Multi dimensional Arrays

↳ 2D Array

↓

Matrix

→ 2D Array

COL ↑

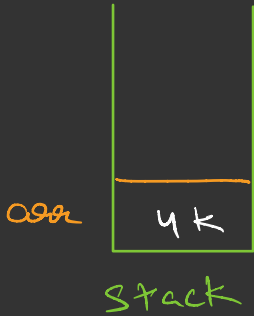int arr [ ] [ ] = new int [3][4]

↓
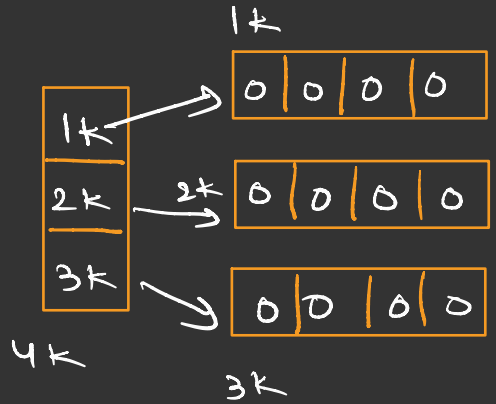Row

Abstract View

	0	1	2	3
0	arr[0][0]	[0][1]	[0][2]	[0][3]
1	[1][0]	[1][1]	[1][2]	[1][3]
2	[2][0]	[2][1]	[2][2]	[2][3]

Memory Management

int arr[][]; ←

arr = new int[3][4];

1k

| 0 | 0 | 0 | 0 |

1k
2k
3k

2k

| 0 | 0 | 0 | 0 |

3k

| 0 | 0 | 0 | 0 |

4k

3k

4k

arr

Stack

Heap

int arr[][] = new int[3][];

arr[0] = new int[2]

arr[1] = new int[4];

| 1k | 2k | n |

5k

| 0 | 0 |

1k

| | | | |

2k

5k

arr
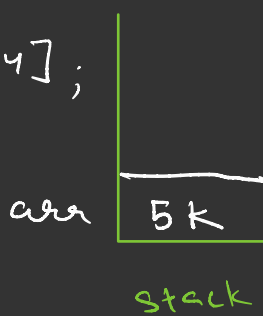
Stack
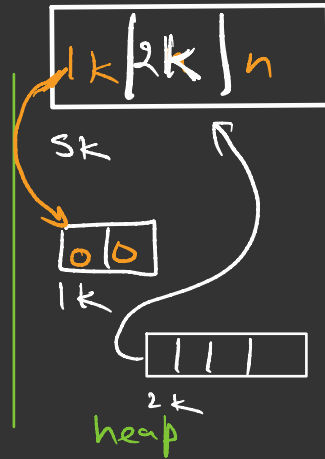
heap

int arr[][] = {{1,2,3}, {1,4,5,6}

Que Display 2D array in rectangle
format   0   1   2   3

    0 (1   2   3   4)

    1   5   6   7   8

    2   9   10   11   12

      00   01   02   03

      10   11   12   13

      20   21   22   23

Ques Print Zig Zag

    1,   2   3   4

    5   6   7   8

    9   10   11   12

   13   14   15   16

→

1
2
3
4
8
7
6
5
9
10
11
12
16
15 14 13

Ques Print Spiral

minc   maxc

1   2   3   4   5   6

minr 7   8   9   10   11   12

13   14   15   16   17   18

maxr 19   20   21   22   23   24

25   26   27   28   29   30

row	fixed	sp	ep	
top-wall	minr	minc	maxc	minr++
right wall	maxc	minr	maxr	maxc --
bottom wall	maxr	maxc	minc	maxr --
left wall	minc	maxr	minr	minc++

t ⟶

r ⟶

b ⟶

⟹ 1, 2, 3 4 5 6   12 18 24 30   29 28

27 26 25   19 13 7 8 9 10 11 17

23 22 21 20   14 15 16