

name of → will acts like  
Array Reference to that array

Q) write code for pattern

# Array

→ It is collection of similar elements.  
Elements of multiple datatype are not valid for  
the array in java (Javascript & python supports  
such stuff)

Array indexing starts from 0

~~arr~~ = [ 0 , 18 , 17 , 11 , 16 , 81 , 91 ]  
x0 x1 x2 x3 x4 x5 x6

so as array has 7 elements so indexing  
of them will start from 0 to 6.

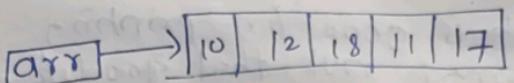
\* Creation of array in java

int Reference A [ ] = new int [ size ];  
                 ↑                                    object

In Java above is way to create arrays  
& an in java each array is an object.  
any size of array will be equal to the size  
which you specify after datatype

B. length of array can be taken by

→ arrayname.length → It will give  
you size of array



Date \_\_\_\_\_  
Page \_\_\_\_\_

also array can be declared like

```
int arr[] = { 10, 12, 18, 11, 17 };
```

\* Array element accessed easily with help of  
arr[]

→ arr[index to fetch]

for loops can be used in order to access the  
array

It is properly  
not a method

```
for (int i=0; i<arr.length; i++) {
```

Code to  
print  
elements  
inside  
array.

```
    System.out.println(A[i]);
```

\* In order to access the array for each loop  
finds very useful

int arr[] → for each x in arr

```
for (int x: arr) {
```

System.out.println(x); → x will fetch

elements inside array

}

One by one & we will print element with

help of loop (for each loop)

→ Such like that for each loop can be used inside  
array.

→ for each loop only in one directional form  
start to end of array.

for each is good for printing array element but that good for using ~~for~~ <sup>Date \_\_\_\_\_</sup> <sup>Page \_\_\_\_\_</sup> ~~foreach~~

loop to take input an all

#Array has type means array can be 1D, 2D or 3D type array

also

int[] a;  $\Rightarrow$  int a[];

② int a = new int[5];

so it will generate array like

[a]  $\rightarrow$  [0 0 0 0 0]

#Increase size of array

to increase the size of arr

int arr = {10, 20, 30, 40};

int z = new int[2\*arr.length];

for (int i=0; i < arr.length; i++) {

    z[i] = arr[i];

} // Changing the reference

new variable z = arr; arr = z;

Array always get created inside  
heap

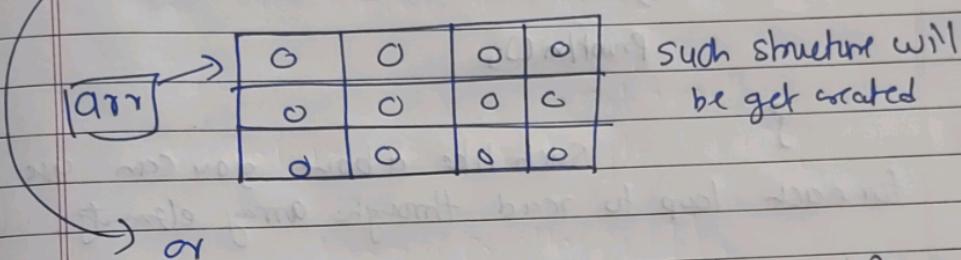
Data  
Page

## # 2D array

It will create a matrix like structure where you are able to store the elements or values inside it.

How can you initialise them

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



or

```
int arr[][] = {{1,2,3},{4,5,6},{7,8,9}};
```

Such how you can create 2D arrays and size will be automatically determine size.

You have use 2D array to get / modify value in 2D array

```
for (int i=0; i<arr.length; i++) {
```

```
    for (int j=0; j<arr[0].length; j++) {
```

```
        S.O.P(arr[i][j]);
```

Such how you print value inside 2D array } so p(kp); + new line)

Jagged array is easy & important as it is useful in many cases

also you can use for each loop

```
for (int x[] : arr)
```

```
for (int a : x)
```

```
s.o.p(a)
```

[A][E] for i = 0 to n - 1  
} for j = 0 to m - 1

for (int a : x)  
Println(a);

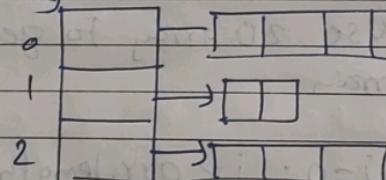
}

Such like above you can use  
for each loop to read through array elements

# so It is not case like you have to  
creat non arr

(# create Jagged array)

[A]



useful  
for graph  
or also useful  
in hash map  
indexing

so how to create such struc

so

{ first create a reference bl

keep  
it by

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

int

arr[0] = new int [4];

arr[1] = new int [2];

arr[2] = new int [3];



to fetch or read such array the for each or  
for loop is good

# Sorting array in Java

Java.util.Arrays.sort( arrayName );



such how you can sort a array