

JAVA.UTIL.ARRAYDEQUE.REMOVEOBJECT METHOD

http://www.tutorialspoint.com/java/util/arraydeque_remove_object.htm

Copyright © tutorialspoint.com

Description

The **java.util.ArrayDeque.removeObject** method removes a single instance of the specified **element** from this **deque**.

Declaration

Following is the declaration for **java.util.ArrayDeque.removeObject** method

```
public boolean removeObject(Object o)
```

Parameters

- **o** -- The element to be removed from this deque, if present

Return Value

This method returns **true** if this deque contains the specified element.

Exception

- NA

Example

The following example shows the usage of **java.util.ArrayDeque.removeObject** method.

```
package com.tutorialspoint;

import java.util.ArrayDeque;
import java.util.Deque;

public class ArrayDequeDemo {
    public static void main(String[] args) {

        // create an empty array deque with an initial capacity
        Deque<Integer> deque = new ArrayDeque<Integer>(8);

        // use add() method to add elements in the deque
        deque.add(25);
        deque.add(30);
        deque.add(20);
        deque.add(40);

        // printing all the elements available in deque
        for (Integer number : deque) {
            System.out.println("Number = " + number);
        }

        // deque contains element 30, returns true
        boolean retval = deque.removeObject(30);
        if (retval == true) {
            System.out.println("element 30 is removed from the deque");
        }
        else {
            System.out.println("element 30 is not contained in the deque");
        }

        // deque does not contain element 15, returns false
        boolean retval = deque.removeObject(15);
        if (retval == true) {
```

```
System.out.println("element 15 is removed from the deque");
}
else {
System.out.println("element 15 is not contained in the deque");
}

// printing remaining elements
for (Integer number : deque) {
System.out.println("Number = " + number);
}
}
}
```

Let us compile and run the above program, this will produce the following result:

```
Number = 25
Number = 30
Number = 20
Number = 40
element 30 is removed from the deque
element 15 is not contained in the deque
Number = 25
Number = 20
Number = 40
```

Loading [Mathjax]/jax/output/HTML-CSS/jax.js