

1	2	3	4	5	6	7	8	9	10	11	12	13
14	15	16	17	18	19	20	21	22	23	24	25	26
27	28	29	30	31								
M	T	W	T	F	S	S	M	T	W	T	F	S

23

March

13th Wk • 082-283

Wednesday

Arrays

8 A list of items of same type is called an Array.

9 Definition of an array (typed)

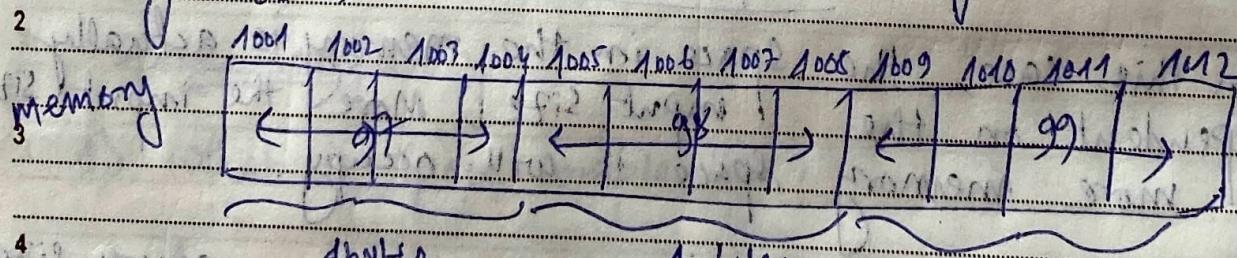
10 $\text{type}[\] \text{ arrayName} = \text{new type}[size];$

11 'new' → this keyword is used to allocate new space in the memory for non-primitive data type.

12 Q) How does memory allocation happen in Array?

→ $\text{int}[\] \text{ marks} = \text{new int}[3];$

1 So, here we happen to take an array called marks of size 3.



2 So, marks[0] = 97; marks[1] = 98; marks[2] = 99;

3 marks[0] = 97; marks[1] = 98; marks[2] = 99;

4 As int type of data has ~~4 bytes~~ 4 bytes size in a 32-bit machine. So, the 3 elements in the marks will occupy $(3 \times 4) = 12$ bytes of memory space.

5 Also, the first element of the array will occupy the initial memory size, then next element & so on.

2022

APRIL	1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30		
M	T	W	T	F	S	S	M	T	W	F

March

13th Wk • 083-282

86

24

Thursday

defining an array (type 2):

type[] arrayName = {1, 2, 3, 4, 5, 6};

✓ If the values of the arrays are not defined explicitly, Java automatically initializes all the elements of the arrays with null() values, or 0 values, depending on the type.

For e.g → for an object, default value is null.

for int, default value is 0.

for float, default value is 0.0.

for boolean, default value is false.

for string, default value is " ".

similarly, for int[] array, all elements of the array will be 0.

E.g → Scanner sc = new Scanner (System.in);

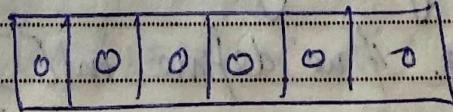
int size = sc.nextInt();

int[] marks = new int[0];

for (int i=0; i<size; i++)

System.out.println(marks[i]);

Memory



marks [size];
if size > 6.

This is called default memory allocation.

In comparison to C/C++ (where garbage value is being stored), this is an advantage to Java where the memory is initialized by default, if nothing is explicitly mentioned in the code.

25

March

13th Wk • 084-281

Friday

2022	87
MARCH	
1	2
3	4
5	6
7	8
9	10
11	12
13	14
15	16
17	18
19	20
21	22
23	24
25	26
27	28
29	30
31	

- 8) Take an array as input from the user. Search for a given number X and print the index of it.
- 9) Find the maximum & minimum number in an array of integers.
- 10) Take an array of numbers as input and check if it is an array sorted in ascending order.

2D Arrays

Java → 0th index language

	0	1	2	3
0				
1			(1,2)	
2				
3		(3,1)		
4				

row = 5 (from 0 to 4)

column = 4 (from 0 to 3)

- 4) ✓ The size of a single cell depends on the type of array.

5) For e.g., if we define int[] array, then size = 4 bytes.
if we define boolean[], then size = 1 byte.

So, Total size of array = (rows x column) x datatype size

E.g., if we define int[] array, & rows = 5, col = 4,

then total size = $(5 \times 4) \times 4$ bytes

$$= 20 \times 4 = 80 \text{ bytes.}$$

2022

	1	2	3	4	5	6	7	8	9	10
	11	12	13	14	15	16	17	18	19	20
	21	22	23	24	25	26	27	28	29	30
	27	28	29	30	31					
MON	TUE	WED	THU	FRI	SAT	SUN				

March

13th Wk • 085-280

26

Saturday

88

Declaration

`type[][] arrayName = new type[][];`

E.g. `int[][] marksClass = new int[10][10];`

Q) How to take input from user for 2D array?

`Scanner sc = new Scanner(System.in);`

`int row = sc.nextInt();`

`int col = sc.nextInt();`

`int[][] numbers = new int[row][col];`

Input `for (int i=0; i<row; i++)` // controlling rows

`{ for (int j=0; j< col; j++)` // controlling columns

~~marksClass~~ `numbers[i][j] = sc.nextInt();`

}

Output

`for (int i=0; i<row; i++)`

{

`for (int j=0; j< col; j++)`

{

`s.o.p(numbers[i][j] + " ");`

`s.o.println();`

}

27

March

13th Wk • 086-279

Sunday

0	1	2	3	4	5	6	7	8	9	10	11	12	13
14	15	16	17	18	19	20	21	22	23	24	25	26	27
28	29	30	31										

M T W T F S S M T W T F S S

Q4) Take a matrix as input from the user. Search for a given number X and print the indices at which it occurs.

Q5) Print spiral order matrix as output for a given matrix of numbers.

E.g →

0	1	2	3	4	5
1	6	5	7	9	10
2	9	10	12	13	20
3	15	25	29	30	32
4	40	70	79	81	95



Outputs →

1, 5, 7, 9, 10, 11, 21, 41, 70, 105, 95, 81, 79, 70, 40, 15, 9, 6, 10, 12, 13, 20, 32, 68, 63, 59, 55, 25, 29, 30, 29.

Strings

✓ It is a non-primitive data type. It also means that users can define and manipulate the length of it.

✓ There is a class in Java called String. So, we can't use that while defining the ~~variable~~ ~~String~~ from user class. Also, we need to use 'S' instead of 's'.

// String declaration

String name = "Tony";

String fullName = ". Tony Stark";