01. Write a Java programs to print the numbers.

import java.util.Scanner;

public class MainClass

{

public static void main(String[] args)

{

Scanner sc = new Scanner(System.in);

System.out.println("How many rows you want in this pattern?");

int rows = sc.nextInt();

System.out.println("Here is your pattern....!!!");

for (int i = 1; i <= rows; i++)

{

for (int j = 1; j <= i; j++)

{

System.out.print(j+" ");

}

System.out.println();

}

for (int i = rows-1; i >= 1; i--)

{

for (int j = 1; j <= i; j++)

{

System.out.print(j+" ");

}

System.out.println();

}

sc.close();

}

}

02. Write a java program to print a left triangle star pattern

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

public class Triangle

{

public static void printStars(int n)

{

int i, j;

for(i=0; i<n; i++)

{

for(j=2\*(n-i); j>=0; j--)

{

System.out.print(" ");

}

for(j=0; j<=i; j++)

{

System.out.print("\* ");

}

System.out.println();

}

}

public static void main(String args[])

{

int n = 5;

printStars(n);

}

}

03. Write a java program to print a number pattern

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

1 2 3 4 5 6

1 2 3 4 5 6 7

1 2 3 4 5 6

1 2 3 4 5

1 2 3 4

1 2 3

1 2

1

import java.util.Scanner;

public class MainClass

{

public static void main(String[] args)

{

Scanner sc = new Scanner(System.in);

System.out.println(&quot;How many rows you want in this pattern?&quot;);

int rows = sc.nextInt();

System.out.println(&quot;Here is your pattern....!!!&quot;);

for (int i = 1; i &lt;= rows; i++)

{

for (int j = 1; j &lt;= i; j++)

{

System.out.print(j+&quot; &quot;);

}

System.out.println();

}

for (int i = rows-1; i &gt;= 1; i--)

{

for (int j = 1; j &lt;= i; j++)

{

System.out.print(j+&quot; &quot;);

}

System.out.println();

}

sc.close();

}

}

04. Write a program in java to create and display doubly linked list

public class DoublyLinkedList {

class Node{

int data;

Node previous;

Node next;

public Node(int data) {

this.data = data;

}

}

Node head, tail = null;

public void addNode(int data) {

Node newNode = new Node(data);

if(head == null) {

head = tail = newNode;

head.previous = null;

tail.next = null;

}

else {

tail.next = newNode;

newNode.previous = tail;

tail = newNode;

tail.next = null;

}

}

public void display() {

Node current = head;

if(head == null) {

System.out.println("List is empty");

return;

}

System.out.println("Nodes of doubly linked list: ");

while(current != null) {

System.out.print(current.data + " ");

current = current.next;

}

}

public static void main(String[] args) {

DoublyLinkedList dList = new DoublyLinkedList();

dList.addNode(1);

dList.addNode(2);

dList.addNode(3);

dList.addNode(4);

dList.addNode(5);

dList.display();

}

}