



ITCS214

Data Structures

ArrayList

All methods with brief explain through drawing . Examples . solve questions from previous exams
ملخص لجميع الميثود مع التوضيح من خلال الرسم . أمثلة متعددة . أسئلة محلولة من امتحانات سابقة

This note is copyrighted material for enrolled students only.

جميع الحقوق محفوظة، هذه المذكورة مخصصة حصرياً للطلاب المسجلين معنا

Purchase access for non-enrolled users through our website.

إذا كنت غير مسجل معنا يمكنك شراء المذكورة من خلال موقعنا الإلكتروني

لتنزيل المذكورة

itcs214.olearninga.com

ArrayList :

- In java there is a built-in class called **ArrayList**.
- To import this class: **import java.util.ArrayList;**

- To create an ArrayList object called **list** that will store **Integer**:
ArrayList<Integer> list = new ArrayList <Integer> ();

or

```
ArrayList<Integer> list = new ArrayList <> ();
```

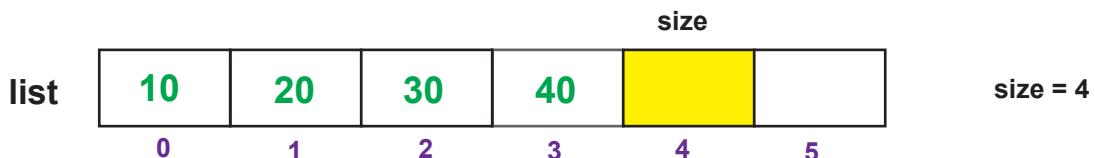
or simply

```
ArrayList<Integer> list = new ArrayList();
```

In the ArrayList class we have the following methods:

1. int size ()

This method returns the actual number of elements in the list.

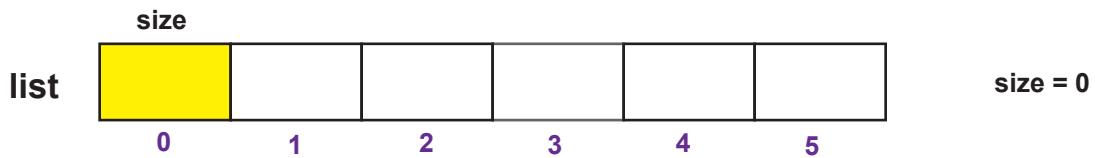


```
int s = list.size();
```



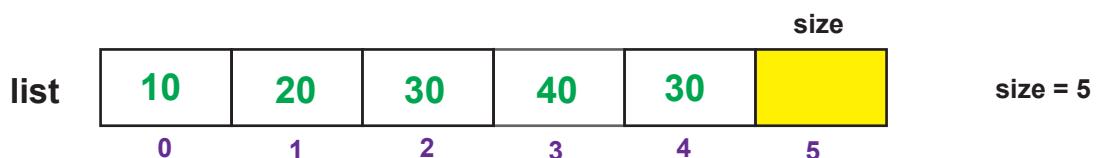
2. boolean isEmpty ()

This method to check whether the list is empty or not.



boolean l1= list.isEmpty();

true



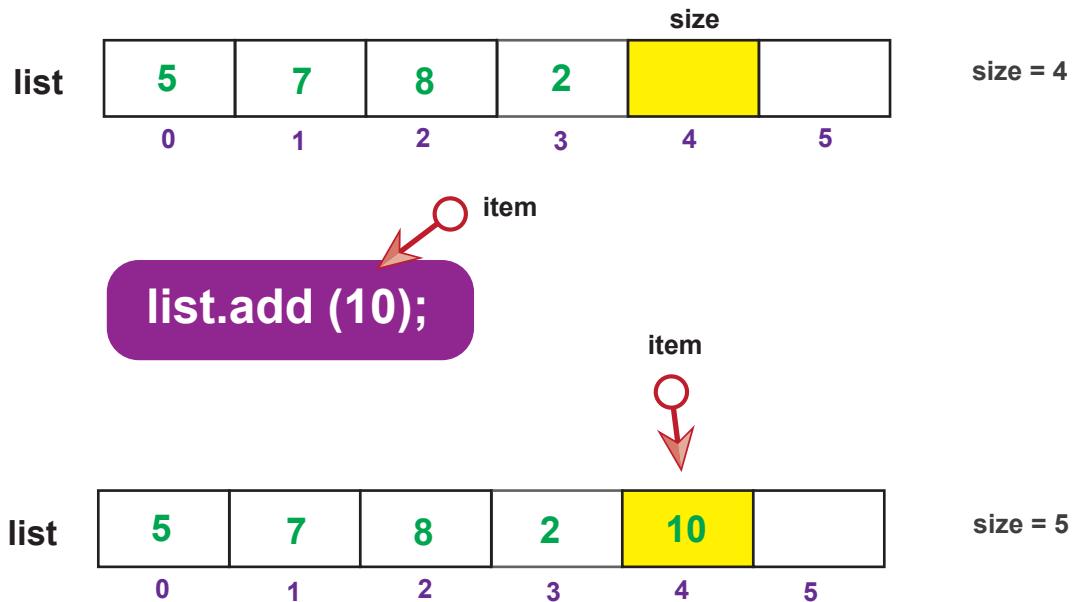
boolean l2= list.isEmpty();

false

3. boolean add (E item)

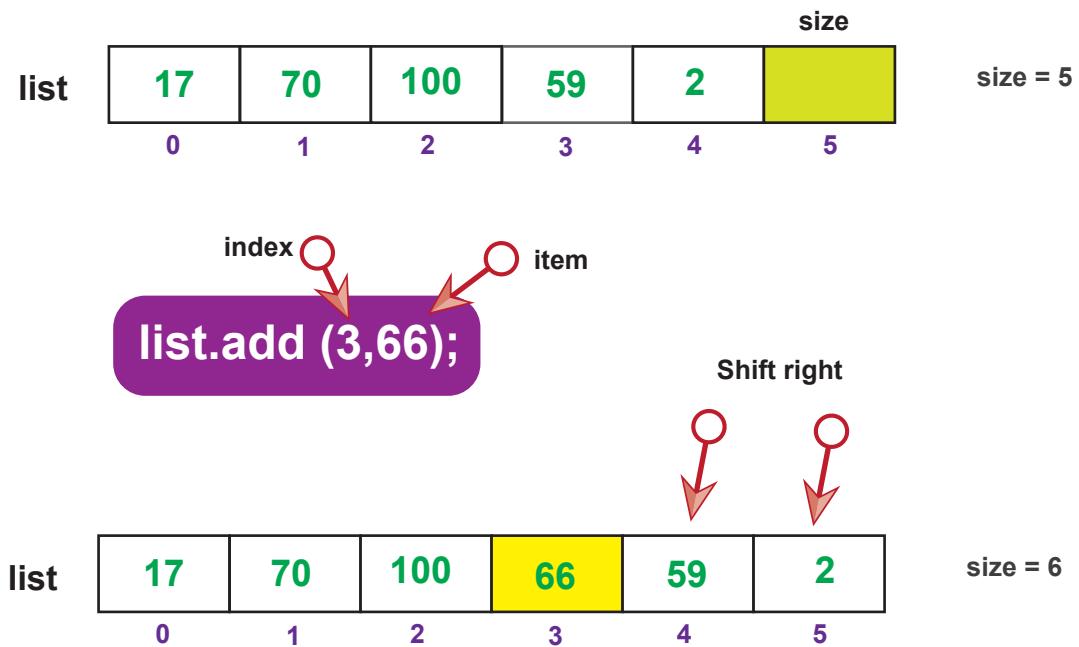
E means generic data type like (int, double,...)

It add item at the end of the list, and always returns true.



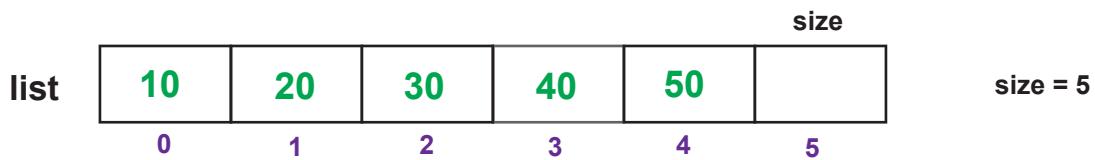
4. void add (int index, E item)

This method adds item at the location index.

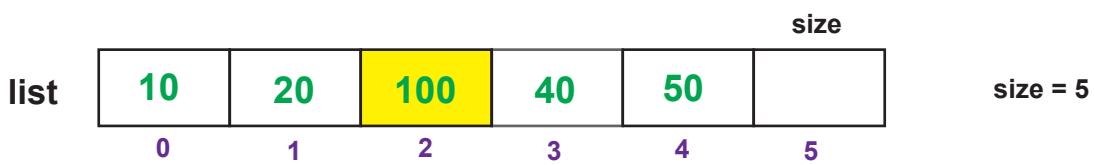


5. E set (int index, E item)

Replace the element at position index with item and return the old value.

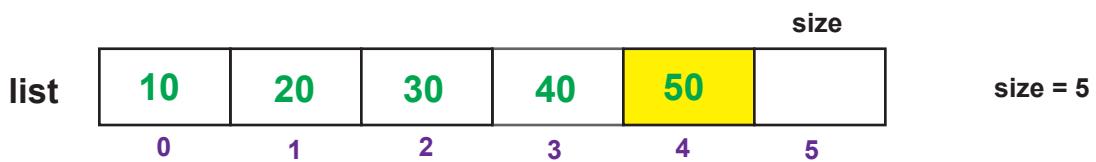


int oldItem= list.set(2,100);



6. E get (int index)

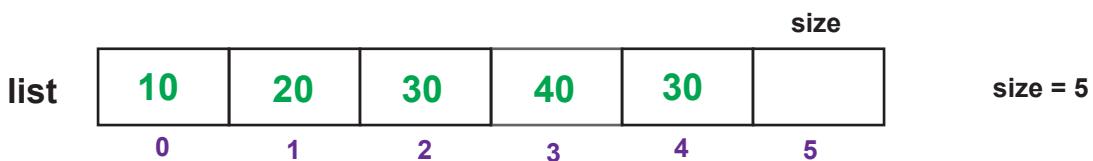
Returns the element at the position index.



int item = list.get(4);

7. int indexOf (E item)

Searches for item in the list and returns the position of the first occurrence, or -1 if it is not found in the list.



`int item1 = list.indexOf(10);`

0



`int item2 = list.indexOf(30);`

2



first occurrence of the item in the list

`int item3 = list.indexOf(100);`

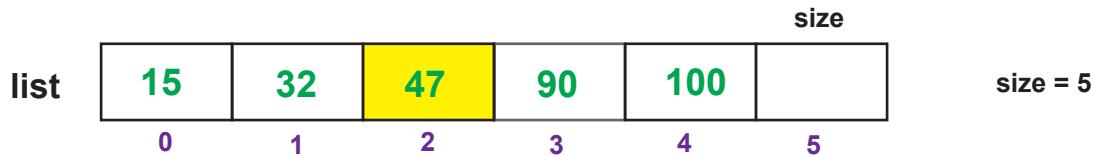
-1



the item not found in the list

8. E remove (int index)

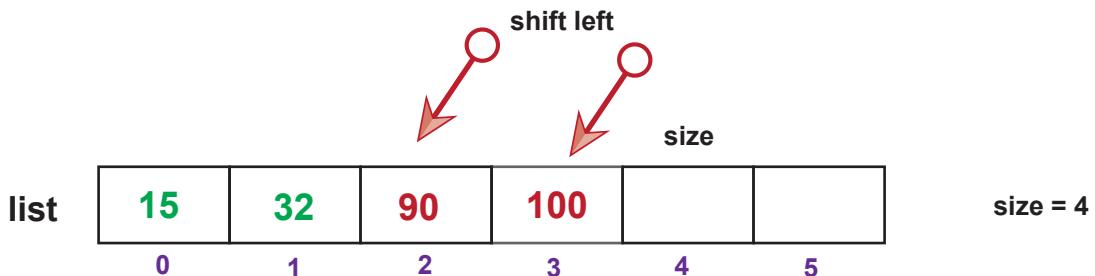
Removes the element at the given index and return it.



int item = list.remove(2);

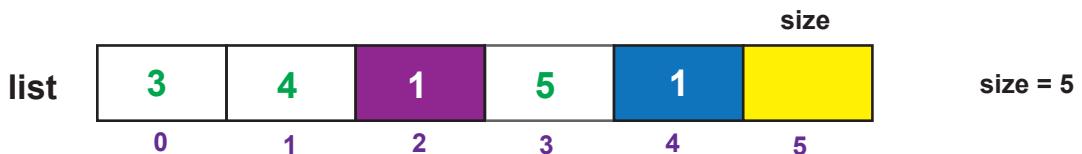
47

index



9. boolean remove (E item)

This method remove the first occurrence of the item from the list and return true.

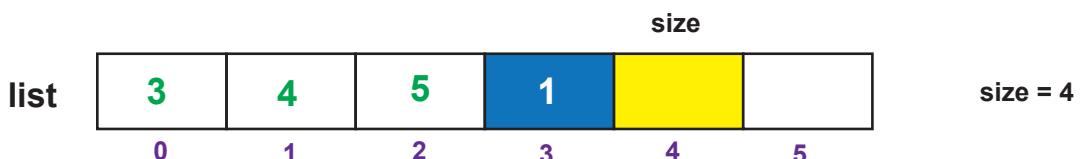


Remove with the argument (Integer), which is Integer object. This is necessary because the remove method takes an object as an argument, not a primitive int.

first occurrence of the item

boolean i1 = list.remove((Integer)1);

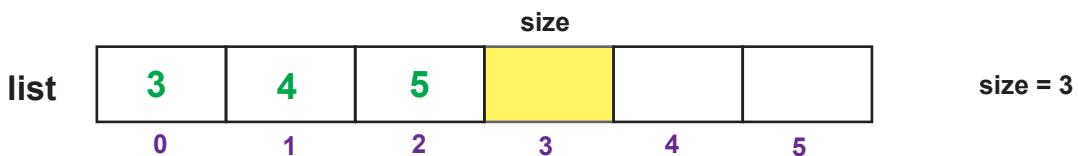
true



or another way:

boolean i2= list.remove((Integer.valueOf(1)));

true



item not found

boolean i3 = list.remove((Integer)2);

false

(1) What is the output of the following code?

```
import java.util.ArrayList;
public class Test {
    public static void main(String[] args) {

        ArrayList<Integer> list = new ArrayList<>();
        System.out.println(list.isEmpty());
        list.add(7);
        list.add(3);
        list.add(4);
        list.add(6);
        list.add(0,7);
        list.set(2,list.indexOf(7));
        list.remove(1);
        for (int i=0; i<list.size(); i++)
            System.out.println(list.get(i));

    }
}
```

Output

```
true
7
0
4
6
```

(2) What is the output of the following code?

```
ArrayList<Integer> list = new ArrayList<>();
for (int i=0; i<4; i++)
    list.add(i,i+1);

System.out.println(list.get(1));
System.out.println(list.size());
list.set(1,100);
list.add(1,100);
list.remove((Integer)1);
for(int i=0; i<list.size(); i++)
    System.out.print(list.get(i) + "\t");
```

Output

Output
2
4
100 100 3 4