Package java.util

Contains the collections framework, legacy collection classes, event model, date and time facilities, internationalization, and miscellaneous utility classes (a string tokenizer, a random-number generator, and a bit array).

See: Description

包含集合框架、遗留的集合类、事件模型、日期和时间设施、国际化、
混杂实用工具类(字符串分割器、随机数生成器、位数组)

Interface Summary

Interface	Description
Collection<E>	The root interface in the *collection hierarchy*.
Comparator<T>	A comparison function, which imposes a *total ordering* on some collection of objects.
Deque<E>	A linear collection that supports element insertion and removal at both ends.
Enumeration<E>	An object that implements the Enumeration interface generates a series of elements, one at a time.
EventListener	A tagging interface that all event listener interfaces must extend.
Formattable	The `Formattable` interface must be implemented by any class that needs to perform custom formatting using the `'s'` conversion specifier of **`Formatter`**.
Iterator<E>	An iterator over a collection.
List<E>	An ordered collection (also known as a *sequence*).
ListIterator<E>	An iterator for lists that allows the programmer to traverse the list in either direction, modify the list during iteration, and obtain the iterator's current position in the list.
Map<K,V>	An object that maps keys to values.
Map.Entry<K,V>	A map entry (key-value pair).
NavigableMap<K,V>	A **`SortedMap`** extended with navigation methods returning the closest matches for given search targets.
NavigableSet<E>	A **`SortedSet`** extended with navigation methods reporting closest matches for given search targets.
Observer	A class can implement the `Observer` interface when it wants to be informed of changes in observable objects.
Queue<E>	A collection designed for holding elements prior to processing.
RandomAccess	Marker interface used by `List` implementations to indicate that they support fast (generally constant time) random access.
Set<E>	A collection that contains no duplicate elements.
SortedMap<K,V>	A **`Map`** that further provides a *total ordering* on its keys.
SortedSet<E>	A **`Set`** that further provides a *total ordering* on its elements.

集合层次结构中的根接口
映射表：将键映射到值
映射条目：键-值对
导航映射表
导航方法：返回给定搜索目标最接近的匹配
有序键映射表：提供基于其键的总排序

Class Summary

Class	Description
AbstractCollection<E>	This class provides a skeletal implementation of the `Collection` interface, to minimize the effort required to implement this interface.
AbstractList<E>	This class provides a skeletal implementation of the **`List`** interface to minimize the effort required to implement this interface backed by a "random access" data store (such as an array).
AbstractMap<K,V>	This class provides a skeletal implementation of the `Map` interface, to minimize the effort required to implement this interface.
AbstractMap.SimpleEntry<K,V>	An Entry maintaining a key and a value.
AbstractMap.SimpleImmutableEntry<K,V>	An Entry maintaining an immutable key and value.
AbstractQueue<E>	This class provides skeletal implementations of some **`Queue`** operations.
AbstractSequentialList<E>	This class provides a skeletal implementation of the `List` interface to minimize the effort required to implement this interface backed by a "sequential access" data store (such as a linked list).
AbstractSet<E>	This class provides a skeletal implementation of the `Set` interface to minimize the effort required to implement this interface.
ArrayDeque<E>	Resizable-array implementation of the **`Deque`** interface.

抽象映射表：提供Map接口的框架实现

ArrayList<E>	Resizable-array implementation of the `List` interface.
Arrays	This class contains various methods for manipulating arrays (such as sorting and searching).
BitSet	This class implements a vector of bits that grows as needed.
Calendar	The `Calendar` class is an abstract class that provides methods for converting between a specific instant in time and a set of **`calendar fields`** such as `YEAR`, `MONTH`, `DAY_OF_MONTH`, `HOUR`, and so on, and for manipulating the calendar fields, such as getting the date of the next week.
Collections	This class consists exclusively of static methods that operate on or return collections.
Currency	Represents a currency.
Date	The class `Date` represents a specific instant in time, with millisecond precision.
Dictionary<K,V>	The `Dictionary` class is the abstract parent of any class, such as `Hashtable`, which maps keys to values.
EnumMap<K extends **Enum**<K>,V> 枚举键映射表	A specialized **`Map`** implementation for use with enum type keys.
EnumSet<E extends **Enum**<E>>	A specialized **`Set`** implementation for use with enum types.
EventListenerProxy<T extends **EventListener**>	An abstract wrapper class for an `EventListener` class which associates a set of additional parameters with the listener.
EventObject	The root class from which all event state objects shall be derived.
FormattableFlags	FomattableFlags are passed to the **`Formattable.formatTo()`** method and modify the output format for **Formattables**.
Formatter	An interpreter for printf-style format strings.
GregorianCalendar	`GregorianCalendar` is a concrete subclass of `Calendar` and provides the standard calendar system used by most of the world.
HashMap<K,V> 哈希映射表：基于哈希表实现	Hash table based implementation of the `Map` interface.
HashSet<E>	This class implements the `Set` interface, backed by a hash table (actually a `HashMap` instance).
Hashtable<K,V>	This class implements a hash table, which maps keys to values.
IdentityHashMap<K,V> 标识哈希映射表：使用引用等价性	This class implements the `Map` interface with a hash table, using reference-equality in place of object-equality when comparing keys (and values).
LinkedHashMap<K,V> 链接哈希映射表：哈希表和链表实现，具有可预测的迭代顺序	Hash table and linked list implementation of the `Map` interface, with predictable iteration order.
LinkedHashSet<E>	Hash table and linked list implementation of the `Set` interface, with predictable iteration order.
LinkedList<E>	Doubly-linked list implementation of the `List` and `Deque` interfaces.
ListResourceBundle	`ListResourceBundle` is an abstract subclass of `ResourceBundle` that manages resources for a locale in a convenient and easy to use list.
Locale	A `Locale` object represents a specific geographical, political, or cultural region.
Locale.Builder	`Builder` is used to build instances of `Locale` from values configured by the setters.
Objects	This class consists of `static` utility methods for operating on objects.
Observable	This class represents an observable object, or "data" in the model-view paradigm.
PriorityQueue<E>	An unbounded priority **queue** based on a priority heap.
Properties	The `Properties` class represents a persistent set of properties.
PropertyPermission	This class is for property permissions.
PropertyResourceBundle	`PropertyResourceBundle` is a concrete subclass of `ResourceBundle` that manages resources for a locale using a set of static strings from a property file.
Random	An instance of this class is used to generate a stream of pseudorandom numbers.
ResourceBundle	Resource bundles contain locale-specific objects.
ResourceBundle.Control	`ResourceBundle.Control` defines a set of callback methods that are invoked by the **`ResourceBundle.getBundle`** factory methods during the bundle loading process.
Scanner	A simple text scanner which can parse primitive types and strings using regular expressions.
ServiceLoader<S>	A simple service-provider loading facility.
SimpleTimeZone	`SimpleTimeZone` is a concrete subclass of `TimeZone` that represents a time zone for use with a Gregorian calendar.

Stack<E>	The Stack class represents a last-in-first-out (LIFO) stack of objects.
StringTokenizer	The string tokenizer class allows an application to break a string into tokens.
Timer	A facility for threads to schedule tasks for future execution in a background thread.
TimerTask	A task that can be scheduled for one-time or repeated execution by a Timer.
TimeZone	TimeZone represents a time zone offset, and also figures out daylight savings.
TreeMap<K,V> 树映射表：基于红黑树实现的导航映射表	A Red-Black tree based NavigableMap implementation.
TreeSet<E>	A NavigableSet implementation based on a TreeMap.
UUID	A class that represents an immutable universally unique identifier (UUID).
Vector<E>	The Vector class implements a growable array of objects.
WeakHashMap<K,V> 弱引用键哈希映射表	Hash table based implementation of the Map interface, with *weak keys*.

Enum Summary

Enum	Description
Formatter.BigDecimalLayoutForm	
Locale.Category	Enum for locale categories.

Exception Summary

Exception	Description
ConcurrentModificationException	This exception may be thrown by methods that have detected concurrent modification of an object when such modification is not permissible.
DuplicateFormatFlagsException	Unchecked exception thrown when duplicate flags are provided in the format specifier.
EmptyStackException	Thrown by methods in the Stack class to indicate that the stack is empty.
FormatFlagsConversionMismatchException	Unchecked exception thrown when a conversion and flag are incompatible.
FormatterClosedException	Unchecked exception thrown when the formatter has been closed.
IllegalFormatCodePointException	Unchecked exception thrown when a character with an invalid Unicode code point as defined by Character.isValidCodePoint(int) is passed to the Formatter.
IllegalFormatConversionException	Unchecked exception thrown when the argument corresponding to the format specifier is of an incompatible type.
IllegalFormatException	Unchecked exception thrown when a format string contains an illegal syntax or a format specifier that is incompatible with the given arguments.
IllegalFormatFlagsException	Unchecked exception thrown when an illegal combination flags is given.
IllegalFormatPrecisionException	Unchecked exception thrown when the precision is a negative value other than -1, the conversion does not support a precision, or the value is otherwise unsupported.
IllegalFormatWidthException	Unchecked exception thrown when the format width is a negative value other than -1 or is otherwise unsupported.
IllformedLocaleException	Thrown by methods in Locale and Locale.Builder to indicate that an argument is not a well-formed BCP 47 tag.
InputMismatchException	Thrown by a Scanner to indicate that the token retrieved does not match the pattern for the expected type, or that the token is out of range for the expected type.
InvalidPropertiesFormatException	Thrown to indicate that an operation could not complete because the input did not conform to the appropriate XML document type for a collection of properties, as per the Properties specification.
MissingFormatArgumentException	Unchecked exception thrown when there is a format specifier which does not have a corresponding argument or if an argument index refers to an argument that does not exist.
MissingFormatWidthException	Unchecked exception thrown when the format width is required.
MissingResourceException	Signals that a resource is missing.
NoSuchElementException	Thrown by the nextElement method of an Enumeration to indicate that there are no more elements in the enumeration.
TooManyListenersException	The TooManyListenersException Exception is used as part of the Java Event model to annotate and implement a unicast special case of a multicast Event Source.

UnknownFormatConversionException	Unchecked exception thrown when an unknown conversion is given.
UnknownFormatFlagsException	Unchecked exception thrown when an unknown flag is given.

Error Summary

Error	Description
ServiceConfigurationError	Error thrown when something goes wrong while loading a service provider.

Package java.util Description

Contains the collections framework, legacy collection classes, event model, date and time facilities, internationalization, and miscellaneous utility classes (a string tokenizer, a random-number generator, and a bit array).

Package Specification

- Collections Framework Overview
- Collections Framework Annotated Outline

Related Documentation

For overviews, tutorials, examples, guides, and tool documentation, please see:

- Collections Framework Tutorial
- Collections Framework Design FAQ

Since:

JDK1.0