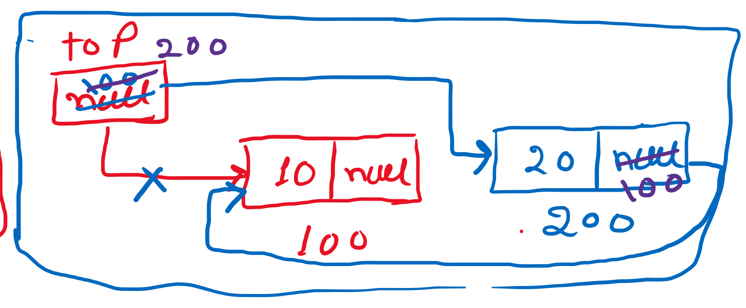
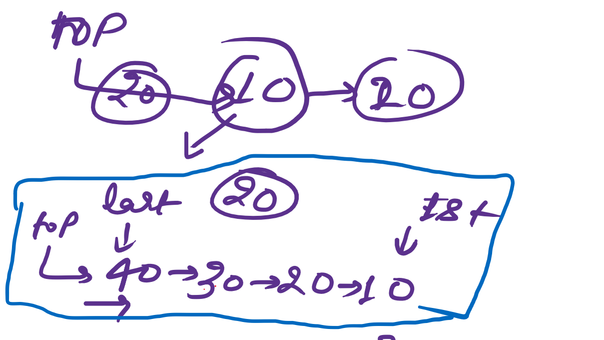
① Push → operation
② POP
③ Peek or (top)

Stack with linked




```
public class StackList{
  Node top;
  class Node{
     int data;
     Node next;
  Node(int data){
   this.data=data;
  }
}
```

```
public void push(int data){
  Node n1 = new Node(data);
   if(top==null){
     top=n1;
   }else{
     n1.next=top;
     top=n1;
   }
}
```



20
Print
40 30 20 10 30
40
LIFO

top = 100 200
top=n1

20 → 10

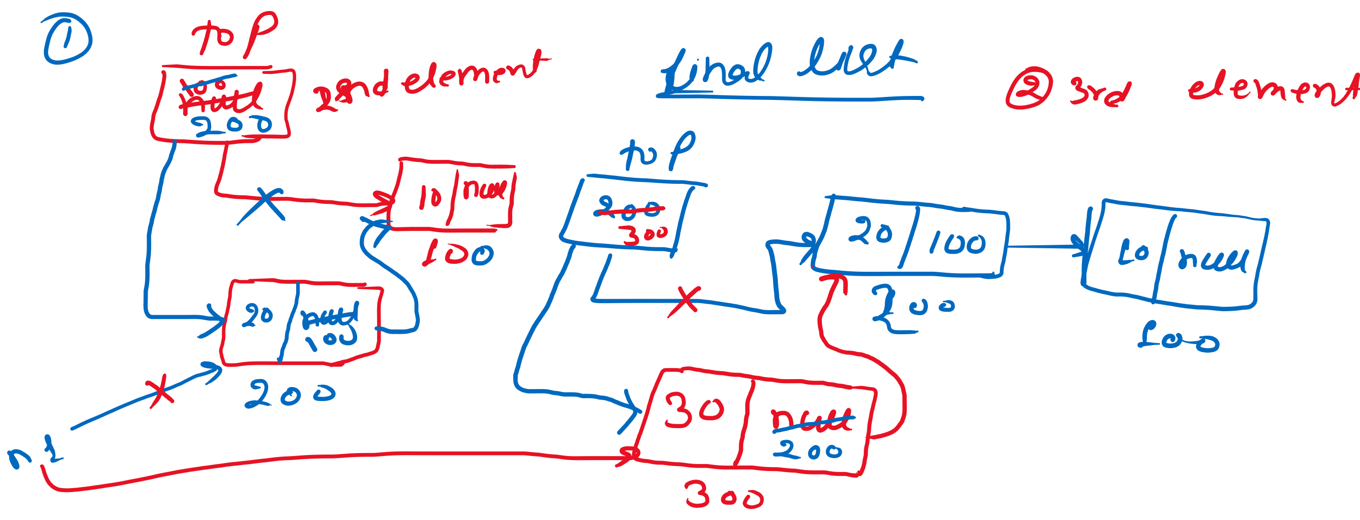
20

10 100
20 200

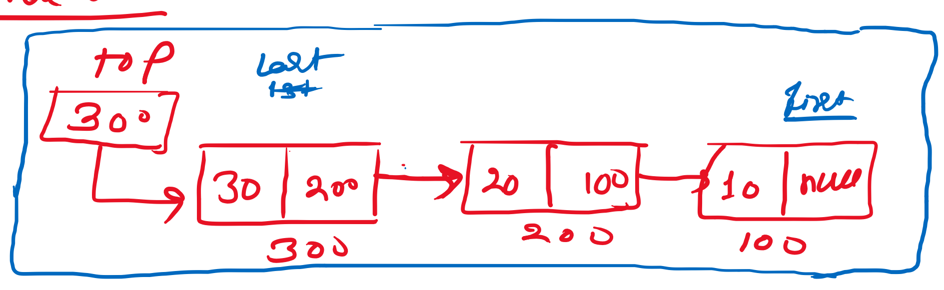
① 2nd element

final list

② 3rd element



final list



top=300 200

top = top.next

top = 300.next
200

Node temp= top

top=top.next;

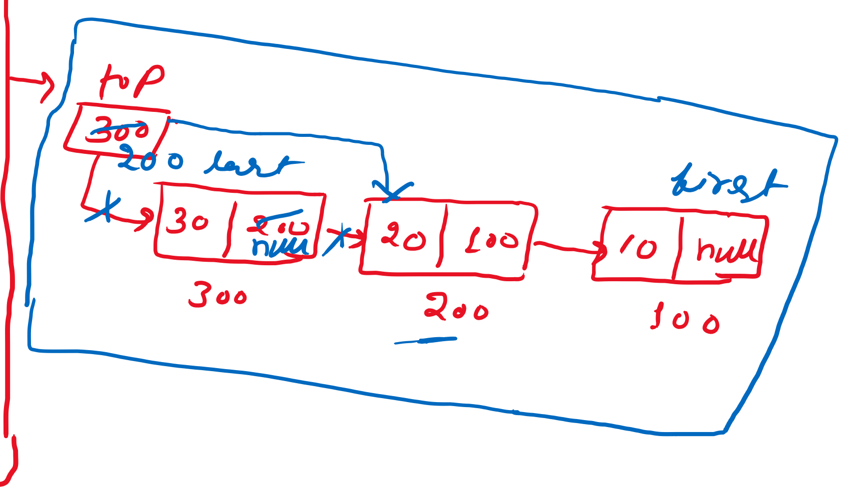
temp.next=null;

temp=null

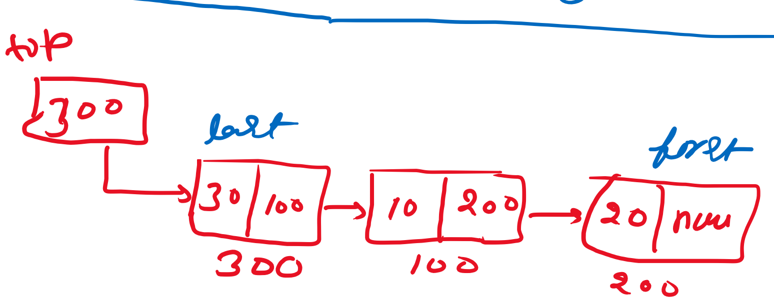
temp=100

Pop operation in linked list

```
public void pop(){
    if(toop==null){
     Sop("stack is underflow---");
    }else{
      Node temp= top;
      top = top.next;
      temp.next= null;
      temp =null;
    }
}
```



(Peek) (Top) operation in stack using linked list ⇒ it will return top most element




```
public void peek(){
   if(top ==null){
   Sop("Stack is underflow..");
  }else{
    int value = top.data;
    Sop(value);
   }
}
```

03-08-2023 (topic covered)

① Stack using linked list (POP, Push, peek)
② Program for stack using linked list.

O(n) O(1)
Push →
POP →

04-08-2023 topic

① Balance of symbol using stack
② Performance of the stack