**LAB 04:**

**Question 01:**

**public class Main{**

**Node node;**

**Node head;**

**Node ptr;**

**class Node{**

**int data;**

**Node next;**

**Node(int data){**

**this.data = data;**

**}**

**}**

**public void insert(int x){**

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

**node.data=x;**

**node.next=null;**

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

**head=node;**

**}**

**else {**

**Node n = head;**

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

**n = n.next;**

**}**

**n.next = node;**

**}**

**}**

**public void remove(int x){**

**Node node=head;**

**while (node.data!=x){**

**ptr=node;**

**node=node.next;**

**}**

**ptr.next=node.next;**

**node=null;**

**}**

**public void insertNodeAfter(int New,int counter){**

**Node node=head;**

**while (node.data!=counter){**

**ptr=node;**

**node=node.next;**

**}**

**Node node1=new Node(New);**

**node1.data=New;**

**node1.next=node.next;**

**node.next=node1;**

**}**

**public void swap(int a,int b) {**

**Node node =head;**

**while(node.data != b) {**

**ptr = node;**

**node = node.next;**

**}**

**int val= head.data;**

**head.data= ptr.data;**

**ptr.data=val;**

**}**

**public void display(){**

**Node node=head;**

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

**System.out.println("List doesn't exists");**

**}**

**else {**

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

**System.out.print(node.data+ " ");**

**node=node.next;**

**}**

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

**}**

**}**

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

**Main t=new Main();**

**int d=0;**

**for (int i=1;i<=10;i++){**

**d=i\*2;**

**t.insert(d);**

**}**

**t.display();**

**t.remove(10);**

**t.display();**

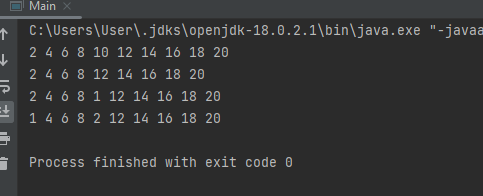
**t.insertNodeAfter(1,8);**

**t.display();**

**t.swap(2,1);**

**t.display();**

**} }**



**Question 02(a,b):**

import java.util.Scanner;

public class Main {

node head;

public static void main(String[] args) {

Main q= new Main();

q.insert(1);

q.insert(3);

q.insert(7);

q.insert(4);

q.insert(4);

q.insert(3);

q.insert(7);

q.insert(2);

Scanner sc = new Scanner(System.in);

System.out.println("=========Linked list========");

q.print();

System.out.println("\nEnter number that you wanna search: ");

System.out.println(q.search(sc.nextInt()));

System.out.println();

q.delete(7);

q.print();

}

class node{

int data;

node next;

}

// Putting values

void insert(int a){

node newnode = new node();

node n = head;

newnode.data = a;

newnode.next = null;

if (head==null) {

head = newnode;

}else {

while (n.next!=null){

n = n.next;

}

n.next = newnode;

}

}

// Searching

boolean search(int data){

node n = head;

while (n.next!=null){

if (n.data==data){

return true;

}

n=n.next;

}

return false;

}

// printing

void print(){

node n = head ;

while(n!=null){

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

n = n.next;

}

}

// Deleting

void delete(int data){

node n = head;

node temp=null;

while(n.data != data){

temp=n;

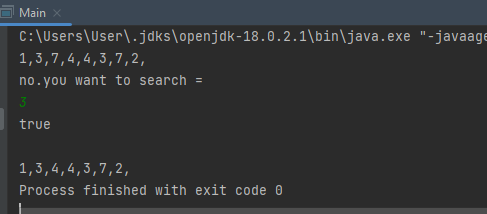
n = n.next;

}

temp.next = n.next;

}

}



**Question 02(c):**

public class Question2\_partC {

node head;

class node{

int data;

node next;

}

// for values

void insert(int a){

node newnode = new node();

node n = head;

newnode.data = a;

newnode.next = null;

if (head==null) {

head = newnode;

}else {

while (n.next!=null){

n = n.next; }

n.next = newnode;

} }

// for deletion

void delete(int data){

node n = head;

node temp;

while (n.next!=null){

temp = n;

if (n.data==data){

temp.next = n.next; }

n = n.next;

} }

void print(){

node n = head ;

while(n!=null){

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

n = n.next;

} }

public static void main(String[] args) {

Question2\_partC q = new Question2\_partC();

q.insert(1);

q.insert(3);

q.insert(7);

q.insert(4);

q.insert(3);

q.insert(7);

q.insert(2);

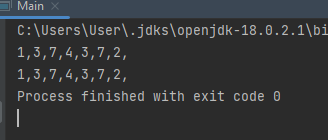
q.print();

q.delete(7);

System.out.println();

q.print ();

} }



**Question 02(d):**

public class Question\_number2D {

node head;

class node{

int data;

node next;

}

// for values

void insert(int a){

node newnode = new node();

node n = head;

newnode.data = a;

newnode.next = null;

if (head==null) {

head = newnode;

}else {

while (n.next!=null){

n = n.next;

}

n.next = newnode;

}

}

// for deletion

void delete(int data){

node n = head;

node temp=null;

while (n.next!=null){

// temp = n;

if (n.data==data) {

// temp=n;

temp = n; }

n = n.next;

}

if (temp != null && temp.next != null) {

temp.data = temp.next.data;

n = temp.next;

temp.next = temp.next.next;

} }

void print(){

node n = head ;

while(n!=null){

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

n = n.next;

} }

public static void main(String[] args) {

Question2\_partC q = new Question2\_partC();

q.insert(1);

q.insert(3);

q.insert(7);

q.insert(4);

q.insert(3);

q.insert(7);

q.insert(2);

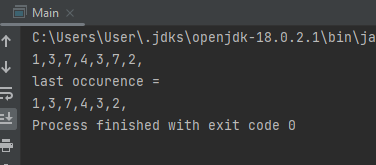
q.print();

q.delete(7);

System.out.println();

q.print ();

} }



**Question 03:**

import java.util.\*;

class List{

int d;

List n;

}

class SList{

List a;

List b;

public void insert(int x,int y){

for(int k=0;k<(2\*x);k++){

if(k%2==0){

List nno=new List();

nno.d=k;

if(a==null){

a=nno; }

else{

List t=a;

while(t.n!=null){

t=t.n; }

t.n=nno;

}

}

}

for(int k=0;k<(2\*y);k++){

if(k%2==1){

List nno=new List();

nno.d=k;

if(b==null){

b=nno; }

else{

List t=b;

while(t.n!=null){

t=t.n; }

t.n=nno;

}

}

}

}

public void displayseparate(){

List p=a;

System.out.print("Separate List A = ");

while(p!=null){

System.out.print(p.d);

System.out.print(">");

p=p.n; }

System.out.println();

List q=b;

System.out.print("Separate List B = ");

while(q!=null){

System.out.print(q.d);

System.out.print(">");

q=q.n; }

System.out.println();

}

public void display(){

List c=a;

List t=b;

System.out.print("Merged List = ");

while(c!=null||t!=null){

if(c!=null){

System.out.print(c.d);

System.out.print(">");

c=c.n; }

if(t!=null){

System.out.print(t.d);

System.out.print(">");

t=t.n;

} }

System.out.println();

}

}

public class Main{

public static void main(String[] args) {

SList l=new SList();

Scanner t=new Scanner(System.in);

System.out.println("1st list indexes should be = ");

int l1=t.nextInt();

System.out.println("2nd list indexes should be = ");

int s=t.nextInt();

l.insert(l1,s);

l.displayseparate();

l.display();

}

}

