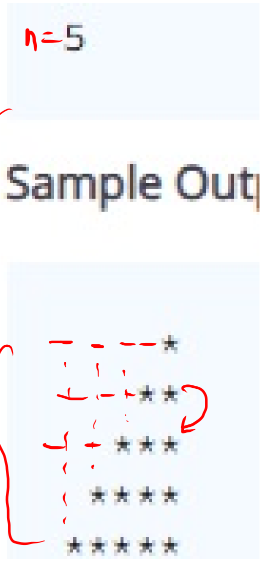
Revision

0th → 4 spaces.

space = (n-1) = ~~4~~ ~~3~~ ~~2~~ ~~1~~ ~~0~~

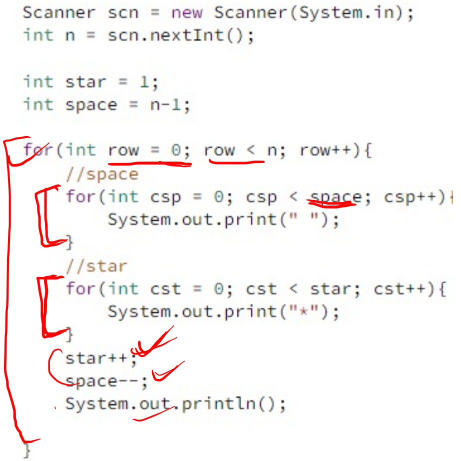
-1

star = ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ 6


n=5

Sample Out

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



```
Scanner scn = new Scanner(System.in);
int n = scn.nextInt();

int star = 1;
int space = n-1;

for(int row = 0; row < n; row++){
    //space
    for(int csp = 0; csp < space; csp++){
        System.out.print(" ");
    }
    //star
    for(int cst = 0; cst < star; cst++){
        System.out.print("*");
    }
    star++;
    space--;
    System.out.println();
}
```

row

0 1 2 3 4 5

0<5 ✓

1<5 ✓

2<5 ✓

3<5 ✓

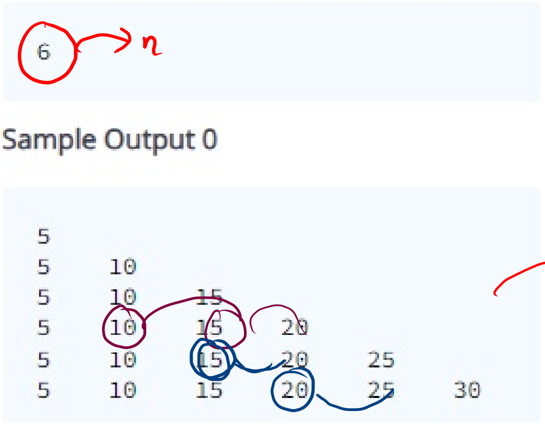
4<5 ✓

5<5 ✗

```
- - - - *
- - - **
- - ***
- ****
*****
```

Pattern -6.

row? = n.

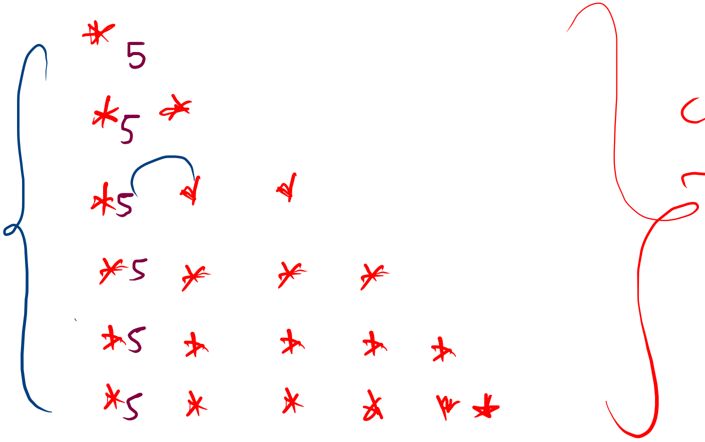


```
6
```

Sample Output 0

```
5
5       10
5       10      15
5       10      15      20
5       10      15      20      25
5       10      15      20      25      30
```

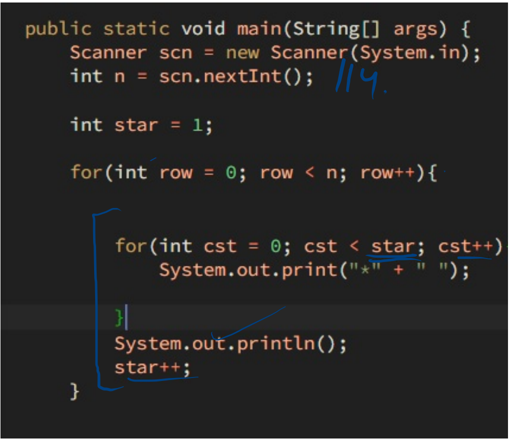
val = 5.



code ?

Yes.

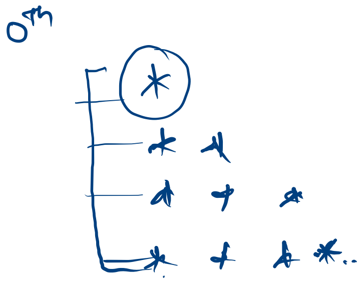
Pattern-4.



```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt(); //4.

    int star = 1;

    for(int row = 0; row < n; row++){

        for(int cst = 0; cst < star; cst++)
            System.out.print("*" + " ");

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

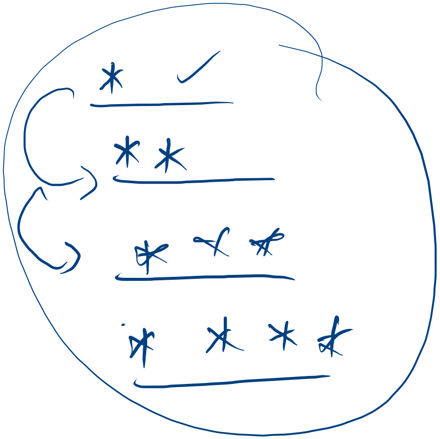
n = 4

row = n



star = ~~1~~ ~~2~~ 3

r = 1

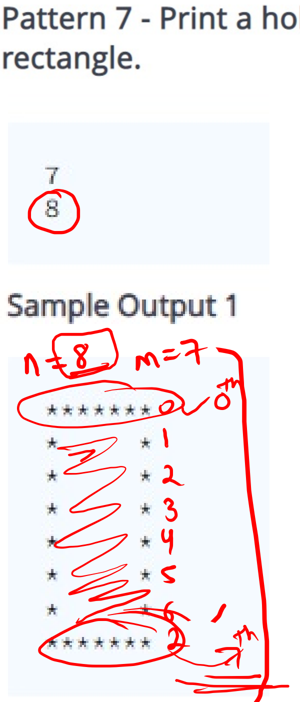


cst = ~~0~~ ~~1~~ 2

0 < 2 ✓

1 < 2 ✓

2 < 2 ✗



Pattern 7 - Print a hollow m by n star rectangle.

```
7
8
```

Sample Output 1

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

n=8 m=7

0th

1

2

3

4

5

6

7th

row = 0

row = m-1

n x m

= row

= col

m → col = 7

n → rows (lines) = 8

* _ _ _ _ *

0
1
2
3
4
5
6
7

8 rows.

Space = m-2

7 stars.

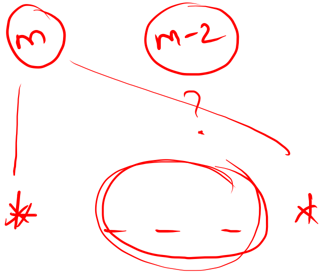
m=5 (* * * * *)

n=4. ___ (rows)

```
0  * * * * *  ✓
1  * _ _ _ *
2  * _ _ _ *
3  * * * * *  ✓
```

0 or n-1

m=5
n=4



m-2

```
                    ·m
0  * * * * *  ✓
1  *       *    2
2  *       *
3  * * * * *  ✓
```

m=5
n=4

i=0 || i~~A~~=n/2

n/2

m=4

n=6

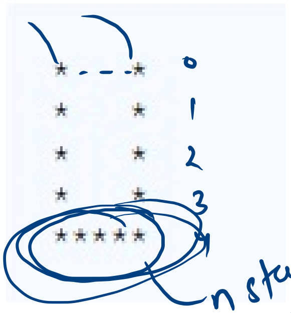
```
0  * * * *
1  *     *
2  * ___ *
3  *     *
   *     *
   * * * *
```

Pattern 8 - Print a hollow square without top

row? = n

n= 5

Sample Output (

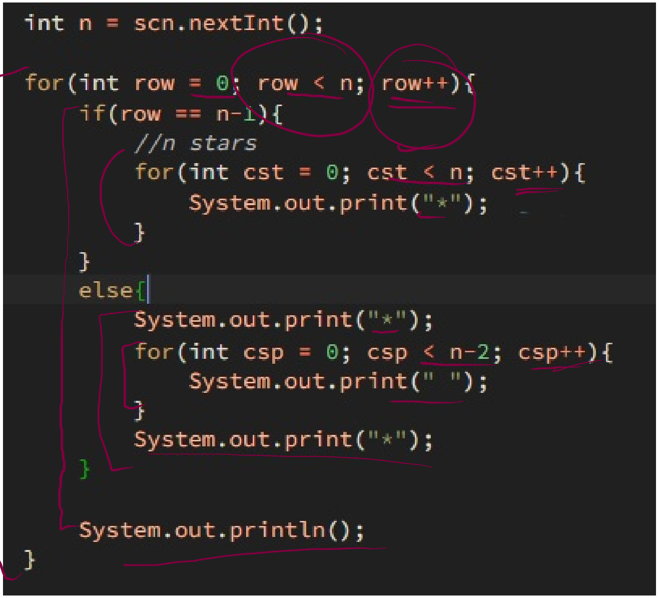


```
*   *   0
*   *   1
*   *   2
*   *   3
*****   4
```

n stars.

n-2

```
if ( row == n-1 )
{

}
else
{

}
```



```
int n = scn.nextInt();

for(int row = 0; row < n; row++){
    if(row == n-1){
        //n stars
        for(int cst = 0; cst < n; cst++){
            System.out.print("*");
        }
    }
    else{
        System.out.print("*");
        for(int csp = 0; csp < n-2; csp++){
            System.out.print(" ");
        }
        System.out.print("*");
    }

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

n=5.

row=~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ 5

0<5✓

1<5✓

5<5 → f

cst=~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ 5

0<5✓

1<5✓

2<5✓

3<5✓

4<5✓

5<5 f

```
*---*
*---*
*---*
*---*
* * ***
```

Pattern-9

n=5

n => odd.

5 rows

n rows.

```
*   *   *   *   *
*   ___   _     *
*   *   *   *   *
*     n .  .  . *
*   *   *   *   *
```

n-2

odd even concept.

n=7

0 . . . n-1

1 . . . n

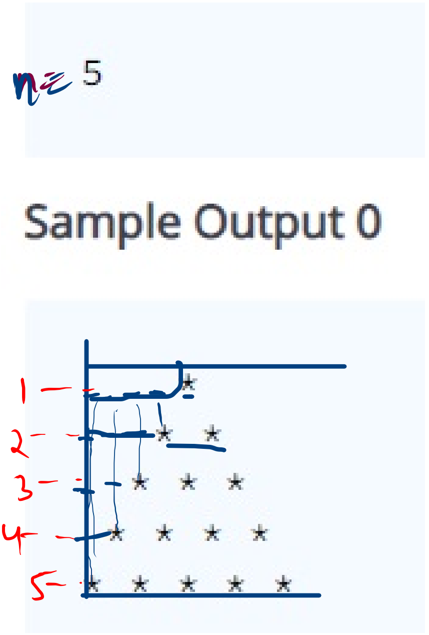
row
↳ odd → all star
↳ even → * — *

n rows.

```
*  *  *  *  *  *  *   → 0    1
*      _ 5 _      *   → 1    2
*  *  *  *  *  *  *   → 2    3
*      _ 5 _      *   → 3    4
*  *  *  *  *  *  *   → 4    5
*      _ 5 _      *   → 5    6
*  *  *  *  *  *  *   → 6    7
```

Pyramid.

rows = ? → n rows.



n = 5

Sample Output 0

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

1st { star = 1
space = n-1 }

```
for ( n rows )
{
    — space —
    — star —
}
```

next
star++
space--