

```

1  package StackAndQueue;
2
3  import java.util.Arrays;
4
5  class ImplementStackWithArray<E> {
6
7      private final int defaultSize = 1;
8      Object[] stack;
9      private int peekIndex = -1;
10
11     public ImplementStackWithArray() {
12
13         stack = new Object[defaultSize];
14     }
15
16     public void push(E val) {
17
18         peekIndex++;
19         if (peekIndex >= stack.length) {
20             enlargeSize();
21         }
22         stack[peekIndex] = val;
23     }
24
25     public E pop() {
26
27         if (peekIndex == -1) {
28             return null;
29         } else {
30             return (E) stack[peekIndex--];
31         }
32     }
33
34     public E peek() {
35         if ( peekIndex == -1 ) {
36             return null;
37         } else {
38             return (E) stack[peekIndex];
39         }
40     }
41
42     public boolean isEmpty() {
43         return peekIndex == -1;
44     }
45
46     private void enlargeSize() {
47
48         stack = Arrays.copyOf(stack, stack.length * 2);
49     }
50
51     public static void main(String[] args) {
52
53         ImplementStackWithArray<Integer> stack = new
54             ImplementStackWithArray<Integer>();
55
56         Integer result1 = stack.pop();
57         Integer result2 = stack.peek();
58
59         for (int i = 0; i < 9; i++ ) {
60             stack.push(i);
61         }
62
63         for (int i = 0; i < 10; i++) {
64             System.out.println(stack.pop());
65         }
66     }

```