Q)what are the predifined functions in collection in java?

|  |  |
| --- | --- |
| **Methods** | **Description** |
| [addAll(Collection<? extends E> c)](https://www.google.com/url?client=internal-element-cse&cx=009682134359037907028:tj6eafkv_be&q=https://www.geeksforgeeks.org/collections-addall-method-in-java-with-examples/&sa=U&ved=2ahUKEwjKuPOa-NnsAhWF6XMBHV52Bxg4ChAWMAV6BAgGEAI&usg=AOvVaw0xyjH8UBNWNsFmLyTyUlZk) | It is used to insert the specified collection elements in the invoking collection. |
| [asLifoQueue​(Deque<T> deque)](https://www.google.com/url?client=internal-element-cse&cx=009682134359037907028:tj6eafkv_be&q=https://www.geeksforgeeks.org/collections-aslifoqueue-method-in-java-with-examples/&sa=U&ved=2ahUKEwjR2pzb-NnsAhXB63MBHfsmCzQ4HhAWMAR6BAgEEAI&usg=AOvVaw0cqoFANseci3YfUQzT91qJ) | This method returns a view of a Deque as a Last-in-first-out (Lifo) Queue. |
| [binarySearch(List<? extends Comparable> list, T key)](https://www.google.com/url?client=internal-element-cse&cx=009682134359037907028:tj6eafkv_be&q=https://www.geeksforgeeks.org/collections-binarysearch-java-examples/&sa=U&ved=2ahUKEwi6uP_199nsAhXj73MBHZbDDPMQFjAJegQIBRAC&usg=AOvVaw3Z6NwOyP0de-c-hRasRUsf) | This method searches the key using binary search in the specified list. |
| [binarySearch​(List<? extends T> list, T key, Comparator<? super T> c)](https://www.google.com/url?client=internal-element-cse&cx=009682134359037907028:tj6eafkv_be&q=https://www.geeksforgeeks.org/collections-binarysearch-java-examples/&sa=U&ved=2ahUKEwi6uP_199nsAhXj73MBHZbDDPMQFjAJegQIBRAC&usg=AOvVaw3Z6NwOyP0de-c-hRasRUsf) | This method searches the specified list for the specified object using the binary search algorithm. |
| [checkedCollection​(Collection<E> c, Class<E> type)](https://www.google.com/url?client=internal-element-cse&cx=009682134359037907028:tj6eafkv_be&q=https://www.geeksforgeeks.org/collections-checkedcollection-method-in-java-with-examples/&sa=U&ved=2ahUKEwjR2pzb-NnsAhXB63MBHfsmCzQ4HhAWMAJ6BAgJEAI&usg=AOvVaw1gjbCCS3zLq1m9WpQf0BAH) | This method returns a dynamically typesafe view of the specified collection. |
| checkedList​(List<E> list, Class<E> type) | This method returns a dynamically typesafe view of the specified list. |
| checkedMap​(Map<K,​V> m, Class<K> keyType, Class<V> valueType) | This method returns a dynamically typesafe view of the specified map. |
| checkedNavigableMap​(NavigableMap<K,​V> m, Class<K> keyType, Class<V> valueType) | This method returns a dynamically typesafe view of the specified navigable map. |
| checkedNavigableSet​(NavigableSet<E> s, Class<E> type) | This method returns a dynamically typesafe view of the specified navigable set. |
| checkedQueue​(Queue<E> queue, Class<E> type) | This method returns a dynamically typesafe view of the specified queue. |
| checkedSet​(Set<E> s, Class<E> type) | This method returns a dynamically typesafe view of the specified set. |
| checkedSortedMap​(SortedMap<K,​V> m, Class<K> keyType, Class<V> valueType) | This method returns a dynamically typesafe view of the specified sorted map. |
| checkedSortedSet​(SortedSet<E> s, Class<E> type) | This method returns a dynamically typesafe view of the specified sorted set. |
| [copy​(List<? super T> dest, List<? extends T> src)](https://www.google.com/url?client=internal-element-cse&cx=009682134359037907028:tj6eafkv_be&q=https://www.geeksforgeeks.org/collections-copy-method-in-java-with-examples/&sa=U&ved=2ahUKEwjJkK-8-NnsAhWe7HMBHWaOAiA4FBAWMAF6BAgHEAI&usg=AOvVaw0dFOeP7RLXpc_ORg_6Wj2U) | This method copies all of the elements from one list into another. |
| [disjoint​(Collection<?> c1, Collection<?> c2)](https://www.google.com/url?client=internal-element-cse&cx=009682134359037907028:tj6eafkv_be&q=https://www.geeksforgeeks.org/java-util-collections-disjoint-method-java-examples/&sa=U&ved=2ahUKEwjKuPOa-NnsAhWF6XMBHV52Bxg4ChAWMAF6BAgIEAI&usg=AOvVaw3c8uSUuGg43wQFmZCF_Bj4) | This method returns true if the two specified collections have no elements in common. |
| emptyEnumeration() | This method returns an enumeration that has no elements. |
| emptyIterator() | This method returns an iterator that has no elements. |
| emptyList() | This method returns an empty list (immutable). |
| emptyListIterator() | This method returns a list iterator that has no elements. |
| emptyMap() | This method returns an empty map (immutable). |
| emptyNavigableMap() | This method returns an empty navigable map (immutable). |
| emptyNavigableSet() | This method returns an empty navigable set (immutable). |
| emptySet() | This method returns an empty set (immutable). |
| emptySortedMap() | This method returns an empty sorted map (immutable). |

Q) what are the underlying ds for linked list arraylist in java?

ArrayList is a resizable array implementation in java. The backing data structure of ArrayList is **an array of Object class**. When creating an ArrayList you can provide initial capacity then the array is declared with the given capacity. The default capacity value is 10.

Q) what are the advantages and disadvatages of array list?

***Advantages:***

1. ArrayList is variable length.
2. Add any type of data into ArrayList.
3. Traverse in both directions.
4. Insert and remove elements also at particular position of ArrayList.
5. ArrayList allows multiple null values.
6. ArrayList allows to add duplicate elements.
7. ArrayList has many methods to manipulate stored objects.
8. When ArrayList exceeds its capacity, then its size increases by **50%.**
9. Retrieval is faster in ArrayList

***Disadvantages:***

1. If a data entry is added to or removed from an array-based list, data needs to be shifted to update the list.

Q) What are iterators and cursors?

**A Java Cursor is an Iterator, which is used to iterate or traverse or retrieve a Collection or Stream object's elements one by one**. Java supports the following four different cursors. Enumeration.

Q)what are the 10 owasp?

The Top 10 OWASP vulnerabilities in 2021 are:

* Injection
* Broken authentication
* Sensitive data exposure
* XML external entities (XXE)
* Broken access control
* Security misconfigurations
* Cross site scripting (XSS)
* Insecure deserialization
* Using components with known vulnerabilities
* Insufficient logging

Q)can catch be return without try in java?

Yes,  we can have try without catch block by using finally block.

You can use try with finally. As you know finally block always executes even if you have exception or return statement in try block except in case of System.exit().

Q)can try be return catch in java?

Whenever try-block executes successfully, then it can always return value for this method. But if any exception is raised & it is handled in the corresponding catch-block –> return statement at the end of method will be executed and returns the value for this method after executing finally-block.

Q)can finally be return without try catch in java?

**Yes, we can have try without catch block by using finally block**. You can use try with finally. As you know finally block always executes even if you have exception or return statement in try block except in case of System.

9)diffarence b/w comparable and comparator in java?

**Comparable in Java is an object to compare itself with another object, whereas Comparator is an object for comparing different objects of different classes**. Comparable provides compareTo() method to sort elements in Java whereas Comparator provides compare() method to sort elements in Java.