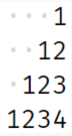
# Lecture 6 : Patterns - 2

## Code : Mirror Image Number Pattern

Send Feedback

Print the following pattern for the given N number of rows.

### Pattern for N = 4



```
···1
··12
·123
1234
```

The dots represent spaces.

### Input format :

Integer N (Total no. of rows)

### Output format :

Pattern in N lines

### Constraints

0 <= N <= 50

### Sample Input 1:

3

### Sample Output 1:

```
  1
 12
123
```

### Sample Input 2:

4

### Sample Output 2:

```
   1
  12
 123
1234
```

```
import java.util.Scanner;

public class Solution {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();

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

## Code : Inverted Number Pattern

Send Feedback

Print the following pattern for the given N number of rows.

**Pattern for N = 4**

4444

333

22

1

**Input format :**

Integer N (Total no. of rows)

**Output format :**

Pattern in N lines

**Constraints :**

0 <= N <= 50

**Sample Input 1:**

5

**Sample Output 1:**

55555

4444

333

22

1

**Sample Input 2:**

6

**Sample Output 2:**

666666

55555

4444

333

22

1

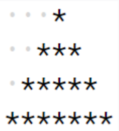
```
import java.util.Scanner;

public class Solution {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();

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

## Code : Star Pattern

Send Feedback

Print the following pattern

Pattern for N = 4



```
···*
··***
·*****
*******
```

The dots represent spaces.

### Input Format :

N (Total no. of rows)

### Output Format :

Pattern in N lines

### Constraints :

0 <= N <= 50

### Sample Input 1 :

3

### Sample Output 1 :

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

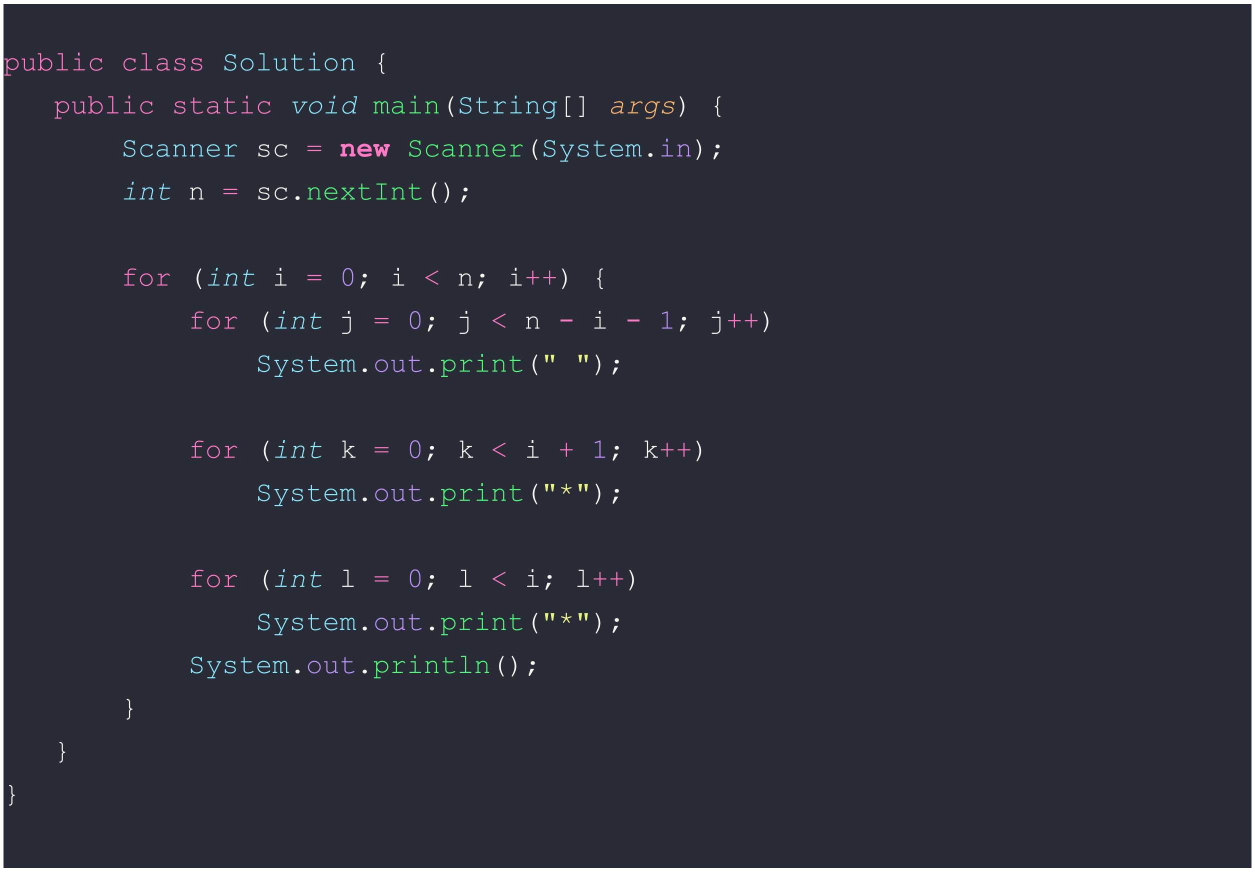
### Sample Input 2 :

4

### Sample Output 2 :

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

```
import java.util.Scanner;
```



```java
public class Solution {
   public static void main(String[] args)  {
       Scanner sc = new Scanner(System.in);
       int n = sc.nextInt();

       for (int i = 0; i < n; i++) {
           for (int j = 0; j < n - i - 1; j++)
               System.out.print(" ");

           for (int k = 0; k < i + 1; k++)
               System.out.print("*");

           for (int l = 0; l < i; l++)
               System.out.print("*");
           System.out.println();
       }
   }
}
```

## Code : Triangle of Numbers

Send Feedback

Print the following pattern for the given number of rows.

### Pattern for N = 4

```
···1
··232
·34543
4567654
```

The dots represent spaces.

### Input format :

Integer N (Total no. of rows)

### Output format :

Pattern in N lines

### Constraints :

0 <= N <= 50

### Sample Input 1:

5

**Sample Output 1:**

```
    1
   232
  34543
 4567654
567898765
```

**Sample Input 2:**

```
4
```

**Sample Output 2:**

```
    1
   232
  34543
 4567654
```

```java
import java.util.Scanner;

public class Solution {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int x = 1;

        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n - i - 1; j++)
                System.out.print(" ");
            x = i + 1;
            for (int k = 0; k < i + 1; k++)
                System.out.print(x++);
            for (int l = i * 2; l >= i + 1; l--)
                System.out.print(l);
            System.out.println();
        }
    }
}
```

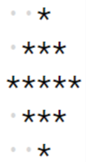
## Code : Diamond of stars

Send Feedback

Print the following pattern for the given number of rows.

**Note: N is always odd.**

**Pattern for N = 5**



```
··*
·***
*****
·***
··*
```

The dots represent spaces.

**Input format :**

N (Total no. of rows and can only be odd)

**Output format :**

Pattern in N lines

**Constraints :**

1 <= N <= 49

**Sample Input 1:**

5

**Sample Output 1:**

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

**Sample Input 2:**

3

**Sample Output 2:**

```
  *
 ***
  *
```

```
import java.util.Scanner;
import java.lang.Math;

public class Solution {
    public static void main(String[] args) {
        // Write your code here
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();

        for (int j = 0; j < n / 2 + 1; j++) {

            for (int k = 0; k < n / 2 - j; k++) {
                System.out.print(" ");
            }
            for (int k = 0; k < j + 1; k++) {
                System.out.print("*");
            }
            for (int k = 0; k < j; k++) {
                System.out.print("*");
            }
            System.out.println();
        }
        for (int j = n / 2; j > 0; j--) {

            for (int k = 0; k < n / 2 - j + 1; k++) {
                System.out.print(" ");
            }
            for (int k = j; k > 0; k--) {
                System.out.print("*");
            }
            for (int k = j - 1; k > 0; k--) {
                System.out.print("*");
            }
            System.out.println();
        }

    }
}
```

## Half Diamond Pattern

Send Feedback

Write a program to print N number of rows for Half Diamond pattern using stars and numbers

**Note : There are no spaces between the characters in a single line.**

**Input Format :**

A single integer: N

**Output Format :**

Required Pattern

**Constraints :**

0 <= N <= 50

**Sample Input 1 :**

```
3
```

**Sample Output 1 :**

```
*
*1*
*121*
*12321*
*121*
*1*
*
```

**Sample Input 2 :**

```
5
```

**Sample Output 2 :**

```
*
*1*
*121*
```

```
*12321*
*1234321*
*123454321*
*1234321*
*12321*
*121*
*1*

*
```

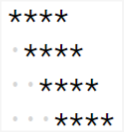
```java
import java.util.Scanner;

public class Solution {

   public static void main(String[] args) {
       // Write your code here
       Scanner sc = new Scanner(System.in);
       int n = sc.nextInt();

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

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

   }
}
```

## Parallelogram Pattern

Send Feedback

Write a program to print parallelogram pattern for the given N number of rows.

**For N = 4**



```
****
·****
··****
···****
```

The dots represent spaces.

**Input Format :**

A single integer : N

**Output Format :**

Required Pattern

**Constraints :**

0 <= N <= 50

**Sample Input 1 :**

3

**Sample Output 1 :**

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

**Sample Input 2 :**

5

**Sample Output 2 :**

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

```java
import java.util.Scanner;

public class Main {

   public static void main(String[] args) {
       // Write your code here
       Scanner sc = new Scanner(System.in);
       int n = sc.nextInt();

       for (int i = 0; i < n; i++) {
           for (int j = 0; j < i; j++) {
               System.out.print(" ");
           }
           for (int j = 0; j < n; j++) {
               System.out.print("*");
           }
```

```
            System.out.println();
        }
    }
}
```

## Sum Pattern

Send Feedback

Write a program to print triangle of user defined integers sum.

**Input Format :**

A single integer, N

**Output Format :**

Required Pattern

**Constraints :**

0 <= N <= 50

**Sample Input 1 :**

3

**Sample Output 1 :**

```
1=1
1+2=3
1+2+3=6
```

**Sample Input 2 :**

5

**Sample Output 2 :**

```
1=1
1+2=3
1+2+3=6
1+2+3+4=10
1+2+3+4+5=15
```

```
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {
        // Write your code here
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int sum = 0;

        for (int i = 1; i <= n; i++) {
            sum = 0;
            for (int j = 1; j <= i; j++) {
                if (j == i)
                    System.out.print(j);
                else
```

```
                    System.out.print(j + "+");
                sum += j;
            }
            System.out.printf("=" + sum);
            System.out.println();
        }
    }
}
```

## Odd Square

Send Feedback

Write a program to print the pattern for the given N number of rows.

For N = 4

```
1357

3571

5713

7135
```

**Input Format :**

A single integer: N

**Output Format :**

Required Pattern

**Constraints :**

0 <= N <= 50

**Sample Input 1 :**

```
3
```

**Sample Output 1 :**

```
135
351
513
```

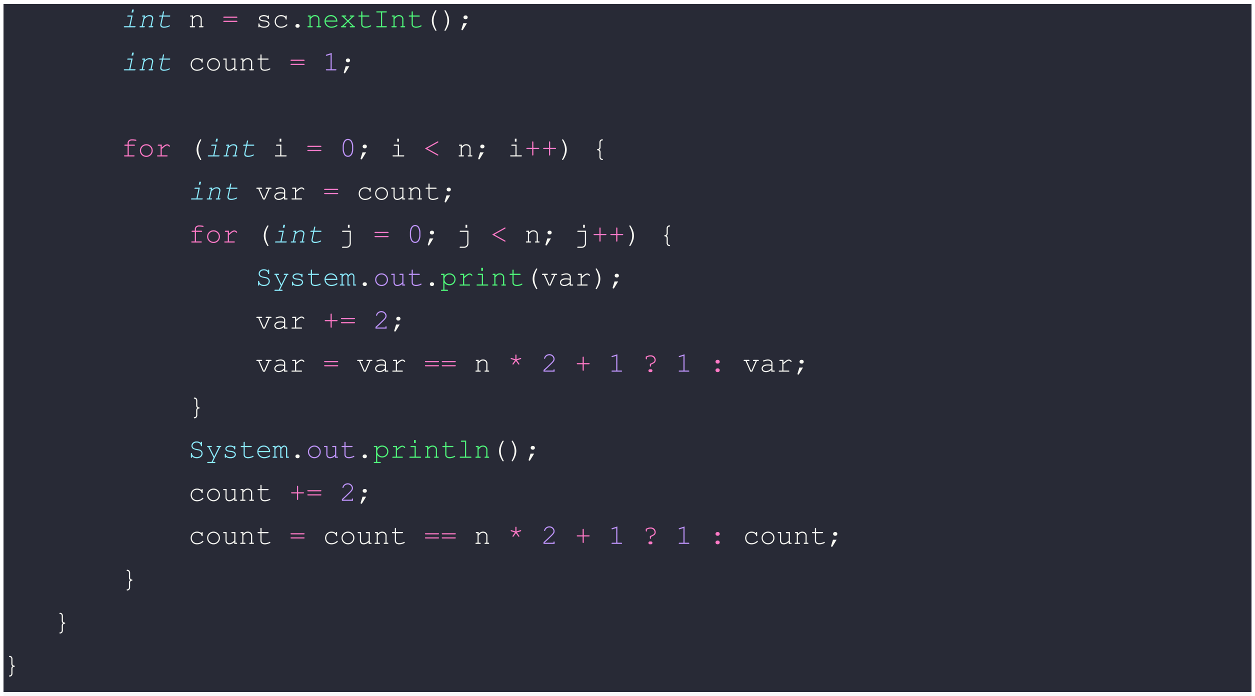
**Sample Input 2 :**

```
5
```

**Sample Output 2 :**

```
13579
35791
57913
79135
91357
```

```
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {
        // Write your code here
        Scanner sc = new Scanner(System.in);
```



```
        int n = sc.nextInt();
        int count = 1;

        for (int i = 0; i < n; i++) {
            int var = count;
            for (int j = 0; j < n; j++) {
                System.out.print(var);
                var += 2;
                var = var == n * 2 + 1 ? 1 : var;
            }
            System.out.println();
            count += 2;
            count = count == n * 2 + 1 ? 1 : count;
        }
    }
}
```