Todays Content:

- ~ 10 Questions
- → Observation

Pattern-1 :

→ gap not indicating space

Input	Output
N = 3	* * *
N = 4	* * * *
N = 5	* * * * *
N = 7	* * * * * * *

Pseudocode :

↳ Editor

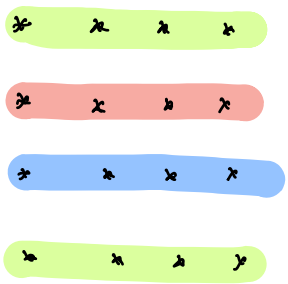
Pattern 2: → Matrix: —— : rows | : columns

Input Output

N=2

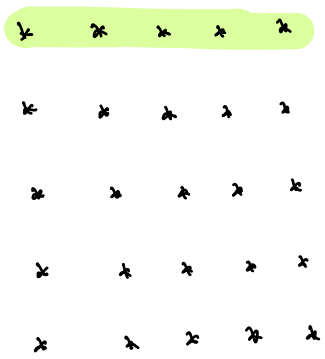
N=3

N=4



```
for(int i=1; i<=4; i++){
    for(int j=1; j<=4; j++){
        Sop(*);
    }
}
```

N=5



```
for(int i=1; i<=5; i++){
    for(int j=1; j<=5; j++){
        Sop(*);
    }
}
```

// Writing loop inside loops Nested loops

// Pseudocode :

```
int N;
Read N;
for(int i=1; i<= N; i++){
    for(int j=1; j<= N; j++){ , * * ... * (N times)
        Sop(*);
    }
    System.out.Println()
}
```

Pattern: 3

→ outerloop - Iterate rows

→ Inner loop Iterate Columns

N = 1 *

N = 2

1 *

2 * *

N = 3

1 *

2 * *

3 * * *

N = 4

1 *

2 * *

3 * * *

4 * * * *

N = 5

1 *

2 * *

3 * * *

4 * * * *

5 * * * * *

→ N: N Rows

Columns

→ In ith Row = i Columns

Pseudocode:

```
for(i = 1; i <= N; i++) {
    for(j = 1; j <= i; j++) {
        Sop(*)
    }
    System.out.println();
}
```

outerloop no: of rows

Inner loop

Pattern-4

// given N → N rows

↓

i^{th} row + # Colums in i^{th} row = N+1

#No: of column in i^{th} row = N+1-i

N=1 *

N=2

row		col	sum
1	* *	2	3
2	*	1	3

N=3

row		col	sum
1	* * *	3	4
2	* *	2	4
3	*	1	4

N=4

row		N+1-i	col	sum
1	* * * *	5-1=	4	5
2	* * *	5-2=	3	5
3	* *	5-3=	2	5
4	*	5-4=	1	5

N=5

row		N+1-i	col	sum
1	* * * * *	6-1=	5	6
2	* * * *	6-2=	4	6
3	* * *	6-3=	3	6
4	* *	6-4=	2	6
6	*	6-5=	1	6

N=1 *

N=2
2 * * : 2
1 * : 1

N=3
3 * * * : 3
2 * * : 2
1 * : 1

N=4
4 * * * * : 4
3 * * * : 3
2 * * : 2
1 * : 1

N=5
5 * * * * * : 5
4 * * * * : 4
3 * * * : 3
2 * * : 2
1 * : 1

```
// given N how many rows = N
for(i=N; i>=1; i=i-1)
    in ith row how how many colums i colums
    for(j=1; j<=i; j=j+1){
        Sop(*);
    }
    Soprintln()
}
```

Pattern - 5 With Space → ↦ indicating space

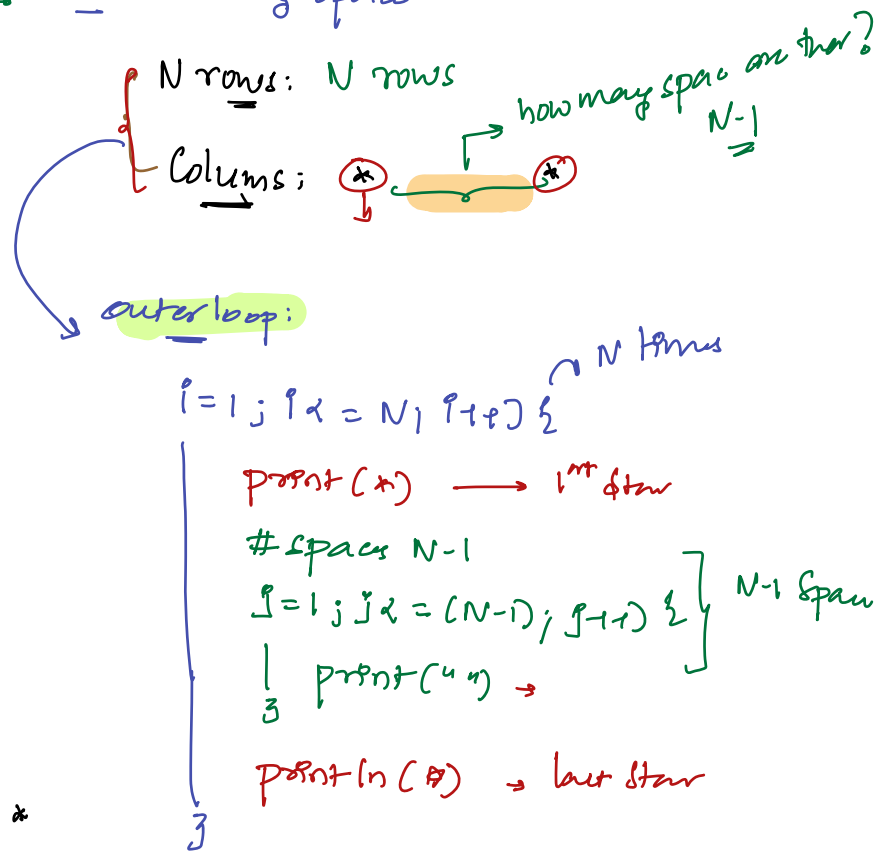
N = 1 * *

N = 2 * _ *
* _ *

N = 3 1 * _ _ *
2 * _ _ *
3 * _ _ *

N = 4 1 * _ _ _ *
2 * _ _ _ *
3 * _ _ _ *
4 * _ _ _ *

N = 5 * _ _ _ _ *
* _ _ _ _ *
* _ _ _ _ *
* _ _ _ _ *
* _ _ _ _ *



Pattern - 6

N = 1 * *

N = 2

1	* _ *	1	2 - 1
2	* *	0	2 - 2

N = 3

1	* _ _ *	2	3 - 1
2	* _ *	1	3 - 2
3	* *	0	3 - 3

N = 4

1	* — — — *	3	4 - 1
2	* — — *	2	4 - 2
3	* — *	1	4 - 3
4	* *	0	4 - 4

N = 5

1	* — — — — *	4	5 - 1
2	* — — — *	3	5 - 2
3	* — — *	2	5 - 3
4	* — *	1	5 - 4
5	* *	0	5 - 5

given N

1) N rows

Row number increasing: [1, N]

- print(i)
- In i^{th} row: N-i space
- println()

Pseudocode:

```
for(i=1; i<=N; i++){
    print(i)
    for(j=1; j<=(N-i); j++){
        print("_")
    }
    println("i");
}
```

TODO:

→ Row number in decreasing way

Pattern-7: ⟶

N = 1; *

N = 2:

		Stars	Spa	N-i
1	_ *	1	1 = 2-1	
2	* *	2	0 = 2-2	

N = 3;

		St	S	N-i
1	_ _ *	1	2	3-1
2	_ * *	2	1	3-2
3	* * *	3	0	3-3

N = 4;

		St	S	N-i
1	_ _ _ *	1	4	4-1
2	_ _ * *	2	3	4-2
3	_ * * *	3	2	4-3
4	* * * *	4	1	4-4

N = 5

		St	S	N-i
1	_ _ _ _ *	1	5	5-1
2	_ _ _ * *	2	4	5-2
3	_ _ * * *	3	3	5-3
4	_ * * * *	4	2	5-4
5	* * * * *	5	1	5-5

given N:

rows : N rows

- At i^{th} row : N-i space
- At i^{th} row : i stars

Pseudocode:

```
for(i=1; i<=N; i++){
    // In Every row
    // N-i space
    for(j=1; j<=N-i; j++){
        print(_)
    }
    // i stars
    for(j=1; j<=i; j++){
        print(*)
    }
    println();
}
```

Pattern - 1 :

N → N rows

↳ # rows → N rows (0 to n-1)

↳ At i^{th} row : i stars

At i^{th} row : N-1 spaces

N = 1

1 *

N = 2

1 * _

2 * *

N = 3

1 * _ _

2 * * _

3 * * *

N = 4

1 * _ _ _

2 * * _ _

3 * * * _

4 * * * *

N = 5

1 * _ _ _ _

2 * * _ _ _

3 * * * _ _

4 * * * * _

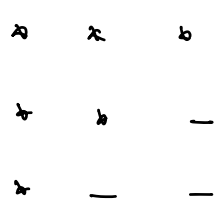
5 * * * * *

Pattern-9 of 3 ⟶ Pattern-10 of 3

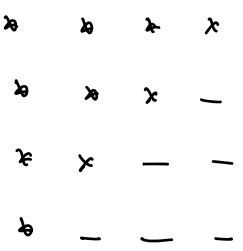
N=1

N=2

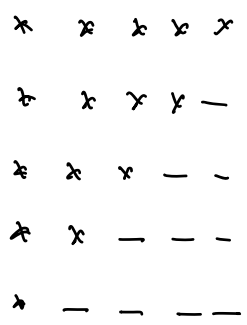
N=3



N=4

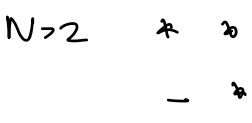


N=5

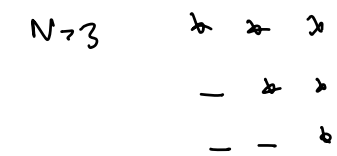


N=1

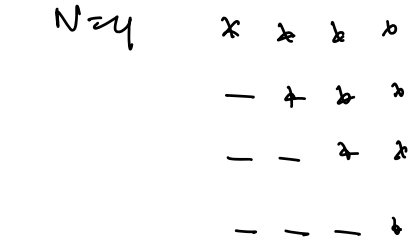
N=2



N=3



N=4



N=5