|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 增 | 删 | 改 | 查 | E | 是否线程安全 | 数据结构 |
| Collection | add(E):boolean | remove(Object):boolean |  | contains(Object):boolean |  |  |  |
| HashSet | add(E):boolean | remove(Object):boolean |  | contains(Object):boolean | 可以为null | 不安全 | 同HashMap |
| TreeSet | add(E):boolean | remove(Object):boolean  pollFirst():E  pollLast():E |  | contains(Object):boolean  first():E  last():E | 不可以为null | 不安全 | 同TreeMap |

Set：

List：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 增 | 删 | 改 | 查 | E | 是否线程安全 | 数据结构 |
| Collection | add(E):boolean | remove(Object):boolean |  | contains(Object):boolean |  |  |  |
| ArrayList | add(E):boolean  add(int,E):void | remove(Object):boolean  remove(int):E | set(int,E):E | contains(Object):boolean  get(int):E | 可以为null | 不安全 | Object数组 |
| LinkedList | add(E):boolean   |  | | --- | | add(int,E):void | | addFirst(E):void | | addLast(E):void |  |  | | --- | | offer(E):boolean | | offerFirst(E):boolean | | offerLast(E):boolean |   push(E):void | remove(Object):boolean  remove():E   |  | | --- | | remove(int):E | | removeFirst():E | | removeLast():E |  |  | | --- | | poll():E | | pollFirst():E | | pollLast():E |   pop():E | set(int,E):E | contains(Object):boolean   |  | | --- | | get(int):E | | getFirst():E | | getLast():E |  |  | | --- | | peek():E | | peekFirst():E | | peekLast():E |   element():E | 可以为null | 不安全 | 双向链表 |
| Vector | add(E):boolean  add(int,E):void  addElement(E):void  insertElementAt(E,int):void | remove(Object):boolean  remove(int):E  removeElement(Object):boolean  removeElementAt(int):void | set(int,E):E  setElementAt(E,int):void | contains(Object):boolean  get(int):E  elementAt(int):E  firstElement():E  lastElement():E | 可以为null | 安全 | Object数组 |
| Stack | add(E):boolean  push(E):void | remove(Object):boolean  pop():E |  | contains(Object):boolean  peek():E | 可以为null | 安全 | Object数组 |

Queue:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 增 | 删 | 改 | 查 | E | 是否线程安全 | 数据结构 |
| Collection | add(E):boolean | remove(Object):boolean |  | contains(Object):boolean |  |  |  |
| PriorityQueue | add(E):boolean  offer(E):boolean | remove(Object):boolean  poll():E |  | contains(Object):boolean  peek():E | 不可以为null | 不安全 | 堆 |
| ArrayDeque | add(E):boolean   |  | | --- | | addFirst(E):void | | addLast(E):void |  |  | | --- | | offer(E):boolean | | offerFirst(E):boolean | | offerLast(E):boolean |   push(E):void | remove(Object):boolean  remove():E   |  | | --- | | removeFirst():E | | removeLast():E |  |  | | --- | | poll():E | | pollFirst():E | | pollLast():E |   pop():E | set(int,E):E | contains(Object):boolean   |  | | --- | | getFirst():E | | getLast():E |  |  | | --- | | peek():E | | peekFirst():E | | peekLast():E |   element():E | 不可以为null | 不安全 | 循环数组 |

BlockingQueue:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 增 | 删 | 改 | 查 | E | 是否线程安全 | 数据结构 |
| Collection | add(E):boolean | remove(Object):boolean |  | contains(Object):boolean |  |  |  |
| ArrayBlockingQueue | add(E):boolean   |  | | --- | | offer(E):boolean | | offer(E,long,TimeUnit):boolean |   put(E):void | |  | | --- | | remove(Object):boolean | | removeAt(int):void |  |  | | --- | | poll():E | | poll(long,TimeUnit):E |   take():E |  | contains(Object):boolean  peek():E | 不可以为null | 安全 | Object数组 |
| LinkedBlockingQueue | add(E):boolean   |  | | --- | | offer(E):boolean | | offer(E,long,TimeUnit):boolean |   put(E):void | remove(Object):boolean   |  | | --- | | poll():E | | poll(long,TimeUnit):E |   take():E |  | contains(Object):boolean  peek():E | 不可以为null | 安全 | 单链表 |
| SynchronousQueue | add(E):boolean   |  | | --- | | offer(E):boolean | | offer(E,long,TimeUnit):boolean |   put(E):void | remove(Object):boolean   |  | | --- | | poll():E | | poll(long,TimeUnit):E |   take():E |  | contains(Object):boolean  peek():E | 不可以为null | 安全 |  |

Map:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 增 | 删 | 改 | 查 | K,V | 是否线程安全 | 数据结构 |
| Map | put(K,V):V  putIfAbsent(K,V):V | remove(Object):V  remove(Object,Object):boolean | replace(K,V):V  replace(K,V,V):boolean | containsKey(Object):boolean  containsValue(Object):boolean  get(Object):V  getOrDefault(Object,V):V |  |  |  |
| HashMap | put(K,V):V  putIfAbsent(K,V):V | remove(Object):V  remove(Object,Object):boolean | replace(K,V):V  replace(K,V,V):boolean | containsKey(Object):boolean  containsValue(Object):boolean  get(Object):V  getOrDefault(Object,V):V | key可以为null  value可以为null | 不安全 | 数组+链表/红黑树 |
| Hashtable | put(K,V):V  putIfAbsent(K,V):V | remove(Object):V  remove(Object,Object):boolean | replace(K,V):V  replace(K,V,V):boolean | containsKey(Object):boolean  containsValue(Object):boolean  get(Object):V  getOrDefault(Object,V):V  contains(Object):boolean | key不可以为null  value不可以为null | 安全 | 数组+链表 |
| TreeMap | put(K,V):V  putIfAbsent(K,V):V | remove(Object):V  remove(Object,Object):boolean | replace(K,V):V  replace(K,V,V):boolean | containsKey(Object):boolean  containsValue(Object):boolean  get(Object):V  getOrDefault(Object,V):V  firstKey():K  lastKey():K | key不可以为null  value可以为null | 不安全 | 红黑树 |
| WeakHashMap | put(K,V):V  putIfAbsent(K,V):V | remove(Object):V  remove(Object,Object):boolean | replace(K,V):V  replace(K,V,V):boolean | containsKey(Object):boolean  containsValue(Object):boolean  get(Object):V  getOrDefault(Object,V):V | key可以为null  value可以为null | 不安全 | 数组+链表 |
| ConcurrentHashMap | put(K,V):V  putIfAbsent(K,V):V | remove(Object):V  remove(Object,Object):boolean | replace(K,V):V  replace(K,V,V):boolean | containsKey(Object):boolean  containsValue(Object):boolean  get(Object):V  getOrDefault(Object,V):V  contains(Object):boolean | key不可以为null  value不可以为null | 安全 | 数组+链表/红黑树 |

一般而言，需要排序的和线程安全的类Key不可以为null

但是也有特例：ArrayDeque，Vector，Stack

key和value均不可以为null：Hashtable，ConcurrentHashMap