```java
package StackAndQueue;

import java.util.Arrays;

class ImplementStackWithArray<E> {

    private final int defaultSize = 1;
    Object[] stack;
    private int peekIndex = -1;

    public ImplementStackWithArray() {

        stack = new Object[defaultSize];
    }

    public void push(E val) {

        peekIndex++;
        if (peekIndex >= stack.length) {
            enlargeSize();
        }
        stack[peekIndex] = val;
    }

    public E pop() {

        if (peekIndex == -1) {
            return null;
        } else {
            return (E)stack[peekIndex--];
        }
    }

    public E peek() {
        if ( peekIndex == -1 ) {
            return null;
        } else {
            return (E)stack[peekIndex];
        }
    }

    public boolean isEmpty() {
        return peekIndex == -1;
    }

    private void enlargeSize() {

        stack = Arrays.copyOf(stack, stack.length * 2);
    }

    public static void main(String[] args) {

        ImplementStackWithArray<Integer> stack = new
        ImplementStackWithArray<Integer>();

        Integer result1 = stack.pop();
        Integer result2 = stack.peek();

        for (int i = 0; i < 9; i++ ) {
            stack.push(i);
        }

        for (int i = 0; i < 10; i++) {
            System.out.println(stack.pop());
        }
    }
}
```