

Java 集合

相关阅读：

- [Java 集合常见面试题总结\(上\) - JavaGuide -2022 <https://javaguide.cn/java/collection/java-collection-questions-01.html>](https://javaguide.cn/java/collection/java-collection-questions-01.html)
- [Java 集合常见面试题总结\(下\) - JavaGuide -2022 <https://javaguide.cn/java/collection/java-collection-questions-02.html>](https://javaguide.cn/java/collection/java-collection-questions-02.html)

基础

说说 `List` , `Set` , `Map` 三者的区别？

💡 提示：可以从这些数据结构中的元素是否有序、是否可以重复、存储的元素类型（比如 `Map` 存储的就是键值对）等方面来回答。

`List` , `Set` , `Map` 在 Java 中分别由哪些对应的实现类？底层的数据结构？

💡 提示：拿 `List` 来举例，`List` 的常见实现类以及它们的数据结构：

- `ArrayList` : `Object[]` 数组
- `Vector` : `Object[]` 数组
- `LinkedList` : 双向链表(JDK1.6 之前为循环链表, JDK1.7 取消了循环)

有哪些集合是线程不安全的？怎么解决呢？

💡 提示：这块比较常问的是 `ArrayList` 和 `Vector` 、 `HashMap` 和 `ConcurrentHashMap` （高频问题，重要）。被问到 `Vector` 的时候，你紧接着可能会被问到 `ArrayList` 和 `Vector` 的区别。被问到 `ConcurrentHashMap` 的时候，你紧接着就可能会被问到 `ConcurrentHashMap` 相关的问题比如 `ConcurrentHashMap` 是如何保证线程安全的。

List

`ArrayList` 和 `Vector` 的区别? ★★☆☆

`ArrayList` 与 `LinkedList` 区别? ★★☆☆☆

💡 提示：底层数据结构、插入和删除是否受元素位置的影响、是否支持快速随机访问、内存空间占用。

`ArrayList` 扩容机制 ★★☆☆

相关阅读：[ArrayList 源码&扩容机制分析 <https://javaguide.cn/java/collection/arraylist-source-code.html>](https://javaguide.cn/java/collection/arraylist-source-code.html) 。

Queue

`Queue` 与 `Deque` 的区别 ★★☆☆

💡 提示：`Queue` 是单端队列，只能从一端插入元素，另一端删除元素，实现上一般遵循先进先出（FIFO）规则。`Deque` 是双端队列，在队列的两端均可以插入或删除元素。

`ArrayDeque` 与 `LinkedList` 的区别 ★★☆☆

`PriorityQueue` 有什么特点 ★★☆☆

💡 提示：元素出队顺序是与优先级相关的，即总是优先级最高的元素先出队。

HashMap

`HashMap` 查询，删除的时间复杂度 ★★☆☆☆

💡 提示：

- 没有哈希冲突的情况
- 转链表的情况
- 链表转红黑树的情况

`HashMap` 的底层实现 ★★☆☆☆☆

💡 提示：

- JDK1.8 之前：数组和链表
- JDK1.8 之后：多了红黑树

`HashMap` 的长度为什么是 2 的幂次方 ★★☆☆

💡 提示：提高运算效率。

比较 `HashSet` 、 `LinkedHashSet` 和 `TreeSet` 三者的异同 ★★☆☆



提示:

- `HashSet`、`LinkedHashSet` 和 `TreeSet` 都是 `Set` 接口的实现类，都能保证元素唯一，并且都不是线程安全的。
- `HashSet`、`LinkedHashSet` 和 `TreeSet` 的主要区别在于底层数据结构不同（底层数据结构不同又导致这三者的应用场景不同）。

`HashMap` 和 `Hashtable` 的区别？`HashMap` 和 `HashSet` 区别？`HashMap` 和 `TreeMap` 区别？☆☆☆☆

ConcurrentHashMap

`ConcurrentHashMap` 和 `Hashtable` 的区别？☆☆☆☆



提示:

- 底层数据结构
- 实现线程安全的方式的区别

`ConcurrentHashMap` 线程安全的具体实现方式/底层具体实现 ☆☆☆☆☆



提示:

- JDK 1.7 : `Segment` 分段锁
- JDK 1.8 : `CAS` 和 `synchronized`