



Iterators

- Java iterator, collection elemanlarımızın arasında gezinmemize ve elemanları değiştirmemize yarar.
- Collections'da her element index yapısını desteklemez, index olmadan tüm elementlere ulaşmak için for-each loop kullanabiliriz ancak for-each loop ile elementleri kalıcı olarak değiştirme veya silme imkanı olmadığı için Iterator kullanımını tercih ederiz.

Method Summary

All Methods

Instance Methods

Abstract Methods

Default Methods

Modifier and Type

Method and Description

default void

forEachRemaining(Consumer<? super E> action)

Performs the given action for each remaining element until all elements have been processed or the action throws an exception.

boolean

hasNext()

Returns true if the iteration has more elements.

E

next()

Returns the next element in the iteration.

default void

remove()

Removes from the underlying collection the last element returned by this iterator (optional operation).



Iterators

Iterator method'lari (turkce aciklama ile)

Sr.No.	Metod ve Açıklama
1	boolean hasNext() Hala dizide eleman varsa true döner değilse false döner
2	Object next() Bir sonraki elemanı döndürür
3	void remove() O anki elemanı silmeye yarar.

Soru 1) Bir listedeki tüm elementleri iterator kullanarak silin.

```
public static void main(String[] args) {  
    List<String> list = new ArrayList<>();  
  
    list.add("A");  
    list.add("B");  
    list.add("C");  
  
    System.out.println(list);  
  
    Iterator<String> li1=list.iterator();  
    while(li1.hasNext()) {  
        li1.next();  
        li1.remove();  
    }  
    System.out.println(list);  
}
```



ListIterators

NOT : ListIterator daha fazla method'a sahip olduğu için daha çok kullanılır.

ListIterator, Iterator interface'in child'idir.

Method and Description
<code>add(E e)</code> Inserts the specified element into the list (optional operation).
<code>hasNext()</code> Returns true if this list iterator has more elements when traversing the list in the forward direction.
<code>hasPrevious()</code> Returns true if this list iterator has more elements when traversing the list in the reverse direction.
<code>next()</code> Returns the next element in the list and advances the cursor position.
<code>nextIndex()</code> Returns the index of the element that would be returned by a subsequent call to <code>next()</code> .
<code>previous()</code> Returns the previous element in the list and moves the cursor position backwards.
<code>previousIndex()</code> Returns the index of the element that would be returned by a subsequent call to <code>previous()</code> .
<code>remove()</code> Removes from the list the last element that was returned by <code>next()</code> or <code>previous()</code> (optional operation).
<code>set(E e)</code> Replaces the last element returned by <code>next()</code> or <code>previous()</code> with the specified element (optional operation).

Iterator ve ListIterator arasındaki en büyük fark,
Iterator'ın koleksiyondaki öğeleri yalnızca ileri yönde hareket ettirebilmesidir
ListIterator ise bir koleksiyondaki öğeleri hem ileri hem de geri yönde hareket ettirebilir .



ListIterators

Soru 2) String bir listedeki tüm elementlerin sonuna X ekleyin

For-Each Loop ile

```
List<String> list = new ArrayList<>();

list.add("A");
list.add("B");
list.add("C");

System.out.println(list);

for (String each : list) {
    each=each+"x";
}

System.out.println(list);
```

ListIterator ile

```
public static void main(String[] args) {

    List<String> list = new ArrayList<>();

    list.add("A");
    list.add("B");
    list.add("C");

    System.out.println(list);

    ListIterator<String> li1=list.listIterator();
    while(li1.hasNext()) {
        String str=li1.next();
        li1.set(str+"X");
    }
    System.out.println(list);
}
```



Iterators

Soru 3) Bir listedeki istenen sayi araliginda olmayan elementleri silen bir program yaziniz ...
(2. liste olusturmadan, gecerli liste uzerinde islem yapiniz)

Orn : [2,13,56,23,45,14,40] istenen aralik 20 ile 40 arasi (sinirlar dahil)
output: [23,40]

Soru 4) Bir listedeki elementleri iterator kullanarak sondan basa dogru yazdirin

Soru 5) (iterator ile index kullanimina ornek) Bir listedeki ilk n elemani iterator kullanarak 5 artirin.

Orn : list : [2,13,56,23,45,14,40]
artirilmesi istenen eleman sayisi : 3
output: [7,18,61,23,45,14,40]