

# Assume you are expert in java programming and a greate problem solver knows all the linkedlist methods then provide list all the methods present in linkedlist without leaving anything

Absolutely! Here's the **complete list of all methods** provided by the `LinkedList` class in Java, assuming **Java SE 17** and beyond. `LinkedList` is a class in the `java.util` package and implements **List**, **Deque**, **Queue**, **Cloneable**, and **Serializable** interfaces.

---

## ✓ Constructor

```
LinkedList<E> list = new LinkedList<>();  
LinkedList<E> list = new LinkedList<>(Collection<? extends E> c);
```

---

## 📦 Methods Inherited from List, Deque, and Queue

---

### 📌 Basic List Operations

| Method                                  | Description                                    |
|---|--|
| <code>int size()</code>                 | Returns the number of elements.                |
| <code>boolean isEmpty()</code>          | Checks if the list is empty.                   |
| <code>boolean contains(Object o)</code> | Checks if an element is in the list.           |
| <code>boolean add(E e)</code>           | Adds element to the end.                       |
| <code>boolean remove(Object o)</code>   | Removes first occurrence of specified element. |
| <code>void clear()</code>               | Clears the list.                               |
| <code>E get(int index)</code>           | Gets element at specified index.               |

|   |  |
|---|--|
| <code>E set(int index, E element)</code>    | Replaces element at index.             |
| <code>void add(int index, E element)</code> | Adds element at specific index.        |
| <code>E remove(int index)</code>            | Removes element at index.              |
| <code>int indexOf(Object o)</code>          | First index of element.                |
| <code>int lastIndexOf(Object o)</code>      | Last index of element.                 |
| <code>Object[] toArray()</code>             | Returns array containing all elements. |
| <code>&lt;T&gt; T[] toArray(T[] a)</code>   | Returns typed array.                   |

---

## Iterator Operations

| Method   | Description                       |
|--|-----------------------------------|
| <code>Iterator&lt;E&gt; iterator()</code>                  | Returns an iterator.              |
| <code>ListIterator&lt;E&gt; listIterator()</code>          | Returns list iterator.            |
| <code>ListIterator&lt;E&gt; listIterator(int index)</code> | Returns list iterator from index. |
| <code>Splitter&lt;E&gt; splitter()</code>                  | For stream processing.            |

---

## Queue & Deque Methods

| Method                          | Description                       |
|---------------------------------|-----------------------------------|
| <code>boolean offer(E e)</code> | Inserts at the end (queue).       |
| <code>E remove()</code>         | Removes head (throws if empty).   |
| <code>E poll()</code>           | Removes head or returns null.     |
| <code>E element()</code>        | Retrieves head (throws if empty). |

`E peek()`      Retrieves head or null.

---

## Deque Specific Methods

| Method   | Description  |
|--|--|
| <code>void addFirst(E e)</code>                          | Inserts at beginning.                                |
| <code>void addLast(E e)</code>                           | Inserts at end.                                      |
| <code>boolean offerFirst(E e)</code>                     | Offers at beginning.                                 |
| <code>boolean offerLast(E e)</code>                      | Offers at end.                                       |
| <code>E removeFirst()</code>                             | Removes first element.                               |
| <code>E removeLast()</code>                              | Removes last element.                                |
| <code>E pollFirst()</code>                               | Retrieves and removes first element, or null.        |
| <code>E pollLast()</code>                                | Retrieves and removes last element, or null.         |
| <code>E getFirst()</code>                                | Gets first element.                                  |
| <code>E getLast()</code>                                 | Gets last element.                                   |
| <code>E peekFirst()</code>                               | Peeks first element.                                 |
| <code>E peekLast()</code>                                | Peeks last element.                                  |
| <code>boolean<br/>removeFirstOccurrence(Object o)</code> | Removes first occurrence.                            |
| <code>boolean<br/>removeLastOccurrence(Object o)</code>  | Removes last occurrence.                             |
| <code>void push(E e)</code>                              | Pushes onto stack (same as <code>addFirst</code> ).  |
| <code>E pop()</code>                                     | Pops from stack (same as <code>removeFirst</code> ). |
| <code>boolean offer(E e)</code>                          | Queue insert at tail.                                |

---

## Bulk Operations

| Method  | Description                           |
|---|---------------------------------------|
| <code>boolean addAll(Collection&lt;? extends E&gt; c)</code>            | Adds all elements from collection.    |
| <code>boolean addAll(int index, Collection&lt;? extends E&gt; c)</code> | Adds all at index.                    |
| <code>boolean containsAll(Collection&lt;?&gt; c)</code>                 | Checks if list contains all elements. |
| <code>boolean removeAll(Collection&lt;?&gt; c)</code>                   | Removes all matching.                 |
| <code>boolean retainAll(Collection&lt;?&gt; c)</code>                   | Keeps only matching.                  |
| <code>void replaceAll(UnaryOperator&lt;E&gt; operator)</code>           | Replaces all elements using function. |
| <code>void sort(Comparator&lt;? super E&gt; c)</code>                   | Sorts list.                           |

---

## Advanced Functional Methods (Java 8+)

| Method   | Description                      |
|--|----------------------------------|
| <code>void forEach(Consumer&lt;? super E&gt; action)</code>      | Performs action on each element. |
| <code>Stream&lt;E&gt; stream()</code>                            | Returns sequential stream.       |
| <code>Stream&lt;E&gt; parallelStream()</code>                    | Returns parallel stream.         |
| <code>boolean removeIf(Predicate&lt;? super E&gt; filter)</code> | Removes if condition met.        |

---

## Cloning and Serialization

| Method                      | Description             |
|-----------------------------|-------------------------|
| <code>Object clone()</code> | Returns a shallow copy. |

|  |                                 |
|--|---------------------------------|
| <code>void</code>                              | For serialization (internal).   |
| <code>writeObject(ObjectOutputStream s)</code> |                                 |
| <code>void</code>                              | For deserialization (internal). |
| <code>readObject(ObjectInputStream s)</code>   |                                 |

---

## Miscellaneous

| Method   | Description                  |
|--|------------------------------|
| <code>List&lt;E&gt; subList(int fromIndex, int toIndex)</code> | Returns portion of the list. |
| <code>boolean equals(Object o)</code>                          | Checks equality.             |
| <code>int hashCode()</code>                                    | Returns hash code.           |

---

If you want **Java code examples** for each method or a **LinkedList cheat sheet project**, I'd be happy to help you build that too.

Would you like a **Java notebook** or **project file** with working examples for all methods?