

# El paquete java.util (sin colecciones)

## CLASES

- o java.lang.[Object](#)
  - o java.util.[Arrays](#)
  - o java.util.[Objects](#)
  - o java.util.[Collections](#) (dentro de las colecciones - siguiente página)

Clases maleta

Clases variadas

- o java.lang.[Object](#)
  - o java.util.[BitSet](#) (implements java.lang.[Cloneable](#), java.io.[Serializable](#))
  - o java.util.[Calendar](#) (implements java.lang.[Cloneable](#), java.lang.[Comparable](#)<T>, java.io.[Serializable](#))
    - o java.util.[GregorianCalendar](#)
  - o java.util.[Currency](#) (implements java.io.[Serializable](#))
  - o java.util.[Date](#) (implements java.lang.[Cloneable](#), java.lang.[Comparable](#)<T>, java.io.[Serializable](#))
  - o java.util.[EventListenerProxy](#)<T> (implements java.util.[EventListener](#))
  - o java.util.[EventObject](#) (implements java.io.[Serializable](#))
  - o java.util.[FormattableFlags](#)
  - o java.util.[Formatter](#) (implements java.io.[Closeable](#), java.io.[Flushable](#))
  - o java.util.[Locale](#) (implements java.lang.[Cloneable](#), java.io.[Serializable](#))
  - o java.util.[Locale.Builder](#)
  - o java.util.[Observable](#)
  - o java.security.[Permission](#) (implements java.security.[Guard](#), java.io.[Serializable](#))
    - o java.security.[BasicPermission](#) (implements java.io.[Serializable](#))
      - o java.util.[PropertyPermission](#)
  - o java.util.[Random](#) (implements java.io.[Serializable](#))
  - o java.util.[ResourceBundle](#)
    - o java.util.[ListResourceBundle](#)
    - o java.util.[PropertyResourceBundle](#)
  - o java.util.[ResourceBundle.Control](#)
  - o java.util.[Scanner](#) (implements java.io.[Closeable](#), java.util.[Iterator](#)<E>)
  - o java.util.[ServiceLoader](#)<S> (implements java.lang.[Iterable](#)<T>)
  - o java.util.[StringTokenizer](#) (implements java.util.[Enumeration](#)<E>)
  - o java.util.[Timer](#)
  - o java.util.[TimerTask](#) (implements java.lang.[Runnable](#))
  - o java.util.[TimeZone](#) (implements java.lang.[Cloneable](#), java.io.[Serializable](#))
    - o java.util.[SimpleTimeZone](#)
  - o java.util.[UUID](#) (implements java.lang.[Comparable](#)<T>, java.io.[Serializable](#))

## INTERFACES

- o java.util.[Enumeration](#)<E>
- o java.util.[EventListener](#)
- o java.util.[Formattable](#)
- o java.util.[Observer](#)

## (Clases) ARROJABLES

- o java.lang.[Object](#)
  - o java.lang.[Throwable](#) (implements java.io.[Serializable](#))
    - o java.lang.[Error](#)
      - o java.util.[ServiceConfigurationError](#)
    - o java.lang.[Exception](#)
      - o java.io.[IOException](#)
        - o java.util.[InvalidPropertiesFormatException](#)
      - o java.lang.[RuntimeException](#)
        - o java.util.[ConcurrentModificationException](#)
        - o java.util.[EmptyStackException](#)
        - o java.lang.[IllegalArgumentException](#)
          - o java.util.[IllegalFormatException](#)
            - o java.util.[DuplicateFormatFlagsException](#)
            - o java.util.[FormatFlagsConversionMismatchException](#)
            - o java.util.[IllegalFormatCodePointException](#)
            - o java.util.[IllegalFormatConversionException](#)
            - o java.util.[IllegalFormatFlagsException](#)
            - o java.util.[IllegalFormatPrecisionException](#)
            - o java.util.[IllegalFormatWidthException](#)
            - o java.util.[MissingFormatArgumentException](#)
            - o java.util.[MissingFormatWidthException](#)
            - o java.util.[UnknownFormatConversionException](#)
            - o java.util.[UnknownFormatFlagsException](#)
      - o java.lang.[IllegalStateException](#)
        - o java.util.[FormatterClosedException](#)
      - o java.util.[IllformedLocaleException](#)
      - o java.util.[MissingResourceException](#)
      - o java.util.[NoSuchElementException](#)
        - o java.util.[InputMismatchException](#)
    - o java.util.[TooManyListenersException](#)

## ENUMERACIONES

- o java.lang.[Object](#)
    - o java.lang.[Enum](#)<E> (implements java.lang.[Comparable](#)<T>, java.io.[Serializable](#))
      - o java.util.[Locale.Category](#)
      - o java.util.[Formatter.BigDecimalLayoutForm](#)
- Es una clase  
Es raíz de enumeraciones

# El paquete java.util

## --- Las colecciones ---

### Los conceptos

(interfaces)

- o java.util.[Comparator](#)<T>
- o java.lang.[Iterable](#)<T>
  - o java.util.[Collection](#)<E>
    - o java.util.[List](#)<E>
    - o java.util.[Queue](#)<E>
      - o java.util.[Deque](#)<E>
    - o java.util.[Set](#)<E>
      - o java.util.[SortedSet](#)<E>
      - o java.util.[NavigableSet](#)<E>
- o java.util.[Iterator](#)<E>
  - o java.util.[ListIterator](#)<E>
- o java.util.[Map](#)<K,V>
  - o java.util.[SortedMap](#)<K,V>
  - o java.util.[NavigableMap](#)<K,V>
- o java.util.[Map.Entry](#)<K,V>
- o java.util.[RandomAccess](#)

### Las implementaciones

- o java.lang.[Object](#)
  - o java.util.[AbstractCollection](#)<E> (implements java.util.[Collection](#)<E>)
  - o java.util.[AbstractList](#)<E> (implements java.util.[List](#)<E>)
    - o java.util.[AbstractSequentialList](#)<E>
      - o java.util.[LinkedList](#)<E> (implements java.lang.[Cloneable](#), java.util.[Deque](#)<E>, java.util.[List](#)<E>, java.io.[Serializable](#))
    - o java.util.[ArrayList](#)<E> (implements java.lang.[Cloneable](#), java.util.[List](#)<E>, java.util.[RandomAccess](#), java.io.[Serializable](#))
    - o java.util.[Vector](#)<E> (implements java.lang.[Cloneable](#), java.util.[List](#)<E>, java.util.[RandomAccess](#), java.io.[Serializable](#)) **X**
    - o java.util.[Stack](#)<E>
  - o java.util.[AbstractQueue](#)<E> (implements java.util.[Queue](#)<E>)
    - o java.util.[PriorityQueue](#)<E> (implements java.io.[Serializable](#))
  - o java.util.[AbstractSet](#)<E> (implements java.util.[Set](#)<E>)
    - o java.util.[EnumSet](#)<E> (implements java.lang.[Cloneable](#), java.io.[Serializable](#))
    - o java.util.[HashSet](#)<E> (implements java.lang.[Cloneable](#), java.io.[Serializable](#), java.util.[Set](#)<E>)
    - o java.util.[LinkedHashSet](#)<E> (implements java.lang.[Cloneable](#), java.io.[Serializable](#), java.util.[Set](#)<E>)
    - o java.util.[TreeSet](#)<E> (implements java.lang.[Cloneable](#), java.util.[NavigableSet](#)<E>, java.io.[Serializable](#))
  - o java.util.[ArrayDeque](#)<E> (implements java.lang.[Cloneable](#), java.util.[Deque](#)<E>, java.io.[Serializable](#))
  - o java.util.[AbstractMap](#)<K,V> (implements java.util.[Map](#)<K,V>)
    - o java.util.[EnumMap](#)<K,V> (implements java.lang.[Cloneable](#), java.io.[Serializable](#))
    - o java.util.[HashMap](#)<K,V> (implements java.lang.[Cloneable](#), java.util.[Map](#)<K,V>, java.io.[Serializable](#))
    - o java.util.[LinkedHashMap](#)<K,V> (implements java.util.[Map](#)<K,V>)
    - o java.util.[IdentityHashMap](#)<K,V> (implements java.lang.[Cloneable](#), java.util.[Map](#)<K,V>, java.io.[Serializable](#))
    - o java.util.[TreeMap](#)<K,V> (implements java.lang.[Cloneable](#), java.util.[NavigableMap](#)<K,V>, java.io.[Serializable](#))
    - o java.util.[WeakHashMap](#)<K,V> (implements java.util.[Map](#)<K,V>)
  - o java.util.[AbstractMap.SimpleEntry](#)<K,V> (implements java.util.[Map.Entry](#)<K,V>, java.io.[Serializable](#))
  - o java.util.[AbstractMap.SimpleImmutableEntry](#)<K,V> (implements java.util.[Map.Entry](#)<K,V>, java.io.[Serializable](#))
  - o java.util.[Dictionary](#)<K,V>
  - o java.util.[Hashtable](#)<K,V> (implements java.lang.[Cloneable](#), java.util.[Map](#)<K,V>, java.io.[Serializable](#)) **X**
  - o java.util.[Properties](#)

### Colecciones en java.util.concurrent

- o java.lang.[Iterable](#)<T>
  - o java.util.[Collection](#)<E>
    - o java.util.concurrent.[BlockingQueue](#)<E>
    - o java.util.concurrent.[ConcurrentLinkedQueue](#)<E> (also extends java.util.[Deque](#)<E>)
    - o java.util.concurrent.[TransferQueue](#)<E>
    - o java.util.concurrent.[BlockingDeque](#)<E> (also extends java.util.concurrent.[BlockingQueue](#)<E>)
- o java.util.[Map](#)<K,V>
  - o java.util.concurrent.[ConcurrentMap](#)<K,V>
  - o java.util.concurrent.[ConcurrentNavigableMap](#)<K,V> (also extends java.util.[NavigableMap](#)<K,V>)
  - o java.util.[SortedMap](#)<K,V>
    - o java.util.[NavigableMap](#)<K,V>
    - o java.util.concurrent.[ConcurrentNavigableMap](#)<K,V> (also extends java.util.concurrent.[ConcurrentMap](#)<K,V>)

Aquí variantes TS de las colecciones en java.util

- o java.util.[AbstractCollection](#)<E> (implements java.util.[Collection](#)<E>)
- o java.util.concurrent.[ArrayBlockingQueue](#)<E> (implements java.util.concurrent.[BlockingQueue](#)<E>, java.io.[Serializable](#))
- o java.util.concurrent.[ConcurrentLinkedQueue](#)<E> (implements java.util.[Queue](#)<E>, java.io.[Serializable](#))
- o java.util.concurrent.[DelayQueue](#)<E> (implements java.util.concurrent.[BlockingQueue](#)<E>)
- o java.util.concurrent.[LinkedBlockingDeque](#)<E> (implements java.util.concurrent.[BlockingDeque](#)<E>, java.io.[Serializable](#))
- o java.util.concurrent.[LinkedBlockingQueue](#)<E> (implements java.util.concurrent.[BlockingQueue](#)<E>, java.io.[Serializable](#))
- o java.util.concurrent.[LinkedTransferQueue](#)<E> (implements java.util.concurrent.[TransferQueue](#)<E>)
- o java.util.concurrent.[PriorityBlockingQueue](#)<E> (implements java.util.concurrent.[BlockingQueue](#)<E>, java.io.[Serializable](#))
- o java.util.concurrent.[SynchronousQueue](#)<E> (implements java.util.concurrent.[BlockingQueue](#)<E>, java.io.[Serializable](#))
- o java.util.[AbstractSet](#)<E> (implements java.util.[Set](#)<E>)
- o java.util.concurrent.[ConcurrentSkipListSet](#)<E> (implements java.lang.[Cloneable](#), java.util.[NavigableSet](#)<E>, java.io.[Serializable](#))
- o java.util.concurrent.[CopyOnWriteArraySet](#)<E> (implements java.io.[Serializable](#))
- o java.util.concurrent.[ConcurrentLinkedDeque](#)<E> (implements java.util.[Deque](#)<E>, java.io.[Serializable](#))
- o java.util.concurrent.[AbstractExecutorService](#) (implements java.util.concurrent.[ExecutorService](#))
- o java.util.concurrent.[ForkJoinPool](#)
- o java.util.concurrent.[ThreadPoolExecutor](#)
  - o java.util.concurrent.[ScheduledThreadPoolExecutor](#) (implements java.util.concurrent.[ScheduledExecutorService](#))
- o java.util.[AbstractMap](#)<K,V> (implements java.util.[Map](#)<K,V>)
- o java.util.concurrent.[ConcurrentHashMap](#)<K,V> (implements java.util.concurrent.[ConcurrentMap](#)<K,V>, java.io.[Serializable](#))
- o java.util.concurrent.[ConcurrentSkipListMap](#)<K,V> (implements java.lang.[Cloneable](#), java.util.concurrent.[ConcurrentNavigableMap](#)<K,V>, java.io.[Serializable](#))