

- [首页](#)
- [技术问答](#)
- [技术教程](#)
- [系统&架构](#)
- [算法&结构](#)
- [编程示例](#)
- [实用工具](#)
- [全部](#)

当前位置: [首页](#) > > [代码示例](#) > > [Java](#) > > 正文

[Java RemovalCause类代码示例](#)

本文整理汇总了Java中

com.google.common.collect.MapMaker.RemovalCause类的典型用法代码示例。如果您正苦于以下问题：Java RemovalCause类的具体用法？Java

RemovalCause怎么用？Java RemovalCause使用的例子？那么恭喜您, 这里精选的类代码示例或许可以为您提供帮助。

RemovalCause类属于com.google.common.collect.MapMaker包，在下文中一共展示了**RemovalCause**类的15个代码示例，这些例子默认根据受欢迎程度排序。您可以为喜欢或者感觉有用的代码点赞，您的评价将有助于我们的系统推荐出更棒的Java代码示例。

示例1: expireEntries

▲ 点赞 3



```
import com.google.common.collect.MapMaker.RemovalCause; //导入依赖的package包/类
@GuardedBy("Segment.this")
void expireEntries() {
    drainRecencyQueue();

    if (expirationQueue.isEmpty()) {
        // There's no point in calling nanoTime() if we have no entries to
        // expire.
        return;
    }
    long now = map.ticker.read();
    ReferenceEntry<K, V> e;
    while ((e = expirationQueue.peek()) != null && map.isExpired(e, now)) {
        if (!removeEntry(e, e.getHash(), RemovalCause.EXPIRED)) {
            throw new AssertionError();
        }
    }
}
```

开发者ID:cplutte, 项目名称:bts, 代码行数:18, 代码来源:[MapMakerInternalMap.java](#)

示例2: removeEntry

▲ 点赞 3



```
import com.google.common.collect.MapMaker RemovalCause; //导入依赖的package包/类
@GuardedBy("Segment.this")
boolean removeEntry(ReferenceEntry<K, V> entry, int hash, RemovalCause cause) {
    int newCount = this.count - 1;
    AtomicReferenceArray<ReferenceEntry<K, V>> table = this.table;
    int index = hash & (table.length() - 1);
    ReferenceEntry<K, V> first = table.get(index);

    for (ReferenceEntry<K, V> e = first; e != null; e = e.getNext()) {
        if (e == entry) {
            ++modCount;
            enqueueNotification(e.getKey(), hash, e.getValueReference().get(), cause);
            ReferenceEntry<K, V> newFirst = removeFromChain(first, e);
            newCount = this.count - 1;
            table.set(index, newFirst);
            this.count = newCount; // write-volatile
            return true;
        }
    }

    return false;
}
```

开发者ID:cplutte, 项目名称:bts, 代码行数:22, 代码来源:[MapMakerInternalMap.java](#)

示例3: expireEntries

▲ 点赞 3



```
import com.google.common.collect.MapMaker RemovalCause; //导入依赖的package包/类
@GuardedBy("this")
void expireEntries() {
    drainRecencyQueue();

    if (expirationQueue.isEmpty()) {
        // There's no point in calling nanoTime() if we have no entries to
        // expire.
        return;
    }
    long now = map.ticker.read();
    ReferenceEntry<K, V> e;
    while ((e = expirationQueue.peek()) != null && map.isExpired(e, now)) {
```

```

        if (!removeEntry(e, e.getHash(), RemovalCause.EXPIRED)) {
            throw new AssertionError();
        }
    }
}

```

开发者ID:sander120786, 项目名称:guava-libraries, 代码行数:18, 代码来源:[MapMakerInternalMap.java](#)

示例4: removeEntry

▲ 点赞 3



```

import com.google.common.collect.MapMaker.RemovalCause; //导入依赖的package包/类
@GuardedBy("this")
boolean removeEntry(ReferenceEntry<K, V> entry, int hash, RemovalCause cause) {
    int newCount = this.count - 1;
    AtomicReferenceArray<ReferenceEntry<K, V>> table = this.table;
    int index = hash & (table.length() - 1);
    ReferenceEntry<K, V> first = table.get(index);

    for (ReferenceEntry<K, V> e = first; e != null; e = e.getNext()) {
        if (e == entry) {
            ++modCount;
            enqueueNotification(e.getKey(), hash, e.getValueReference().get(), cause);
            ReferenceEntry<K, V> newFirst = removeFromChain(first, e);
            newCount = this.count - 1;
            table.set(index, newFirst);
            this.count = newCount; // write-volatile
            return true;
        }
    }

    return false;
}

```

开发者ID:sander120786, 项目名称:guava-libraries, 代码行数:22, 代码来源:[MapMakerInternalMap.java](#)

示例5: testRemovalListener_collected

▲ 点赞 3



```

import com.google.common.collect.MapMaker RemovalCause; //导入依赖的package包/类
public void testRemovalListener_collected() {
    QueuingRemovalListener<Object, Object> listener =
        new QueuingRemovalListener<Object, Object>();
    MapMakerInternalMap<Object, Object> map = makeMap(createMapMaker()
        .concurrencyLevel(1)
        .softValues()
        .removalListener(listener));
    Segment<Object, Object> segment = map.segments[0];
    assertTrue(listener.isEmpty());

    Object one = new Object();
    Object two = new Object();
    Object three = new Object();

    map.put(one, two);
    map.put(two, three);
    assertTrue(listener.isEmpty());

    int hash = map.hash(one);
    ReferenceEntry<Object, Object> entry = segment.getEntry(one, hash);
    map.reclaimValue(entry.getValueReference());
    assertNotified(listener, one, two, RemovalCause.COLLECTED);

    assertTrue(listener.isEmpty());
}

```

开发者ID:sander120786, 项目名称:guava-libraries, 代码行数:26, 代码来源:[MapMakerInternalMapTest.java](#)

示例6: testRemovalListener_size

▲ 点赞 3



```

import com.google.common.collect.MapMaker RemovalCause; //导入依赖的package包/类
public void testRemovalListener_size() {
    QueuingRemovalListener<Object, Object> listener =
        new QueuingRemovalListener<Object, Object>();
    MapMakerInternalMap<Object, Object> map = makeMap(createMapMaker()
        .concurrencyLevel(1)
        .maximumSize(2)
        .removalListener(listener));
    assertTrue(listener.isEmpty());

    Object one = new Object();
    Object two = new Object();

```

```

Object three = new Object();
Object four = new Object();

map.put(one, two);
map.put(two, three);
assertTrue(listener.isEmpty());
map.put(three, four);
assertNotified(listener, one, two, RemovalCause.SIZE);

assertTrue(listener.isEmpty());
}

```

开发者ID:sander120786, 项目名称:guava-libraries, 代码行数:23, 代码来源:[MapMakerInternalMapTest.java](#)

示例7: evictEntries

▲ 点赞 2 ▼

```

import com.google.common.collect.MapMaker.RemovalCause; //导入依赖的package包/类
/**
 * Performs eviction if the segment is full. This should only be called prior to adding a new
 * entry and increasing {@code count}.
 *
 * @return {@code true} if eviction occurred
 */
@GuardedBy("Segment.this")
boolean evictEntries() {
    if (map.evictsBySize() && count >= maxSegmentSize) {
        drainRecencyQueue();

        ReferenceEntry<K, V> e = evictionQueue.remove();
        if (!removeEntry(e, e.getHash(), RemovalCause.SIZE)) {
            throw new AssertionError();
        }
        return true;
    }
    return false;
}

```

开发者ID:cplutte, 项目名称:bts, 代码行数:20, 代码来源:[MapMakerInternalMap.java](#)

示例8: clear

▲ 点赞 2



```
import com.google.common.collect.MapMaker.RemovalCause; //导入依赖的package包/类
void clear() {
    if (count != 0) {
        lock();
        try {
            AtomicReferenceArray<ReferenceEntry<K, V>> table = this.table;
            if (map.removalNotificationQueue != DISCARDING_QUEUE) {
                for (int i = 0; i < table.length(); ++i) {
                    for (ReferenceEntry<K, V> e = table.get(i); e != null; e = e.getNext()) {
                        // Computing references aren't actually in the map yet.
                        if (!e.getValueReference().isComputingReference()) {
                            enqueueNotification(e, RemovalCause.EXPLICIT);
                        }
                    }
                }
            }
            for (int i = 0; i < table.length(); ++i) {
                table.set(i, null);
            }
            clearReferenceQueues();
            evictionQueue.clear();
            expirationQueue.clear();
            readCount.set(0);

            ++modCount;
            count = 0; // write-volatile
        } finally {
            unlock();
            postWriteCleanup();
        }
    }
}
```

开发者ID:cplutte, 项目名称:bts, 代码行数:32, 代码来源:[MapMakerInternalMap.java](#)

示例9: reclaimKey

▲ 点赞 2



```
import com.google.common.collect.MapMaker.RemovalCause; //导入依赖的package包/类
/**
 * Removes an entry whose key has been garbage collected.
 */
boolean reclaimKey(ReferenceEntry<K, V> entry, int hash) {
```

```

lock();
try {
    int newCount = count - 1;
    AtomicReferenceArray<ReferenceEntry<K, V>> table = this.table;
    int index = hash & (table.length() - 1);
    ReferenceEntry<K, V> first = table.get(index);

    for (ReferenceEntry<K, V> e = first; e != null; e = e.getNext()) {
        if (e == entry) {
            ++modCount;
            enqueueNotification(
                e.getKey(), hash, e.getValueReference().get(), RemovalCause.COLLECTED);
            ReferenceEntry<K, V> newFirst = removeFromChain(first, e);
            newCount = this.count - 1;
            table.set(index, newFirst);
            this.count = newCount; // write-volatile
            return true;
        }
    }

    return false;
} finally {
    unlock();
    postWriteCleanup();
}
}

```

开发者ID:cplutte, 项目名称:bts, 代码行数:31, 代码来源:[MapMakerInternalMap.java](#)

示例10: reclaimValue

▲ 点赞 2 ▼

```

import com.google.common.collect.MapMaker.RemovalCause; //导入依赖的package包/类
/**
 * Removes an entry whose value has been garbage collected.
 */
boolean reclaimValue(K key, int hash, ValueReference<K, V> valueReference) {
    lock();
    try {
        int newCount = this.count - 1;
        AtomicReferenceArray<ReferenceEntry<K, V>> table = this.table;
        int index = hash & (table.length() - 1);
        ReferenceEntry<K, V> first = table.get(index);

        for (ReferenceEntry<K, V> e = first; e != null; e = e.getNext()) {

```

```

K entryKey = e.getKey();
if (e.getHash() == hash && entryKey != null
    && map.keyEquivalence.equivalent(key, entryKey)) {
    ValueReference<K, V> v = e.getValueReference();
    if (v == valueReference) {
        ++modCount;
        enqueueNotification(key, hash, valueReference.get(), RemovalCause.COLLECTED);
        ReferenceEntry<K, V> newFirst = removeFromChain(first, e);
        newCount = this.count - 1;
        table.set(index, newFirst);
        this.count = newCount; // write-volatile
        return true;
    }
    return false;
}
}

return false;
} finally {
    unlock();
    if (!isHeldByCurrentThread()) { // don't cleanup inside of put
        postWriteCleanup();
    }
}
}
}

```

开发者ID:cplutte, 项目名称:bts, 代码行数:38, 代码来源:[MapMakerInternalMap.java](#)

示例11: compute

▲ 点赞 2



```

import com.google.common.collect.MapMaker.RemovalCause; //导入依赖的package包/类
V compute(K key, int hash, ReferenceEntry<K, V> e,
    ComputingValueReference<K, V> computingValueReference)
    throws ExecutionException {
    V value = null;
    long start = System.nanoTime();
    long end = 0;
    try {
        // Synchronizes on the entry to allow failing fast when a recursive computation is
        // detected. This is not fool-proof since the entry may be copied when the segment
        // is written to.
        synchronized (e) {
            value = computingValueReference.compute(key, hash);
            end = System.nanoTime();
        }
    }
}

```



```

    }
    if (value != null) {
        // putIfAbsent
        V oldValue = put(key, hash, value, true);
        if (oldValue != null) {
            // the computed value was already clobbered
            enqueueNotification(key, hash, value, RemovalCause.REPLACED);
        }
    }
    return value;
} finally {
    if (end == 0) {
        end = System.nanoTime();
    }
    if (value == null) {
        clearValue(key, hash, computingValueReference);
    }
}
}
}

```

开发者ID:cplutte, 项目名称:bts, 代码行数:33, 代码来源:[ComputingConcurrentHashMap.java](#)

示例12: evictEntries

▲ 点赞 2



```

import com.google.common.collect.MapMaker.RemovalCause; //导入依赖的package包/类
/**
 * Performs eviction if the segment is full. This should only be called prior to adding a new
 * entry and increasing {@code count}.
 *
 * @return {@code true} if eviction occurred
 */
@GuardedBy("this")
boolean evictEntries() {
    if (map.evictsBySize() && count >= maxSegmentSize) {
        drainRecencyQueue();

        ReferenceEntry<K, V> e = evictionQueue.remove();
        if (!removeEntry(e, e.getHash(), RemovalCause.SIZE)) {
            throw new AssertionError();
        }
        return true;
    }
    return false;
}

```

```
}
```

开发者ID:sander120786, 项目名称:guava-libraries, 代码行数:20, 代码来源:[MapMakerInternalMap.java](#)

示例13: testRemovalListener_replaced

▲ 点赞 2 ▼

```
import com.google.common.collect.MapMaker.RemovalCause; //导入依赖的package包/类
public void testRemovalListener_replaced() {
    QueuingRemovalListener<Object, Object> listener =
        new QueuingRemovalListener<Object, Object>();
    MapMakerInternalMap<Object, Object> map = makeMap(createMapMaker()
        .removalListener(listener));
    assertTrue(listener.isEmpty());

    Object one = new Object();
    Object two = new Object();
    Object three = new Object();
    Object four = new Object();
    Object five = new Object();
    Object six = new Object();

    map.put(one, two);
    map.put(one, three);
    assertNotified(listener, one, two, RemovalCause.REPLACED);

    Map<Object, Object> newMap = ImmutableMap.of(one, four);
    map.putAll(newMap);
    assertNotified(listener, one, three, RemovalCause.REPLACED);

    map.replace(one, five);
    assertNotified(listener, one, four, RemovalCause.REPLACED);

    map.replace(one, five, six);
    assertNotified(listener, one, five, RemovalCause.REPLACED);
}
```

开发者ID:sander120786, 项目名称:guava-libraries, 代码行数:29, 代码来源:[MapMakerInternalMapTest.java](#)

示例14: assertNotified

▲ 点赞 2



```
import com.google.common.collect.MapMaker RemovalCause; //导入依赖的package包/类
static <K, V> void assertNotified(
    QueuingRemovalListener<K, V> listener, K key, V value, RemovalCause cause) {
    RemovalNotification<K, V> notification = listener.remove();
    assertSame(key, notification.getKey());
    assertSame(value, notification.getValue());
    assertSame(cause, notification.getCause());
}
```

开发者ID:sander120786, 项目名称:guava-libraries, 代码行数:8, 代码来源:[MapMakerInternalMapTest.java](#)

示例15: testRemoveEntry

▲ 点赞 2



```
import com.google.common.collect.MapMaker RemovalCause; //导入依赖的package包/类
public void testRemoveEntry() {
    MapMakerInternalMap<Object, Object> map = makeMap(createMapMaker()
        .concurrencyLevel(1)
        .initialCapacity(1)
        .maximumSize(SMALL_MAX_SIZE)
        .expireAfterWrite(99999, SECONDS)
        .removalListener(new CountingRemovalListener<Object, Object>()));
    Segment<Object, Object> segment = map.segments[0];
    AtomicReferenceArray<ReferenceEntry<Object, Object>> table = segment.table;
    assertEquals(1, table.length());

    Object key = new Object();
    Object value = new Object();
    int hash = map.hash(key);
    DummyEntry<Object, Object> entry = createDummyEntry(key, hash, value, null);

    // remove absent
    assertFalse(segment.removeEntry(entry, hash, RemovalCause.COLLECTED));

    // remove live
    segment.recordWrite(entry);
    table.set(0, entry);
    segment.count = 1;
    assertTrue(segment.removeEntry(entry, hash, RemovalCause.COLLECTED));
    assertNotificationEnqueued(map, key, value);
    assertTrue(map.removalNotificationQueue.isEmpty());
    assertFalse(segment.evictionQueue.contains(entry));
}
```

```
assertFalse(segment.expirationQueue.contains(entry));
assertEquals(0, segment.count);
assertNull(table.get(0));
}
```

开发者ID:sander120786, 项目名称:guava-libraries, 代码行数:32, 代码来源:[MapMakerInternalMapTest.java](#)

注: [本文](#)中的**com.google.common.collect.MapMaker.RemoveCause**类示例整理自Github/MSDocs等源码及文档管理平台, 相关代码片段筛选自各路编程大神贡献的开源项目, 源码版权归原作者所有, 传播和使用请参考对应项目的**License**; 未经允许, 请勿转载。

相关方法

- [RemoveCause.COLLECTED](#)
- [RemoveCause.EXPLICIT](#)