Patterns

Question 1

Given N as input, print '*' N times.

```
N=5      * * * * *

N=3      * * *
```

run loop N times & print '*'

```
void pattern( int N){
    for(int i=1 ; i<=N; ++i) {
        SOP('*');
    }
}
```

Question 2

Given N as input. Print a square of size NxN containing '*' in each cell.

```
         * * *
N=3      * * *
         * * *
```

```
void pattern( int N){
    for (int i=1 ; i<=N; ++i){
        for(int j=1 ; j<=N;  ++j) {
            SOP('*');
        }
        System.out.println();    //go to next line
    }
}
```

Question 3

Given N,M as input. Print rectangle of size N×M containing '*' in each cell.

N=2
M=3

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

N=5
M=2

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

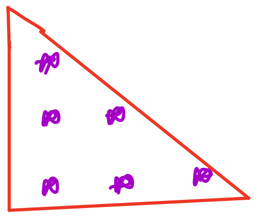
Print M '*' N times.

```
void pattern( int N , int M ){
    for (int i=1 ; i<=N; ++i ){
        for(int j=1 ; j<=M; ++j) {
            SOP('*');
        }
        System.out.println();
    }
}
```

Question 4

Given N as input. Print staircase pattern.

N=3



N=4

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

Row	Stars
1	1
2	2
3	3
⋮	⋮
N	N

```
void pattern( int N){
    for (int i=1; i<=N; ++i){
        for(int j=1; j<=i; ++j) {
            SOP('*');
        }
        System.out.println();
    }
}
```

Question 5

Given N as input. Print the pattern as shown below.

N=3

```
*
*  2
*  2  *
```

N=4

```
*
*  2
*  2  *
*  2  *  4
```

Print regular staircase pattern using '*' but print column no. if it is even.

```
void pattern( int N){
    for (int i=1; i<=N; ++i){
        for(int j=1; j<=i; ++j) {
                if(j%2 ==0)
                    SOP(j);
                else
                    SOP('*');

            }
            System.out.println();

        }
    }
```

Question 6

Given N as input. Print the pattern below.

N>=2

N=3

```
*  _  *
*  _  *
*  _  *
```

N=4

```
*  _  _  *
*  _  _  *
*  _  _  *
*  _  _  *
```

Print N rows & N columns.

1st & last column has '*' and remaining (N-2) columns has '_'.

```
function pattern( int N) {
    for ( int i=1 ; i<=N; ++i) {
        SOP('*');
        for ( int j=1; j<=N-2; ++j) {
            SOP('_');
        }
        SOP('*');
        system.out.println();
    }
}
```

Question 7

Given N as input. Print the pattern below.

N=3

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

N=4

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

N=4

row		star			
1	+	4	=	5	(N+1)
2	+	3	=	5	
3	+	2	=	5	
4	+	1	=	5	

row no. + star count = N+1

star count = N+1 - row no.

```
void pattern( int N){
    for (int i=1 ; i<=N; ++i){
        for(int j=1 ; j <= N+1-i ; j++){
            SOP('*');
        }
        System.out.println();
    }
}
```

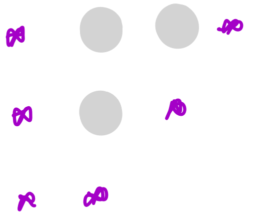
OR

```
void pattern( int N){
    for (int i=N; i >=1; i--){
        for(int j=1 ; j <= i; j++){
            SOP('*');
        }
        System.out.println();
    }
}
```

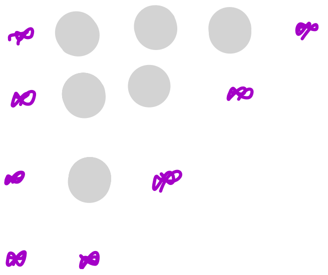
Question 8

Given N as input, print the following pattern.

N=3



N=4



N=4

row		space		
1	+	3	=	4 (N)
2	+	2	=	4
3	+	1	=	4
4	+	0	=	4

Space = N - row

```
function pattern ( int N) {
    for ( int i=1 ; i<=N ; ++i ) {
        SOP('*');
        for ( int j=1; j<= N-i ; ++j ) {
            SOP(' ');

        }
        SOP('*');
```

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

    }
```

Question 9

Given N as input. Print the following pattern.

N=3

```
● ● *
● * *
* * *
```

N=4

```
● ● ● *
● ● * *
● * * *
* * * *
```

N=4

row	space	stars
1	3	1
2	2	2
3	1	3
4	0	4
	= N-row	= row no.

```
for ( int i=1; i<=N; ++i ) {
    for( int j=1 ; j<= N-i ; ++j ) {
        SOP(' ');
    }
    for ( int j=1 ; j<=i; ++j ) {
            SOP('*');
        }
        SOPln();
    }
```

Question 10

Given N as input. Print the pattern below.

N=4

```
* * * * * * * *
* * * ● ● * * *
* * ● ● ● ● * *
* ● ● ● ● ● ● *
```

N=4

2xrow	row	star	space	star
2	1	4	0	4
4	2	3	2	3
6	3	2	4	2
8	4	1	6	1
		=N-row+1	= 2xrow -2	=N-row+1

```
for( int i=1; i<=N; ++i ){
    for( int j=1 ; j<= N-i+1   ; ++j){
        SOP('*');
    }
    for ( int j=1 ; j<= 2*i -2 ; ++j ){
        SOP(' ');
    }
    for( int j=1 ; j<= N-i+1   ; ++j){
        SOP('*');
    }
    SOPln();
}
```

Question 11

Given N as input. Print pattern below.

```
N=4   ● ● ● *
      ● ● * * *
      ● * * * * *
      * * * * * * *

N=3   ● ● *
      ● * * *
      * * * * *
```

N=4

2*row-1	row	space	star
2-1	1	3	1
4-1	2	2	3
6-1	3	1	5
8-1	4	0	7
		= N-row	= 2*row-1

```
for( int i=1; i<=N; ++i ){
    for( int j=1 ; j<=N-i; ++j ){
        SOP(' ');
    }
    for( int j=1 ; j<=2*i-1; ++j){
        SOP('*');
    }
    SOPln();
}
```