1. Write a program to print the following patterns :

A

```
*
* *
* * *
* * * *
.
.
```

up to n lines.

```
public class LeftTriangle {
public static void rightTriangle(int n) {
    int i, j;
    for(i=0; i<n; i++){  //outer loop for number of rows(n)
        for(j=0; j<=i; j++) //  inner loop for columns
            System.out.print("* "); // print star
        System.out.println(); // ending line after each row
    }
}
public static void main(String args[])
{
    int n = 5;
    rightTriangle(n);
}
}
```

B

```
      *
    * *
  * * *
* * * *
.
.
```

up to n terms

```
public class RightTriangle {
 public static void rightTriangle(int n) {
```

```
        int i, j;
        for(i=0; i<n; i++){  //outer loop for number of
rows(n)
            for(j=i; j<n; j++) // inner loop for spaces
                System.out.print("  "); // printing space
            for(j=0; j<=i; j++) //  inner loop for columns
                System.out.print("* "); // print star
            System.out.println(); // ending line after each
row
        }
    }
    public static void main(String args[])
    {
        int n = 5;
        rightTriangle(n);
    }
}
```

C

```
      *
     * *
    * * *
   * * * *
  * * * * *
```

upto n terms

```
public class Pyramid {
    public static void pyramidPattern(int n)
    {
        for (int i=0; i<n; i++) { //outer loop for number
of rows(n)
            for (int j=1; j<n-i; j++) //inner loop for spaces
                System.out.print("."); //print space
```

```
            for (int j=0; j<=i; j++ ) //inner loop for number
of columns
                System.out.print("* "); //print star
            System.out.println(); //ending line after each
row
        }
    }
    public static void main(String args[]) //driver
function
    {
        int n = 5;
        pyramidPattern(n);
    }
}
```

D

```
    *
    * *
    * * *
    * * * *
    * * *
    * *
    *
```

```
import java.util.Scanner;
public class RightPascalTriangle {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of rows: ");

        int rows = sc.nextInt();
        for (int i= 0; i<= rows-1 ; i++) {
            for (int j=0; j<=i; j++) {
                System.out.print("*"+ " ");
            }
```

```
            System.out.println("");
        }
        for (int i=rows-1; i>=0; i--) {
            for(int j=0; j <= i-1;j++) {
                System.out.print("*"+ " ");
            }
            System.out.println("");
        }
    }
}
```

E

```
/*
rows = 4
1->    432*
2->    43*1*
3->    4*123*
4->    *12345*
3->    4*123*
2->    43*1*
1->    432*

 */
```

```java
public class Diamond {
    public static void main(String[] args) {
        int rows = 4;
        for (int i = 1; i <= rows; i++) {
            for (int j = rows; j > i; j--) {
                System.out.print(" ");
            }
            //Printed the initial spaces
            System.out.print("*");
            //Printed the 1st *
            for (int k = 1; k < 2 * (i - 1); k++) {
                System.out.print(" ");
            }
            //Printed the middle spaces
            if (i == 1) {
                System.out.println("");
            }
            else {
                System.out.println("*");
            }
            //Printed the 2nd *
        }
        //System.out.println();
        for (int i = rows - 1; i >= 1; i--) {
            for (int j = rows; j > i; j--) {
                System.out.print(" ");
            }
            //Printed the initial spaces
            System.out.print("*");
            //Printed the 1st *
            for (int k = 1; k < 2 * (i - 1); k++) {
                System.out.print(" ");
            }
            //Printed the middle spaces
```

```
            if (i != 1)
                System.out.println("*");
            //Printed the 2nd *
        }
    }
}
```