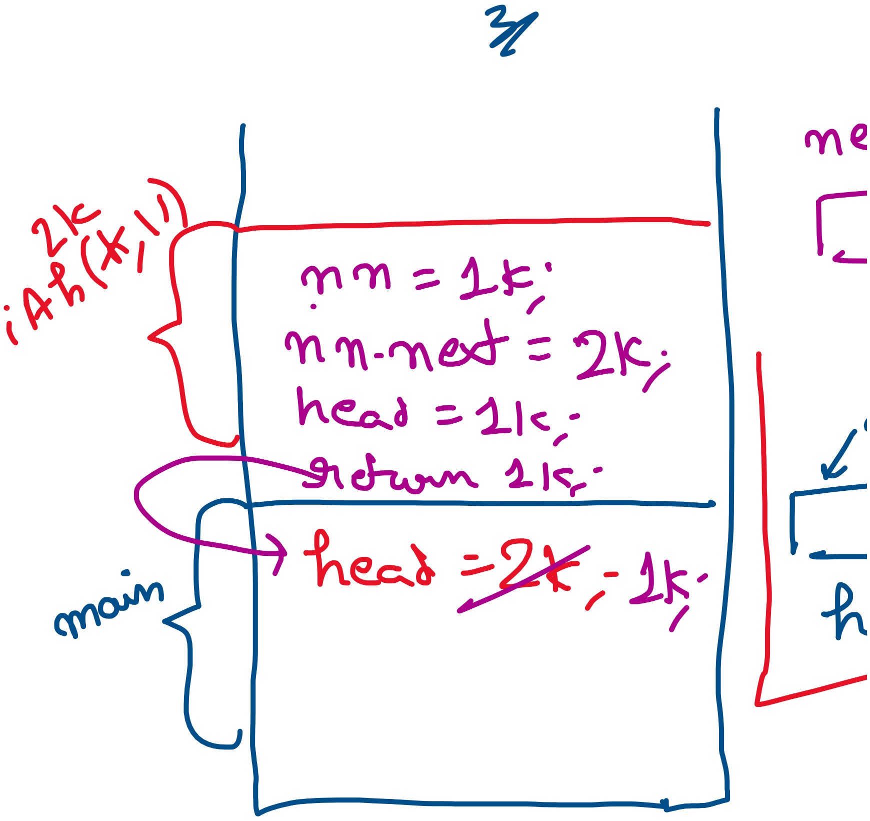
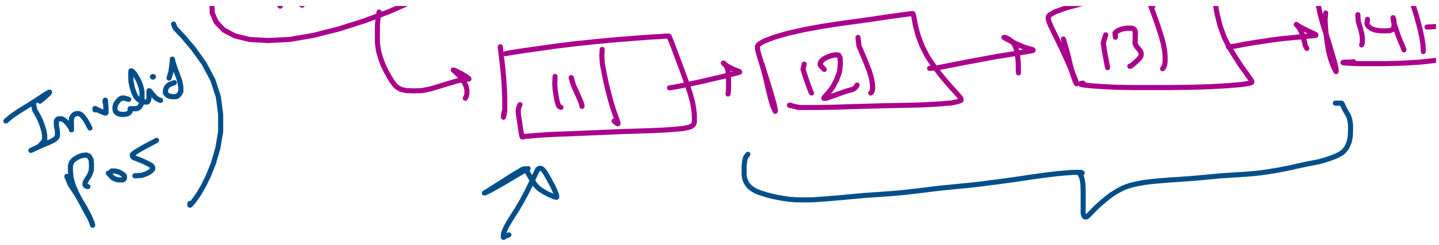
06 February 2024 15:28

```
import java.util.*;

public class Main {

  static class node{
    int data;
    node next;

    //constructor

    node(int val){
      this.data = val;
      this.next = null;
    }
  }

  public static node insertAtHead(node head,int val){
    node newnode = new node(val);

    newnode.next = head;

    head = newnode;

    return head;
  }


  public static node insertAtTail(node tail,int val){
    node newnode = new node(val);

    tail.next = newnode;

    tail = newnode;

    return tail;
  }


  public static void traverse(node head){
    node temp = head;

    while(temp != null){
      System.out.print(temp.data + " ");
      temp = temp.next;
    }
  }
  public static void main(String[] args) {
    node head = new node(12);

    node tail = head;

    head  =  insertAtHead(head,11);
    head  =  insertAtHead(head,10);
    head  =  insertAtHead(head,9);
    head  =  insertAtHead(head,8);

    traverse(head);
    System.out.println();

    tail = insertAtTail(tail,13);
    tail = insertAtTail(tail,14);
    tail = insertAtTail(tail,15);
    tail = insertAtTail(tail,16);
    tail = insertAtTail(tail,17);

    traverse(head);
    System.out.println();

    insertAtAnyNode(head,3,1000);
  }
}
```

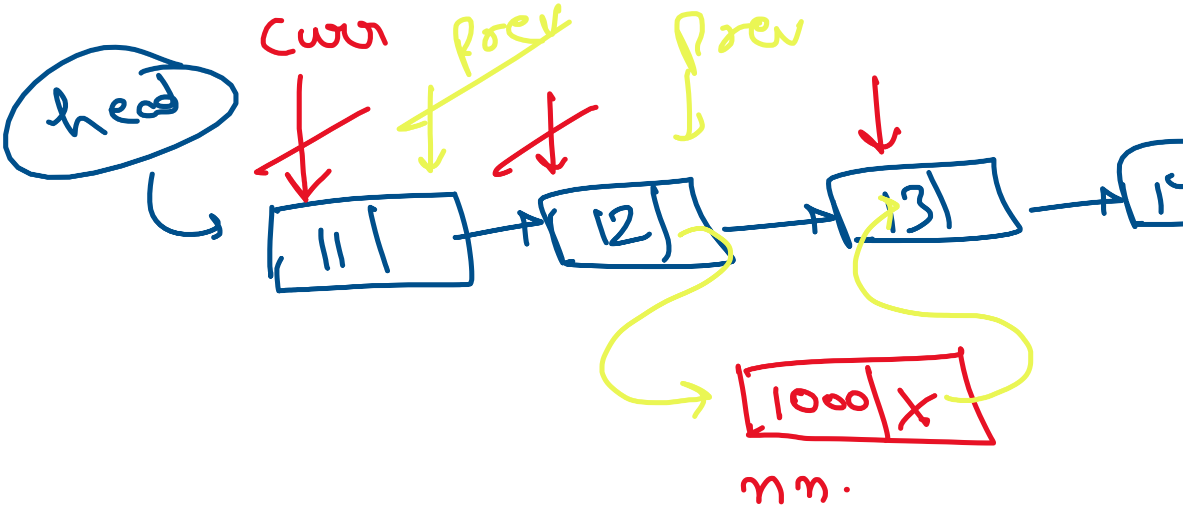
2k, 11

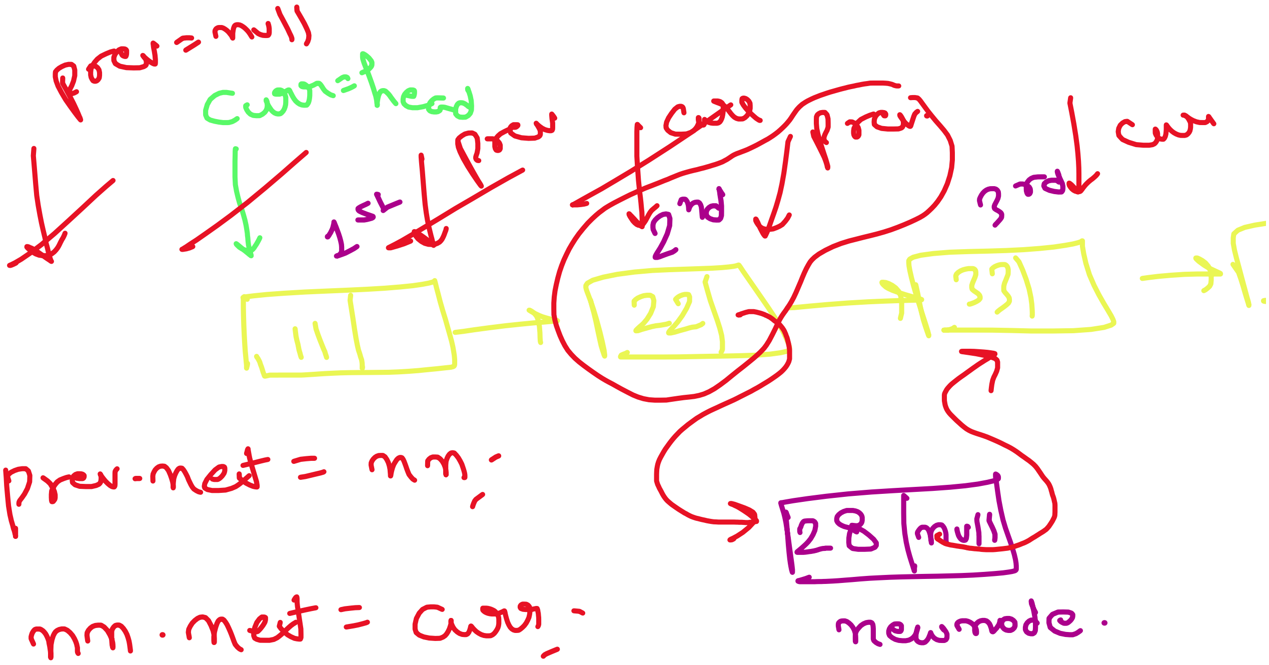




(Insert At head)

CASE-I



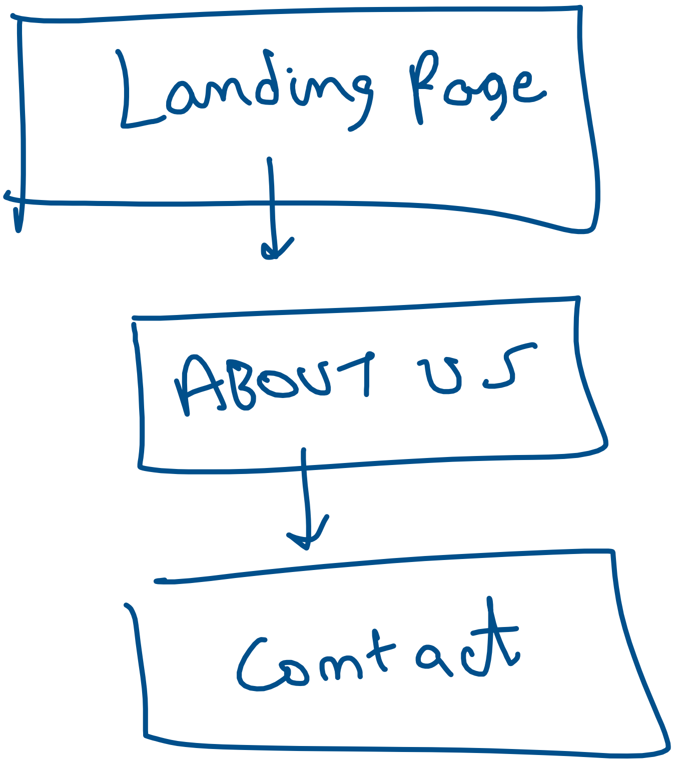


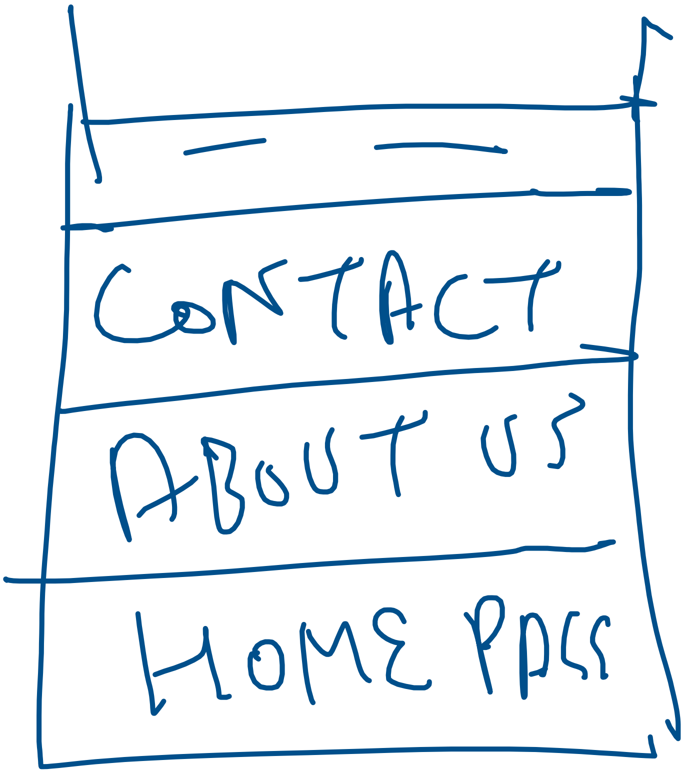
prev.next = nn;

nn.next = curr;

STACK

- Linear D.S
- LIFO order





D.S

- LI

DS

Array →

Linear

{ Kisi

pr (

D.S

→ Insert

→ Delete

→ Travers

#

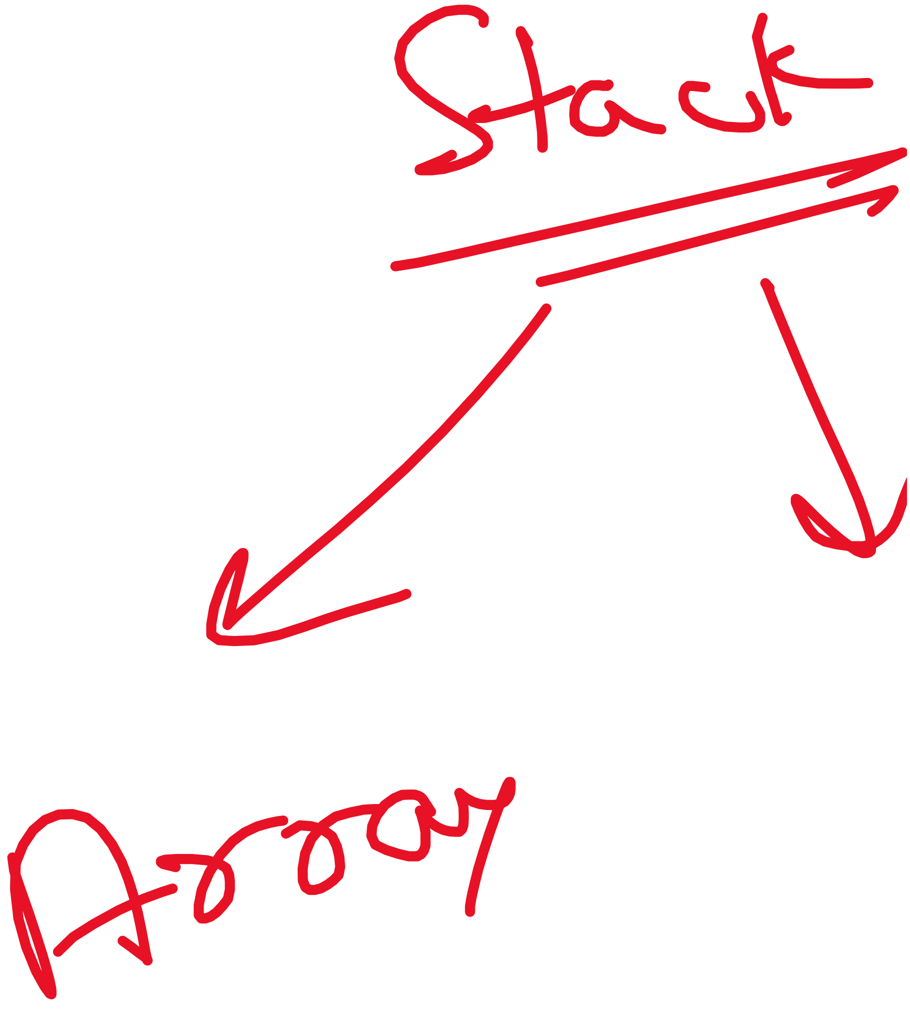
Stack <

STAC

STAC

Lin

push/pop!



Stack<

D-T

→ Push

→ Pop

→ peek

isEmpty()

Stack<Integer

st.push(1);

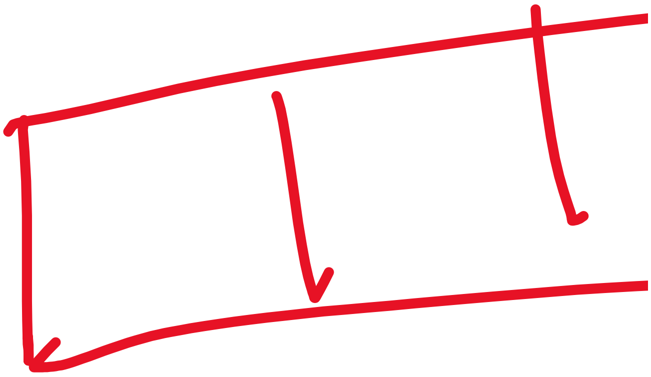
st.push(2);

st.push(3);

st.push(4);

st.pop().

arr



Class St