Link list

**Insertion at first**

import java.util.\*;

public class addatfirst{

static class node{

int data;

node next;

node(int data){

this.data=data;

this.next=null;

}}

node head=null;

public void create(){

int data;

System.out.println("enter the element you want enter");

Scanner sc=new Scanner(System.in);

int n;

do{

data=sc.nextInt();

node n2=new node(data);

if(head==null){

head=n2;

}

else{

n2.next=head;

head=n2;

}

System.out.println("do you want to enter more enter 1");

n=sc.nextInt();

}while(n==1);

}

public void display(){

node temp=head;

if(head==null){

System.out.println("no linklist");

}

else{

while(temp!=null){

System.out.println(temp.data);

temp=temp.next;

}

}

}

public static void main (String [] args){

addatfirst a=new addatfirst();

a.create();

a.display();

}

}

**Insert at end**

**import java.util.\*;**

**public class addatend{**

**static class node{**

**int data;**

**node next;**

**node(int data){**

**this.data=data;**

**this.next=null;**

**}**

**}**

**node head;**

**public void create(){**

**int data;**

**node temp=head; // to find the address of the last node and link to new node**

**System.out.println("enter the element you want enter");**

**Scanner sc=new Scanner(System.in);**

**int n;**

**do{**

**data=sc.nextInt();**

**node n2=new node(data);**

**if(head==null){**

**head=n2;**

**temp=head;**

**}**

**else{**

**temp.next=n2;**

**temp=temp.next;**

**}**

**System.out.println("do you want to enter more enter 1");**

**n=sc.nextInt();**

**}while(n==1);**

**}**

**public void display(){**

**node temp=head;**

**System.out.println("display");**

**if(head==null){**

**System.out.println("no linklist");**

**}**

**else{**

**while(temp!=null){**

**System.out.println(temp.data);**

**temp=temp.next;**

**}**

**}**

**}**

**public static void main (String [] args){**

**addatend a=new addatend();**

**a.create();**

**a.display();**

**}**

**}**

**Insert at any postion**

**import java.util.\*;**

**public class addatpos {**

**static class node{**

**int data;**

**node next;**

**node(int data){**

**this.data=data;**

**this.next=null;**

**}**

**}**

**node head;**

**public void create(){**

**int data;**

**System.out.println("enter the element you want enter");**

**Scanner sc=new Scanner(System.in);**

**int n;**

**do{**

**data=sc.nextInt();**

**node n2=new node(data);**

**if(head==null){**

**head=n2;**

**}**

**else{**

**n2.next=head;**

**head=n2;**

**}**

**System.out.println("\ndo you want to enter more enter 1");**

**n=sc.nextInt();**

**}while(n==1);**

**}**

**public void display(){**

**node temp=head;**

**if(head==null){**

**System.out.println("no linklist");**

**}**

**else{**

**while(temp!=null){**

**System.out.print(temp.data+"-\t");**

**temp=temp.next;**

**}**

**}**

**}**

**public void insertpos(int x,int pos){**

**node n2=new node(x);**

**int l=0;**

**node temp=head;**

**node prev=head;**

**if(pos==0){**

**n2.next=head;**

**head=n2;**

**}**

**while(temp!=null){**

**if(pos==l){**

**prev.next=n2;**

**n2.next=temp;**

**return;**

**}**

**prev=temp;**

**temp=temp.next;**

**l++;**

**}**

**System.out.println("input is wrong");**

**return ;**

**}**

**public static void main (String [] args){**

**addatpos a=new addatpos();**

**Scanner sc=new Scanner(System.in);**

**a.create();**

**a.display();**

**System.out.println("enter the element and postion");**

**int x=sc.nextInt();**

**int pos=sc.nextInt();**

**a.insertpos(x,pos);**

**a.display();**

**}**

**}**

**Detete at first**

**import java.util.\*;**

**public class deleteatfirst {**

**static class node{**

**int data;**

**node next;**

**node head;**

**node(int data){**

**this.data=data;**

**this.next=null;**

**}**

**public void create(){**

**int data;**

**node temp=head;**

**System.out.println("enter the element you want enter");**

**Scanner sc=new Scanner(System.in);**

**int n;**

**do{**

**data=sc.nextInt();**

**node n2=new node(data);**

**if(head==null){**

**head=n2;**

**temp=head;**

**}**

**else{**

**temp.next=n2;**

**temp=temp.next;**

**}**

**System.out.println("do you want to enter more enter 1");**

**n=sc.nextInt();**

**}while(n==1);**

**}**

**public void delete(){**

**int n;**

**node temp=head;**

**Scanner sc=new Scanner(System.in);**

**if(head==null){**

**System.out.println("no list ");**

**return ;**

**}**

**System.out.println("\ndo you want to delete the first node. enter 1 to delete");**

**n=sc.nextInt();**

**while(n==1){**

**System.out.println("do you want to delete the first node. enter 1 to delete");**

**n=sc.nextInt();**

**head=head.next;**

**temp.next=null;**

**System.out.println("deleted element is "+temp.data);**

**}**

**}**

**public void display(){**

**node temp=head;**

**System.out.println("display");**

**if(head==null){**

**System.out.println("no linklist");**

**}**

**else{**

**while(temp!=null){**

**System.out.print(temp.data+"-\t");**

**temp=temp.next;**

**}**

**}**

**}**

**}**

**public static void main (String [] args){**

**node a=new node(10);**

**a.create();**

**a.display();**

**a.delete();**

**a.display();**

**}**

**}**

**Delete from end**

**import java.util.\*;**

**public class deleteatend {**

**static class node{**

**int data;**

**node next;**

**node(int data){**

**this.data=data;**

**this.next=null;**

**}**

**}**

**node head;**

**public void create(){**

**int data;**

**node temp=head;**

**System.out.println("enter the element you want enter");**

**Scanner sc=new Scanner(System.in);**

**int n;**

**do{**

**data=sc.nextInt();**

**node n2=new node(data);**

**if(head==null){**

**head=n2;**

**temp=head;**

**}**

**else{**

**temp.next=n2;**

**temp=temp.next;**

**}**

**System.out.println("do you want to enter more enter 1");**

**n=sc.nextInt();**

**}while(n==1);**

**}**

**public void delete(){**

**int n;**

**node temp=head;**

**node prev=head;**

**Scanner sc=new Scanner(System.in);**

**if(head==null){**

**System.out.println("no list ");**

**return ;**

**}**

**System.out.println("\ndo you want to delete the end node. enter 1 to delete");**

**n=sc.nextInt();**

**while(n==1){**

**System.out.println("do you want to delete the end node. enter 1 to delete");**

**n=sc.nextInt();**

**while(temp.next!=null){**

**prev=temp;**

**temp=temp.next;**

**}**

**System.out.println("deleted element is "+temp);**

**prev.next=null;**

**}**

**}**

**public void display(){**

**node temp=head;**

**System.out.println("display");**

**if(head==null){**

**System.out.println("no linklist");**

**}**

**else{**

**while(temp!=null){**

**System.out.print(temp.data+"-\t");**

**temp=temp.next;**

**}**

**}**

**}**

**public static void main (String [] args){**

**deleteatend a=new deleteatend();**

**a.create();**

**a.display();**

**a.delete();**

**a.display();**

**}**

**}**

**Delete from any pos.**

**import java.util.\*;**

**public class delete {**

**static class node{**

**int data;**

**node next;**

**node head;**

**node(int data){**

**this.data=data;**

**this.next=null;**

**}}**

**node head;**

**public void create(){**

**int data;**

**node temp=head;**

**System.out.println("enter the element you want enter");**

**Scanner sc=new Scanner(System.in);**

**int n;**

**do{**

**data=sc.nextInt();**

**node n2=new node(data);**

**if(head==null){**

**head=n2;**

**temp=head;**

**}**

**else{**

**temp.next=n2;**

**temp=temp.next;**

**}**

**System.out.println("do you want to enter more enter 1");**

**n=sc.nextInt();**

**}while(n==1);**

**}**

**public void deletenode(){**

**int n,m;**

**node temp=head;**

**node prev=head;**

**Scanner sc=new Scanner(System.in);**

**do{**

**System.out.println("enter where you want to delete 1.start\n 2.end\n 3.middle");**

**m=sc.nextInt();**

**if(head==null){**

**System.out.println("no list ");**

**return ;**

**}**

**switch (m){**

**case 1: System.out.println("deleted from start");**

**head=head.next;**

**temp=temp.next;**

**break;**

**case 2: System.out.println("deleted from last");**

**while(temp.next!=null){**

**prev=temp;**

**temp=temp.next;**

**}**

**prev.next=null;**

**break;**

**case 3:System.out.println("deleted from position");**

**System.out.println("enter the position");**

**int x=sc.nextInt();**

**for(int i=0;i<=x-2;i++){**

**prev=temp;**

**temp=temp.next;**

**}**

**prev.next=prev.next.next;**

**break;**

**default:System.out.println("wrong input ");**

**break;**

**}**

**System.out.println("you want to delete more enter 1");**

**n=sc.nextInt();**

**}while(n==1);**

**}**

**public void display(){**

**node temp=head;**

**System.out.println("display");**

**if(head==null){**

**System.out.println("no linklist");**

**}**

**else{**

**while(temp!=null){**

**System.out.print(temp.data+"-\t");**

**temp=temp.next;**

**}**

**}**

**}**

**public static void main (String [] args){**

**delete a=new delete();**

**a.create();**

**a.display();**

**a.deletenode();**

**a.display();**

**}**

**}**

**Reverse**

**import java.util.\*;**

**public class reverselinklist{**

**static class node{**

**int data;**

**node next;**

**node(int data){**

**this.data=data;**

**this.next=null;**

**}}**

**node head=null;**

**public void create(){**

**int data;**

**node temp=null;**

**System.out.println("enter the element you want enter");**

**Scanner sc=new Scanner(System.in);**

**int n;**

**do{**

**data=sc.nextInt();**

**node n2=new node(data);**

**if(head==null){**

**head=n2;**

**temp=head;**

**}**

**else{**

**temp.next=n2;**

**temp=temp.next;**

**}**

**System.out.println("do you want to enter more enter 1");**

**n=sc.nextInt();**

**}while(n==1);**

**}**

**public void display(){**

**node temp=head;**

**if(head==null){**

**System.out.println("no linklist");**

**}**

**else{**

**while(temp!=null){**

**System.out.println("array"+temp.data);**

**temp=temp.next;**

**}**

**}**

**}**

**public void reverse(){**

**node prev=null;**

**node temp=head;**

**node ptr=head;**

**while(head.next!=null){**

**head=head.next;**

**}**

**while(ptr!=null){**

**ptr=temp.next;**

**temp.next=prev;**

**prev=temp;**

**temp=ptr;**

**}**

**}**

**public static void main (String [] args){**

**reverselinklist a=new reverselinklist();**

**a.create();**

**a.reverse();**

**a.display();**

**}**

**}**

**Double link list**

**Insertion at end**

**import java.util.\*;**

**public class insertatendlist{**

**static class node{**

**int data;**

**node next;**

**node prev;**

**node(int data){**

**this.data=data;**

**this.next=null;**

**this.prev=null;**

**}}**

**node head=null;**

**node tail=null;**

**public void create(){**

**int data;**

**System.out.println("enter the element you want enter");**

**Scanner sc=new Scanner(System.in);**

**int n;**

**do{**

**data=sc.nextInt();**

**node n2=new node(data);**

**if(head==null){**

**head=n2;**

**tail=n2;**

**}**

**else{**

**tail.next=n2;**

**n2.prev=tail;**

**tail=n2;**

**}**

**System.out.println("do you want to enter more enter 1");**

**n=sc.nextInt();**

**}while(n==1);**

**}**

**public void display(){**

**node temp=head;**

**System.out.println("double \tlinklist");**

**if(head==null){**

**System.out.println("no linklist");**

**}**

**else{**

**while(temp!=null){**

**System.out.println(temp.data+"\n");**

**temp=temp.next;**

**}**

**}**

**}**

**public static void main (String [] args){**

**insertatendlist a=new insertatendlist();**

**a.create();**

**a.display();**

**}**

**}**

**Insert at first**

**import java.util.\*;**

**public class inseratstart{**

**static class node{**

**int data;**

**node next;**

**node prev;**

**node(int data){**

**this.data=data;**

**this.next=null;**

**this.prev=null;**

**}}**

**node head=null;**

**node tail=null;**

**public void create(){**

**int data;**

**System.out.println("enter the element you want enter");**

**Scanner sc=new Scanner(System.in);**

**int n;**

**do{**

**data=sc.nextInt();**

**node n2=new node(data);**

**if(head==null){**

**head=n2;**

**tail=n2;**

**}**

**else{**

**head.prev=n2;**

**n2.next=head;**

**head=n2;**

**}**

**System.out.println("do you want to enter more enter 1");**

**n=sc.nextInt();**

**}while(n==1);**

**}**

**public void display(){**

**node temp=head;**

**if(head==null){**

**System.out.println("no linklist");**

**}**

**else{**

**while(temp!=null){**

**System.out.println(temp.data);**

**temp=temp.next;**

**}**

**}**

**}**

**public static void main (String [] args){**

**inseratstart a=new inseratstart();**

**a.create();**

**a.display();**

**}**

**}**

**Insert at any postion**

**import java.util.Scanner;**

**public class insertatpos{**

**static class node{**

**node next;**

**node prev;**

**int data;**

**node(int data){**

**this.data=data;**

**this.next=null;**

**this.prev=null;**

**}**

**}**

**node head=null;**

**node tail=null;**

**public void create(){**

**int data,n;**

**Scanner sc=new Scanner(System.in);**

**do{**

**System.out.println("enter the element");**

**data=sc.nextInt();**

**node n2=new node(data);**

**if(head==null){**

**head=n2;**

**tail=n2;**

**}**

**else{**

**tail.next=n2;**

**n2.prev=tail;**

**tail=n2;**

**}**

**System.out.println("do you want enter more enter 1");**

**n=sc.nextInt();**

**}while(n==1);**

**}**

**public void display(){**

**node temp=head;**

**if(head==null){**

**System.out.println("no link list");**

**}**

**else{**

**while(temp!=null){**

**System.out.print(temp.data+"->");**

**temp=temp.next;**

**}**

**}**

**}**

**public void pos(){**

**int data,n,m;**

**Scanner sc = new Scanner (System . in) ;**

**node temp=head;**

**node pre=head;**

**do{**

**System.out.println("enter the postion");**

**n=sc.nextInt();**

**System.out.println("enter the element");**

**data=sc.nextInt();**

**node n2=new node(data);**

**if(n==1){**

**n2.next=head;**

**head.prev=n2;**

**head=n2;**

**}**

**else{**

**for(int i=0;i<n-2;i++){**

**temp=temp.next;**

**pre=pre.next;**

**}**

**temp=temp.next;**

**pre.next=n2;**

**n2.prev=pre;**

**n2.next=temp;**

**temp.prev=n2;**

**}**

**System.out.println("wnat to enter more press 1");**

**m=sc.nextInt();**

**}while(m==1);**

**}**

**public static void main(String [] args){**

**insertatpos a = new insertatpos();**

**a.create();**

**a.display();**

**a.pos();**

**a.display();**

**}**

**}**

**Stack**

**import java.util.\*;**

**public class stack1{**

**Scanner sc=new Scanner(System.in);**

**int n=5;**

**int c;**

**int a[]=new int[n];**

**int top;**

**public void push(){**

**do{**

**if(top==(n-1)){**

**System.out.println("overflow");**

**}**

**else{**

**System.out.println("enter the data");**

**int e=sc.nextInt();**

**top=top+1;**

**a[top]=e;**

**}**

**System.out.println("enter 1 to enter more");**

**c=sc.nextInt();**

**}while(c==1);**

**}**

**public void pop(){**

**if(top==-1){**

**System.out.println("underflow");**

**}**

**else{**

**do{ System.out.println("poping");**

**top--;**

**System.out.println("enter 1 more delete");**

**n=sc.nextInt();**

**}while(n==1);**

**}**

**}**

**public void display(){**

**if(top==-1){**

**System.out.println("stack is empty");**

**}**

**else{**

**for (int i =top ;i>0;i--){**

**System.out.println(a[i]);**

**}**

**}**

**}**

**public static void main(String[] args){**

**stack1 s=new stack1();**

**s.push();**

**s.pop();**

**s.push();**

**s.display();**

**}**

**}**

**Queue**

**import java.util.\*;**

**public class c1{**

**int f=-1,r=-1,n=10;**

**int q[]=new int[n];**

**void enqueue(){**

**Scanner sc=new Scanner(System.in);**

**if(r==(n-1)){**

**System.out.println("overflow");**

**}**

**else{**

**System.out.println("enter the ele");**

**int i=sc.nextInt();**

**if(r==-1 & f==-1){**

**f=0;**

**r=0;**

**q[r]=i;**

**}**

**else{**

**r=r+1;**

**q[r]=i;**

**}**

**}**

**}**

**void dequeu(){**

**if(r==-1 &f==-1){**

**System.out.println("underflow");**

**}**

**else{**

**f=f+1;**

**}**

**}**

**void display(){**

**System.out.println("ele are");**

**for(int i=f;i<=r;i++){**

**System.out.println(q[i]);**

**}**

**}**

**public static void main(String [] args){**

**Scanner sc= new Scanner(System.in);**

**c1 c=new c1();**

**int l,d;**

**do{**

**System.out.println("enter 1 enqueue");**

**System.out.println("enter 2 to dequeue");**

**System.out.println("enter your choice");**

**System.out.println("enter 3 display");**

**d=sc.nextInt();**

**switch(d){**

**case 1: c.enqueue();**

**break;**

**case 2:c.dequeu();**

**break;**

**case 3:c.display();**

**break;**

**default:System.out.println("wrong input");**

**break;**

**}**

**System.out.println("enter any key exit");**

**l=sc.nextInt();**

**}while(l==1);**

**System.out.println("completed");**

**}**

**}**

**Circular queue**

**import java.util.\*;**

**public class circularq{**

**int f=-1,r=-1,n=5;**

**int q[]=new int[n];**

**void enqueue(){**

**Scanner sc=new Scanner(System.in);**

**if(f==(r+1)%n){**

**System.out.println("overflow");**

**}**

**else{**

**System.out.println("enter the ele");**

**int i=sc.nextInt();**

**if(r==-1 & f==-1){**

**f=0;**

**r=0;**

**q[r]=i;**

**}**

**else{**

**r=(r+1)%n;**

**q[r]=i;**

**}**

**}**

**}**

**void dequeu(){**

**if(r==-1 &f==-1){**

**System.out.println("underflow");**

**}**

**else{**

**f=f+1;**

**}**

**}**

**void display(){**

**System.out.println("ele are");**

**for(int i=f;i<=r;i++){**

**System.out.println(q[i]);**

**}**

**}**

**public static void main(String [] args){**

**Scanner sc= new Scanner(System.in);**

**circularq c=new circularq();**

**int l,d;**

**do{**

**System.out.println("enter 1 enqueue");**

**System.out.println("enter 2 to dequeue");**

**System.out.println("enter your choice");**

**System.out.println("enter 3 display");**

**d=sc.nextInt();**

**switch(d){**

**case 1: c.enqueue();**

**break;**

**case 2:c.dequeu();**

**break;**

**case 3:c.display();**

**break;**

**default:System.out.println("wrong input");**

**break;**

**}**

**System.out.println("enter any key exit");**

**l=sc.nextInt();**

**}while(l==1);**

**System.out.println("completed");**

**}**

**}**

**Dequeue**

**// Deque implementation in Java**

**class dequeue {**

**static final int MAX = 100;**

**int arr[];**

**int front;**

**int rear;**

**int size;**

**public dequeue(int size) {**

**arr = new int[MAX];**

**front = -1;**

**rear = 0;**

**this.size = size;**

**}**

**boolean isFull() {**

**return ((front == 0 && rear == size - 1) || front == rear + 1);**

**}**

**boolean isEmpty() {**

**return (front == -1);**

**}**

**void insertfront(int key) {**

**if (isFull()) {**

**System.out.println("Overflow");**

**return;**

**}**

**if (front == -1) {**

**front = 0;**

**rear = 0;**

**}**

**else if (front == 0)**

**front = size - 1;**

**else**

**front = front - 1;**

**arr[front] = key;**

**}**

**void insertrear(int key) {**

**if (isFull()) {**

**System.out.println(" Overflow ");**

**return;**

**}**

**if (front == -1) {**

**front = 0;**

**rear = 0;**

**}**

**else if (rear == size - 1)**

**rear = 0;**

**else**

**rear = rear + 1;**

**arr[rear] = key;**

**}**

**void deletefront() {**

**if (isEmpty()) {**

**System.out.println("Queue Underflow\n");**

**return;**

**}**

**// Deque has only one element**

**if (front == rear) {**

**front = -1;**

**rear = -1;**

**} else if (front == size - 1)**

**front = 0;**

**else**

**front = front + 1;**

**}**

**void deleterear() {**

**if (isEmpty()) {**

**System.out.println(" Underflow");**

**return;**

**}**

**if (front == rear) {**

**front = -1;**

**rear = -1;**

**} else if (rear == 0)**

**rear = size - 1;**

**else**

**rear = rear - 1;**

**}**

**int getFront() {**

**if (isEmpty()) {**

**System.out.println(" Underflow");**

**return -1;**

**}**

**return arr[front];**

**}**

**int getRear() {**

**if (isEmpty() || rear < 0) {**

**System.out.println(" Underflow\n");**

**return -1;**

**}**

**return arr[rear];**

**}**

**public static void main(String[] args) {**

**dequeue dq = new dequeue(4);**

**System.out.println("Insert element at rear end : 12 ");**

**dq.insertrear(12);**

**System.out.println("insert element at rear end : 14 ");**

**dq.insertrear(14);**

**System.out.println("get rear element : " + dq.getRear());**

**dq.deleterear();**

**System.out.println("After delete rear element new rear become : " + dq.getRear());**

**System.out.println("inserting element at front end");**

**dq.insertfront(13);**

**System.out.println("get front element: " + dq.getFront());**

**dq.deletefront();**

**System.out.println("After delete front element new front become : " + +dq.getFront());**

**}**

**}**