

Revision.

6 $n=5$.

5 rows.

(2)

★	-	-	-	★	...	1
★	-	-	-	★	...	2
★	-	-	-	★	...	3
★	-	-	-	★	...	4
★	★	★	★	★	...	5

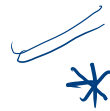
→ same

→ n stars.

★ $\overbrace{\quad\quad\quad}^{n-2}$ ★

```
9  int n = scn.nextInt(); // 5
10  → for(int row = 1; row <= n; row++){
11      if(row == n){
12          // n stars
13          for(int cst = 1; cst <= n; cst++){
14              System.out.print("*");
15          }
16      }
17      else{ // 1-4
18          System.out.print("*");
19          for(int csp = 1; csp <= n-2; csp++){
20              System.out.print(" ");
21          }
22          System.out.print("*");
23      }
24
25      System.out.println();
26  }
27
28 }
```

n = 5



- - -



Pattern 9 - Square Ladder with top and bottom

$n=5$

Problem

Submissions

Leaderboard

Discussions

Take n as an integer input, then

print n tab separated stars in the first line,

then in the second line print a star, then $n-2$ tabs, then print a star.

then print n tab separated stars in the third line.

then in the **fourth** line print a star, then $n-2$ tabs, then print a star .

```
★ ★ ★ ★ ★ ... 1
★                               ★ ... 2
★ ★ ★ ★ ★ ... 3
★                               ★ ... 4
★ ★ ★ ★ ★ ... 5
```

$n=6$

```
★ ★ ★ ★ ★ ★ n
★                               ★
★ ★ ★ ★ ★ ★ n
★                               ★
★ ★ ★ ★ ★ ★ n
★                               ★
★ ★ ★ ★ ★ ★ n
★                               ★
```

$n=5$

* * * * * ... 1

* * * * * ... 2

* * * * * ... 3

* * * * * ... 4

* * * * * ... 5

row \rightarrow odd \rightarrow n stars

\hookrightarrow even \rightarrow * $n-2$ sp. *

```

4 public class Solution {
5
6     public static void main(String[] args) {
7         Scanner scn = new Scanner(System.in);
8         int n = scn.nextInt();
9
10        for(int row = 1; row <= n; row++){
11            if(row % 2 == 0){
12                //row is even
13                System.out.print("\t\t");
14                for(int csp = 1; csp <= n-2; csp++){
15                    System.out.print("\t\t");
16                }
17                System.out.print("\t\t");
18            }
19            else{
20                // row is odd: n stars
21                for(int cst = 1; cst <= n; cst++){
22                    System.out.print("\t\t");
23                }
24            }
25            System.out.println();
26        }
27    }
28
29 }
30
31 }

```

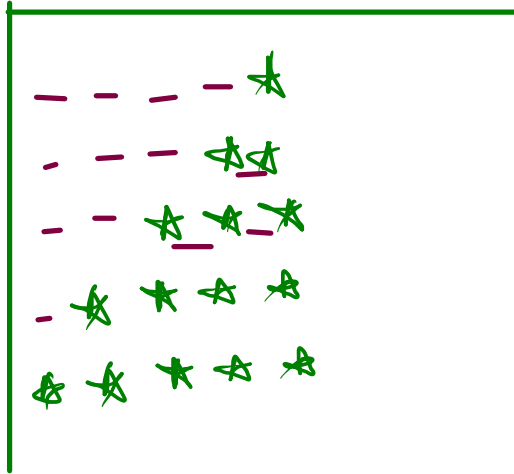
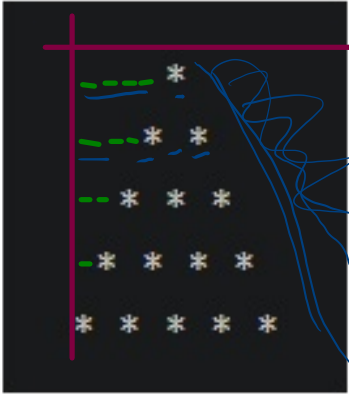
$n=5$

$n=2 \rightarrow \odot$
 $4 \rightarrow e$

1. ★ _ ★ _ ★ _ ★
2. ★ _ _ _ ★
3. ★ ★ ★ ★
4. ★ _ _ _ ★
5. (

Pyramid.

$n=5$



```

6 public static void main(String[] args) {
7     Scanner scn = new Scanner(System.in);
8     int n = scn.nextInt();
9
10    int star = 1;
11    int space = n-1;
12
13    ✓ for(int row = 1; row <= n; row++){
14        for(int csp = 1; csp <= space; csp++){
15            System.out.print(" ");
16        }
17        for(int cst = 1; cst <= star; cst++){
18            System.out.print("* ");
19        }
20
21        star++;
22        space--;
23        System.out.println();
24    }
25 }
26 }
27 }
28 }

```

$n=5$

star = ~~1~~ 2 3

space = ~~4~~ 2

1. ✓

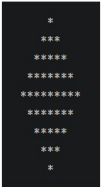
```

- - - - *
- - - * - * -
- - * - * - * -

```

GKSTR29_Pattern_12_Diamond

Take Integer N as input and print the following pattern.



Input Format

An Integer Value N

Constraints

1 <= N <= 100

Output Format

(2*N-1) Line of Pattern as shown in problem statement.

Sample Input 0

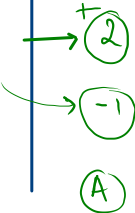
5

Sample Output 0



✓
 $row = 2n - 1$
 $stars = 1$
 $space = n - 1$

↓



row (1) - A 5 - B
(2) - A 6 - B
3 - A 7 - B
4 - A 8 - B
9 - B

Star = 1 2 3 4 5

Space = 4 3 2 1 0

```
for(int row = 1; row <= (2*n-1); row++){
    for(int csp = 1; csp <= space; csp++){
        System.out.print(" ");
    }

    for(int cst = 1; cst <= star; cst++){
        System.out.print("*");
    }

    System.out.println();

    //half
    if(row < n){
        star += 2;
        space--;
    }
    // after half
    else{
        star -= 2;
        space++;
    }
}
```

future.

4 < 5

row = 1

5 > 5

3 < 5

row < n
5 < 5

Dry Run.
Duty

n=5

Sample Output 0

```

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

Chat

n = w = 5

Star = 1 2 3 4 5
Space = 4 3 2 1 0



Doubt Session

Boosters

Tried

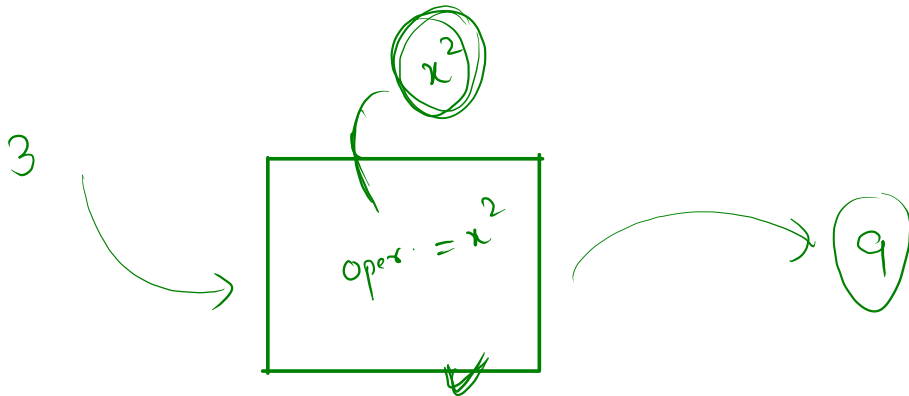
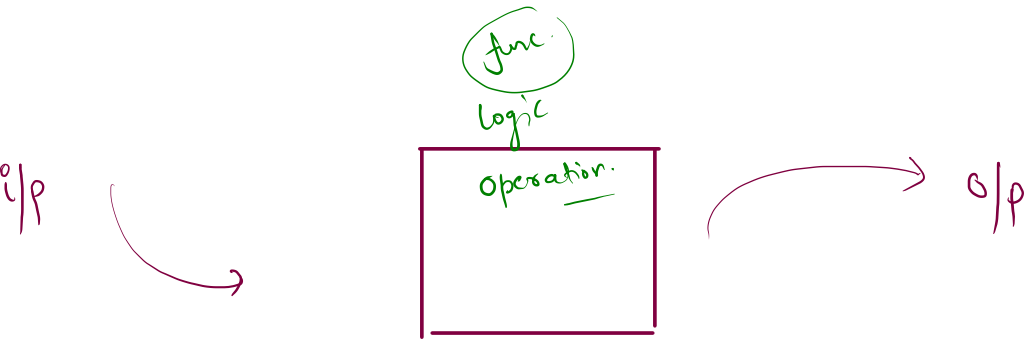
2 - 3

... and

1:1

function.

→ piece/ block of code which is used to perform operation. ✓



www.Amazon.in

name

logic

Hello
Rishabh.

Syed.

Block of code.

Syntax.

return type

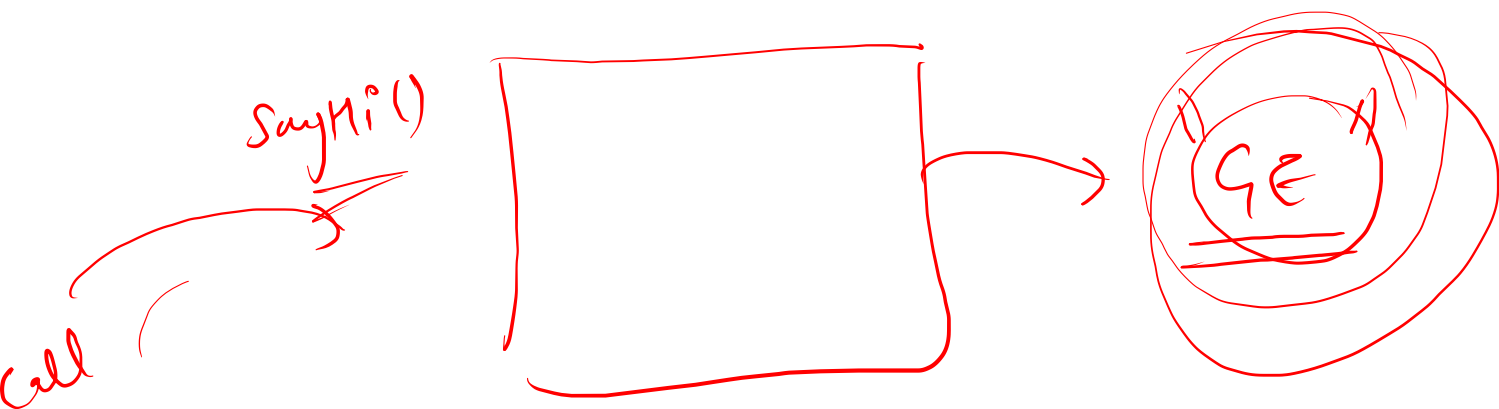
name of funcⁿ

(para 1, para 2)

{

// body → logic / operation.

}



```
2 public class Main
3 {
4
5
6     public static String sayHi(String name){
7         String ans = "Good Evening " + name ;
8         return ans;
9     }
10
11
12     public static void main(String[] args) {
13         String s = sayHi("Neeraj");
14         System.out.println(s);
15
16     }
17 }
18 }
```

main → entry pt.

$$2 - 3$$

without

function

with func.

Factorial of N

Problem

Submissions

Leaderboard

Discussions

A **teacher** is explaining **factorial** to the students. Since, the calculation involved in factorial is a bit difficult for him to do. The teacher wants to write a **program**, so that he doesn't need to **calculate** the **factorial** of every number. You have to help the teacher in writing a program for factorial calculation.

Take an **integer N** as input.

Print the factorial of N.

Note: Factorial of 0 is 1.

factorial,

$$0! = 1$$

$$5! =$$

$$5 \times 4 \times 3 \times 2 \times 1$$

$$6! =$$

$$6 \times 5 \times 4 \times 3 \times 2 \times 1$$

$$3! = 3 \times 2 \times 1$$

$$n! = n \times n-1 \times n-2 \times n-3 \times \dots \times 1$$

Sample Input 0

$N = 5$

Sample Output 0

120 ✓

$N = 5$

$$5! = 5 \times 4 \times 3 \times 2 \times 1$$

$$5! = 120 \quad \checkmark$$

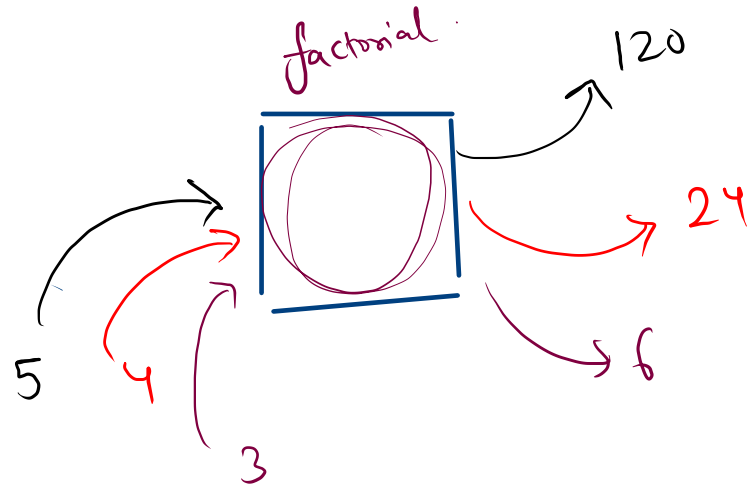
?

for n to 1

```

1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5
6     public static void main(String[] args) {
7         Scanner scn = new Scanner(System.in);
8         int n = scn.nextInt();
9
10        int fact = 1;
11
12        for(int i = n; i >= 1; i--){
13            fact *= i;
14        }
15
16        System.out.println(fact);
17    }
18 }
19

```



Find sum using a function