Arrays

String

BS

Searching

ArrayList

Stack

Queue

LinkedList

OOPS

HashMap

HashSet

Arrays

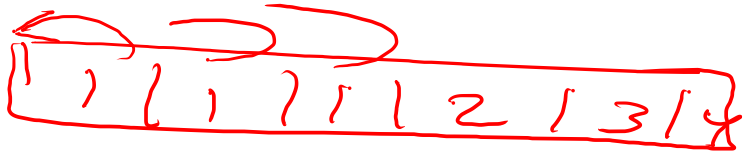
1. We have to always give the array size

```
int[] arr = new int[size];
```

2. Fixed

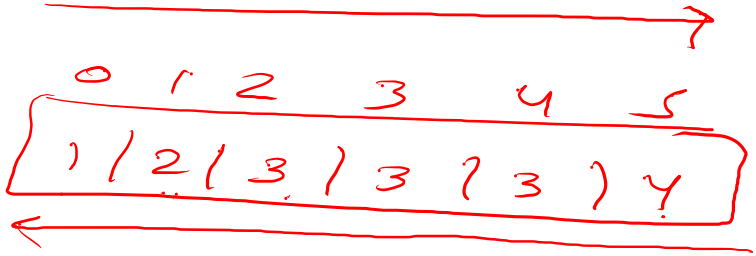
3. Predefined function

ArrayList



1. ArrayList is a predefined class in java
2. It comes from java.util
3. Not required to mention size while creating the ArrayList
4. Growable
5. Allow duplicates data

(6)



ans 1 = 2

ans = 4

K = 3

+ Index of
+ last Index of

{2, 4}

arr. IndexOf(3) ↓

arr. lastIndexOf ↓

arr[-3] = -1

arr[5] = -1

Creating ArrayList

1. ArrayList objName = new ArrayList(); // Type unsafe ArrayList
2. ArrayList<className> objName = new ArrayList<>(); //Typesafe ArrayList

int

int a = 10

a.

Integer a = 10;

a.

0	1	2
10	"Sunday"	true

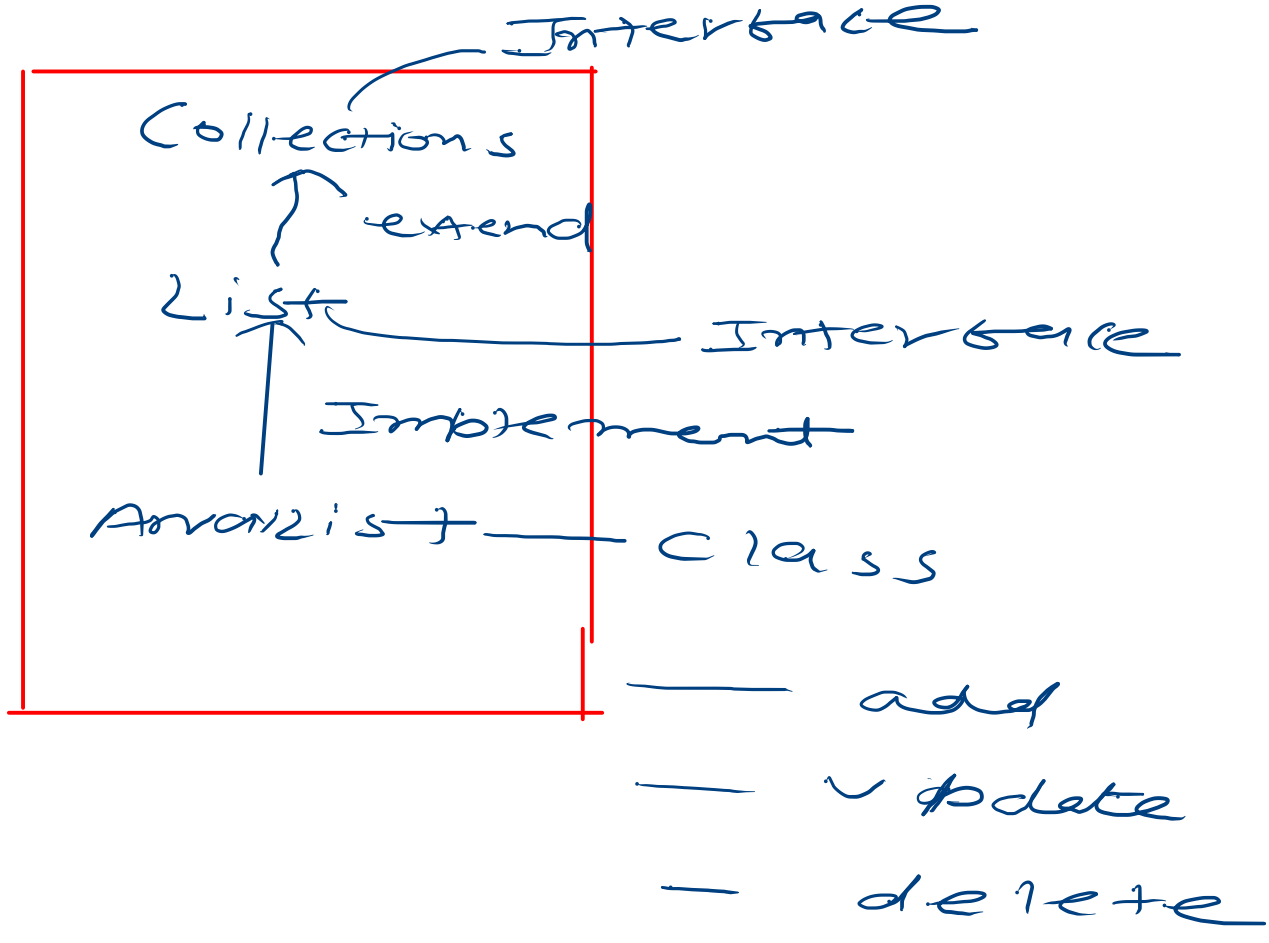
num.add(10);

num.add("Sunday");

num.add(true);

int a = num.get(0);

int a = (int) num.get(0);


Interface
Collections
extend
List
Interface
Implement
ArrayList
Class
add
update
delete

- **case 1**: Print the size of the ArrayList in a separate line.
- **case 2**: Print and Remove element from the last index of the ArrayList.
- **case 3**: Print x and Add x in last index of the ArrayList.
- **case 4**: Print and Remove an element from the starting (index = 0) of the ArraList.
- **case 5**: Print x and Add x at beginning (index = 0) of the ArrayList.
- **case 6**: Print all the elements from left to right that are there inside the ArrayList.

```
8
2
6
3 2
5 1
6
1
3 3
2
```

2, 4, 6

while(t--> 0){

invalid move

invalid move

2

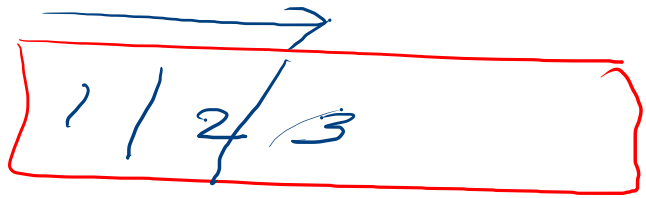
1

1, e

2

3

3



ArrayList with if-else

```java
import java.text.*;
import java.math.*;
import java.util.regex.*;

public class Solution {

    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
        Scanner sc = new Scanner(System.in);
        int t = sc.nextInt();
        ArrayList<Integer> arr = new ArrayList();
        for(int i=1;i<=t;i++){
            int ca = sc.nextInt();
            if(ca==1){
                System.out.println(arr.size());
            }else if(ca==2){
                if(arr.size()==0){
                    System.out.println("invalid-move");
                }else{
                    int ans = arr.remove(arr.size()-1);
                    System.out.println(ans);
                }
            }else if(ca==3){
                int x = sc.nextInt();
                System.out.println(x);
                arr.add(x);

            }else if(ca==4){
                 if(arr.size()==0){
                    System.out.println("invalid-move");
                }else{
                    int ans = arr.remove(0);
                    System.out.println(ans);
                }

            }else if(ca==5){
                int x = sc.nextInt();
                System.out.println(x);
                arr.add(0,x);

            }else if(ca==6){
                if(arr.size()==0){
                    System.out.println("invalid-move");
                }
                else{
                    for(int j=0;j<arr.size();j++){
                        System.out.print(arr.get(j)+" ");
                    }
                    System.out.println();
                }
            }
        }
```

(1) Has a huge amount of predefined methods for performing various operations like search, delete, update, add etc