Java集合框架Set接口

Set接口概念

Java集合框架中的Set接口是一种不允许包含重复元素的集合。Set接口继承自Collection接口，因此它具有Collection接口定义的所有方法。同时，Set接口还具有自己的特殊方法，例如：添加元素、删除元素、判断元素是否存在等。

Set接口的实现类包括HashSet、LinkedHashSet和TreeSet。

HashSet是基于哈希表实现的Set集合，它不保证集合中元素的顺序。由于哈希表的实现机制，HashSet的添加、删除和查找操作都具有很好的性能，时间复杂度为O(1)。

LinkedHashSet是基于链表和哈希表实现的Set集合，它保证集合中元素的插入顺序。在插入元素时，它既会将元素插入到链表的末尾，又会将元素的哈希值和引用存储到哈希表中。因此，LinkedHashSet在性能上稍逊于HashSet，但在维护元素插入顺序方面具有优势。

TreeSet是基于红黑树实现的Set集合，它可以保证集合中元素的有序性。红黑树的实现机制使得TreeSet的插入、删除和查找操作的时间复杂度为O(logN)，因此在处理大量数据时，TreeSet具有更好的性能。

Set接口常用的方法

add(E e)：将指定元素添加到集合中，如果元素已经存在则不会添加。

remove(Object o)：从集合中移除指定元素，如果集合中不包含该元素，则返回false。

contains(Object o)：判断集合中是否包含指定元素，如果包含则返回true，否则返回false。

size()：返回集合中元素的个数。

isEmpty()：判断集合是否为空，如果集合中没有元素则返回true，否则返回false。

clear()：清空集合中的所有元素。

iterator()：返回一个迭代器，可以用于遍历集合中的元素。

toArray()：返回一个包含集合中所有元素的数组。

addAll(Collection<? extends E> c)：将指定集合中的所有元素添加到当前集合中。

retainAll(Collection<?> c)：保留当前集合和指定集合中的公共元素，移除其他元素。

removeAll(Collection<?> c)：移除当前集合中和指定集合中的公共元素。

示例

Java | 复制代码

```java
import java.util.HashSet;
import java.util.Set;

public class SetExample {
    public static void main(String[] args) {
        // 创建一个HashSet集合
        Set<String> set = new HashSet<>();

        // 向集合中添加元素
        set.add("apple");
        set.add("banana");
        set.add("orange");
        set.add("pear");

        // 输出集合大小
        System.out.println("集合大小: " + set.size());

        // 遍历集合
        for (String fruit : set) {
            System.out.println(fruit);
        }

        // 删除一个元素
        set.remove("banana");

        // 判断集合中是否包含指定元素
        System.out.println("集合中是否包含pear: " + set.contains("pear"));

        // 清空集合
        set.clear();

        // 判断集合是否为空
        System.out.println("集合是否为空: " + set.isEmpty());
    }
}
```