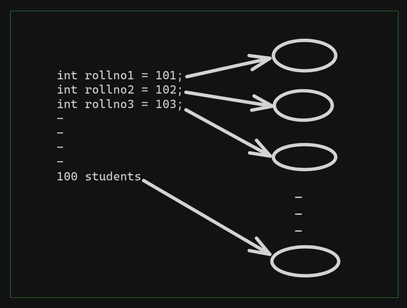
Arrays

⇒ Introduction :-
- → Arrays are the objects which stores the similar of data (homogeneous data)
- → It stores multiple data under a single variable name.

→ Syntax :-
```
int[] rollno = {101, 102, 103, -,-,-,-};
```




```
public class MainApp1
{
    public static void main(String[] args)
    {
        int[] arr = {101, 102, 103, 104, 105, 106, 107, 108, 109, 110};

        System.out.println(arr[2]);

    }
}
```

101	102	103	104	105	106	107	108	109	110
0	1	2	3	4	5	6	7	8	9

→ Disadvantages :-
1. Fixed Size
2. Memory Wastage
3. Stores same data type
4. Performing operations (insertion, deletion) are costly

→ Types of Arrays :-
>> There are 2 types of arrays:
1. Single Dimensional Array
   - 1D Array
2. Multi-Dimensional Array
   - 2D Array
   - Matrix Array
   - Jagged Array
   - Higher Dimensional Array (3D, 4D, 5D.... Array)

⇒ Single Dimensional Array :-
- → It is an array having only single dimensional i.e. only one row or one column
- → For eg. 1D Array

⇒ 1D Array :-

→ Declare 1D Array
- int[] arr; (recommended)
- int []arr;
- int [] arr;
- int arr[];

→ Create 1D Array
- arr = new int[3];

Declaration & creation in single line

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

arr

0	0	0
0	1	2

→ Initialize 1D Array
- arr[0] = 11;
- arr[1] = 22;
- arr[2] = 33;

```
int[] arr = {11, 22, 33};
```

Declaration, creation & Initialization in single line

arr

11	22	33
0	1	2

→ TASK : Create an array and insert elements into array by taking from user i.e. at runtime

→ Retrieve Array Elements :-
- for loop
- for-each loop

⇒ Multi-Dimensional Array :-
- → Multi-Dimensional Arrays are also known as "array or array"
- → For eg 2D, 3D, 4D ... array, matrix array, jagged array

⇒ 2D Array :-

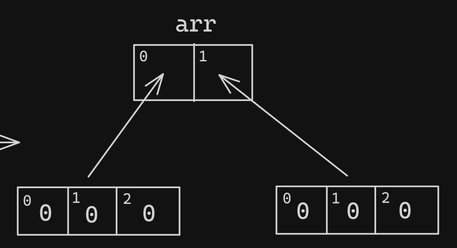
→ Declare 2D Array:
- int[][] arr; (recommended)
- int[]arr[];
- int arr[][];

→ Create 2D Array
- arr = new int[2][3];

```
int[][] arr = new int[2][3];
```

arr

0,0 0	0,1 0	0,2 0
1,0 0	1,1 0	1,2 0

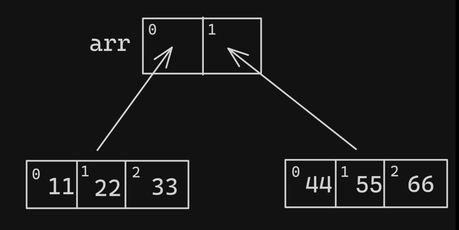


→ Initialize 2D Array:
- arr[0][0] = 11;
- arr[0][1] = 22;
- arr[0][2] = 33;
- arr[1][0] = 44;
- arr[1][1] = 55;
- arr[1][2] = 66;

```
int[][] arr = { {11, 22, 33} , {44, 55, 66} };
```

arr

0,0 11	0,1 22	0,2 33
1,0 44	1,1 55	1,2 66



→ Traverse 2D Array:
- nested for loop
- nested for-each loop

⇒ JAGGED Array