

# Lesson:



## Pattern Problems based on loops



# Pre-Requisites

- Basic Java syntax
- Conditionals
- Good understanding of loops (for, while, do while)

Have you gone through the previous lecture/ lesson plan? If not, we highly recommend you to be thorough with the previous lectures and assignments on loops to have a well-informed discussion here.

## List of Concepts Involved

- Pattern-based questions

Did you know, you can actually draw the basic shapes on the output screen with the knowledge acquired so far?

Yes, we are serious! You absolutely can!

Now that you are curious, let us not waste time arguing and get to work straight away!

Are we ready?

In the questions ahead, we are going to draw patterns and shapes using an asterisk(\*), characters(alphabets) and numbers.

## Topic: Rectangular pattern

The problems discussed here would involve printing a rectangular pattern of different types with the help of the concept of loops.

**Problem 1:** Given height h and width w, print a rectangular pattern as shown in the example below.

**Example:** h=3,w=6

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

**Solution:**

```
public static void main(String args[])
{
int h=3,w=6, i, j;

for(i = 1; i <= h; i++)
{
    for(j = 1; j <= w; j++)
    {
        System.out.print("*");
    }
    System.out.print("\n");
}
}
```

### Explanation:

- Input the height and width; i.e h=3, w=6 respectively.
- For the height of the rectangle, run the outer loop from 1 to height.  
`for (i = 1; i <= h; i++)`
- For the columns of the rectangle, run the inner loop from 1 to width.  
`for (j = 1; j <= w; j++)`
- Print star.
- After printing all columns of a row, print a new line after the inner loop.

**Problem 2:** Given height h and width w, print a rectangular pattern as shown in the example below.

**Example:** h=4,w=6

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

### Solution

```
public static void main(String args[])
{
int h=4,w=6;
for (int i = 0; i < h; i++)
{
    System.out.println();
    for (int j = 0; j < w; j++)
    {
        // Print * if this is first
        // row or last row. Or this
        // column is first or last.
        if (i == 0 || i == h-1 ||
            j == 0 || j == w-1)
            System.out.print("*");
        else
            System.out.print(" ");
    }
}
```

### Explanation:

- Input the height and width of the rectangle; i.e h=4, w=6 respectively.
- For height of the rectangle run the outer loop from 0 to h-1.  
`for (i = 0; i < h; i++)`
- For width of the rectangle run the inner loop from 0 to w-1.  
`for (j = 0; j < w; j++)`
- Print a star for the first or last row (h value) or for the first or last column (w value), otherwise print blank space.
- After printing all columns of a row, print a new line after the inner loop.

**Problem 3:** Given height h and width w, print a rectangular pattern as shown in the example below.

**Example:** h=4,w=6

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

### Solution

```
public static void main(String args[])
{
int h=4,w=6, i, j;

for(i = 1; i <= h; i++)
{
    for(j = 1; j <= w; j++)
    { if(j%2==0){
        System.out.print(".");
    }
    else{
        System.out.print("*");
    }
    System.out.print("\n");
}
}
```

### Explanation:

- Input the height and width; i.e h=4, w=6 respectively.
- For height of the rectangle run the outer loop from 1 to h.  
`for (i = 1; i <= h; i++)`
- For width of the rectangle run the inner loop from 1 to w  
`for (j = 1; j <= w; j++)`
- Print . if j is even, otherwise print \*
- After printing through the width(w) of a given height(h), print a new line after the inner loop.

## Topic: Triangular pattern

After a good contemplation on rectangular patterns, let us now move to draw the next shape viz. triangle.

**Problem 1:** Given n, print a triangular pattern as shown in the example below.

**Exampleif n=4**

```
*
```

```
**
```

```
***
```

```
****
```

### Solution

```

public static void main(String args[])
{
    int n=4;
    for(int i=1; i<=n; i++)
    {
        for(int j=1; j<=i; j++)
        {
            System.out.print("*");
        }
        System.out.println();
    }
}

```

### Explanation:

- Input number of rows i.e n=4
- For rows of the triangle run the outer loop from 1 to rows.  
`for (i = 1; i <= n ;i++)`
- For the columns of the triangle run the inner loop from 1 to i.  
`for (j = 1; j <= i; j++)`
- In the above pattern, you can observe that ith row has i elements
- Print star.
- After printing all columns of a row, print a new line after the inner loop.

**Problem 2 :** Given n, print a triangular pattern as shown in the example below.

**Example if n=4**

```

 ****
  ***
   **
  *

```

### Solution

```

public static void main(String args[])
{
    int n=4;
    for(int i=n; i>=1; i--)
    {
        for(int j=1; j<=i; j++)
        {
            System.out.print("*");
        }
        System.out.println();
    }
}

```

### **Explanation:**

- Input the number of rows i.e n=4
- For rows of the triangle run the outer loop from rows to 1.  
for (*i* = n; *i* >= 1; *i*--)
- For the columns of the triangle run the inner loop from 1 to *i*.  
for (*j* = 1; *j* <= *i*; *j*++)
- In the above pattern, you can observe that *i*th row has (n-*i*+1) elements
- Print star.
- After printing all columns of a row, print a new line after the inner loop.

### **Try these Set 1:**

**Problem 1** Print a triangular pattern as shown in the example below.

**Example for n=4**

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

(Try it yourself first and verify the code/approach with the solution provided at the end of this lesson plan)

## **Topic: Numerical Rectangular pattern**

Numbers always fascinate a programmer. So why not explore the patterns based on numbers ? Ready ? 1,2,3... let's start !!

**Problem1** Given n, print a rectangular pattern as shown in the example below.

**Example if n=7**

```
1234567  
2345671  
3456712  
4567123  
5671234  
6712345  
7123456
```

**Solution:**

```
public static void main(String args[])
{ int n=7;
    for(int i=1;i< n+1 ;i++)
    {
        for(int j=i; j < n+1 ;j++)
        {
            System.out.print(j + " ");
        }
        for(int k=1; k < i ;k++)
        {
            System.out.print(k + " ");
        }
        System.out.println();
    }
}
```

### **Explanation:**

- Input the number of rows i.e n=7
- For rows of the rectangle run the outer loop from 1 to rows.  
`for (i = 1; i < n+1 ; i++)`
- For the columns of the rectangle run the inner loop from i to n.  
`for(int j=i; j < n+1 ;j++)`  
Print the value of j here
- For the columns of the rectangle run the inner loop from 1 to i-1.  
`for(int k=1; k < i ;k++)`  
Print the value of k here
- After printing all columns of a row, print a new line after the inner loop.

### **Try these Set 2:**

(Try these yourself first and verify the code/approach with the solution provided at the end of this lesson plan)

**Problem 1:** Print a rectangular pattern as shown in the example below.

**Example if row=4,col=6**

```
123456
123456
123456
123456
```

**Problem 2:** Print a rectangular pattern as shown in the example below.

**Example if row=4,col=6**

```
123456
 1   6
 1   6
123456
```

**Problem 3 :** Given n, print a rectangular pattern as shown in the example below.

**Example if rows=4,col=6**

```
121212
212121
121212
212121
```

## **Topic: Numerical Triangular pattern**

**Problem:** Given n, print a triangular pattern as shown in the example below.

**Example if n=4**

```
1
12
123
1234
```

**Solution:**

```

public static void main(String args[])
{
    int n=4;

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

        System.out.println();
    }
}

```

**Explanation:**

- Input the number of rows i.e n=4
- For rows of the rectangle run the outer loop from 1 to rows.  
`for (i = 1; i <= n ; i++)`
- For columns of the triangle run the inner loop from 1 to i.  
`for (j = 1; j <= i; j++)`  
print the value of j here.
- In the above pattern, you can observe that ith row has i elements
- After printing all columns of a row, print a new line after the inner loop.

**Try these Set-3:**

(Try it yourself first and verify the code/approach with the solution provided at the end of this lesson plan)

**Problem 1** Given n, print a triangular pattern as shown in the example below.

**Example if n=4**

```

1
121
12321
1234321

```

**Problem 2** Given n, print a triangular pattern as shown in the example below.

**Example if n=4**

```

1
2 2
3 3
4 4

```

Hope you enjoyed solving these problems. Do not forget to see the solutions to unsolved problems. Scroll down further.

# Upcoming Class Teasers

- Number system (decimal and binary concept)

## Solutions to 'Try these Sets'

### Set -1

#### Solution 1:

```
import java.io.*;
import java.util.*;
public class Main
{
    public static void main(String[] args) {
        int rows;

        Scanner sc = new Scanner(System.in);
        rows=sc.nextInt();

        for(int i = 1, k = 0; i <= rows; ++i, k = 0)
        {
            for(int s = 1; s <= rows-i; ++s)
            {
                System.out.print(" ");
            }

            while(k != 2*i-1)
            {
                System.out.print("*");
                ++k;
            }
            System.out.println("");
        }
    }
}
```

## Set -2

### Solution 1:

```
import java.io.*;
import java.util.*;
public class Main
{
    public static void main(String[] args) {
        int rows,col;

        Scanner sc = new Scanner(System.in);
        rows=sc.nextInt();
        col=sc.nextInt();
        for(int i=1;i<=rows;i++){
            for(int j=1;j<=col;j++){
                System.out.print(j);
            }
            System.out.println("");
        }
    }
}
```

**Solution 2:**

```
import java.io.*;
import java.util.*;
public class Main
{
    public static void main(String[] args) {
        int i, j, rows, cols;

        Scanner sc = new Scanner(System.in);
        rows=sc.nextInt();
        cols=sc.nextInt();

        for(i = 1; i <= rows; i++)
        {
            for(j = 1; j <= cols; j++)
            {
                if (i == 1 || i == rows || j == 1 || j == cols )
                {
                    System.out.print(j);
                }
                else
                {
                    System.out.print(" ");
                }
            }
            System.out.println("");
        }
    }
}
```

**Solution 3:**

```
import java.io.*;
import java.util.*;
public class Main
{
    public static void main(String[] args) {
        int i, j, rows, cols;

        Scanner sc = new Scanner(System.in);
        rows=sc.nextInt();
        cols=sc.nextInt();

        for(i = 1; i <= rows; i++)
        {
            for(j = 1; j <= cols; j++)
            {
                if ((i+j)%2==0 )
                {
                    System.out.print(1);
                }
                else
                {
                    System.out.print(2);
                }
            }
            System.out.println("");
        }
    }
}
```

## Set -3

### Solution 1:

```
import java.io.*;
import java.util.*;
public class Main
{
    public static void main(String[] args) {
        int i, j, k, rows;

        Scanner sc = new Scanner(System.in);
        rows = sc.nextInt();

        for(i = rows+1; i > 1; i--)
        {
            for(j = i; j > 2; j--)
            {
                System.out.print(" ");
            }
            for(k=1; k <= (rows-i+1); k++)
            {
                System.out.print(k);
            }
            for(int m=k; m >= 1; m--)
            {
                System.out.print(m);
            }
            System.out.println("");
        }
    }
}
```

**Solution 2:**

```
import java.io.*;
import java.util.*;
public class Main
{
    public static void main(String[] args) {
        int i, j, k, rows;

        Scanner sc = new Scanner(System.in);
        rows = sc.nextInt();

        for(i = rows+1; i > 1; i--) {
            for(j = i; j > 2; j--) {
                System.out.print(" ");
            }
            for(k=1;k<=(rows-i+1);k++) {
                if(k==1){ System.out.print(rows-i+2);}
                else{ System.out.print(" ");}
            }
            for(int m=k;m>=1;m--) {
                if(m==1){ System.out.print(rows-i+2);}
                else{ System.out.print(" ");}
            }
            System.out.println("");
        }
    }
}
```

**End of lesson**