V>
    extends AbstractMap<K,V>
    implements Map<K,V>, Cloneable, Serializable
    -----
//Hashtable的源码
public class Hashtable<K,V>
    extends Dictionary<K,V>
    implements Map<K,V>, Cloneable, java.io.Serializable
```

```
//Hashtable的put方法有以下语句块，大伙看了都知道
// Make sure the value is not null
if (value == null) {
    throw new NullPointerException();
}

//那么，我们再来看下HashMap的put方法中，有如下语句
//调用某个方法直接把key为null，值为value的键值对插入进去。
if (key == null)
    return putForNullKey(value);
```

```
//以下是Hashtable的方法
public synchronized boolean contains(Object value)
public synchronized boolean containsKey(Object key)
public boolean containsValue(Object value)

//以下是HashMap中的方法，注意，没有contains方法
public boolean containsKey(Object key)
public boolean containsValue(Object value)
```

hashmap	线程不安全	允许有null的键和值	效率高一点、	方法不是Synchronize的要提供外同步	有containsvalue和containsKey方法	HashMap 是Java1.2 引进的Map interface 的一个实现
hashtable	线程安全	不允许有null的键和值	效率稍低、	方法是是Synchronize的	有contains方法	Hashtable 继承于Dictionary 类

