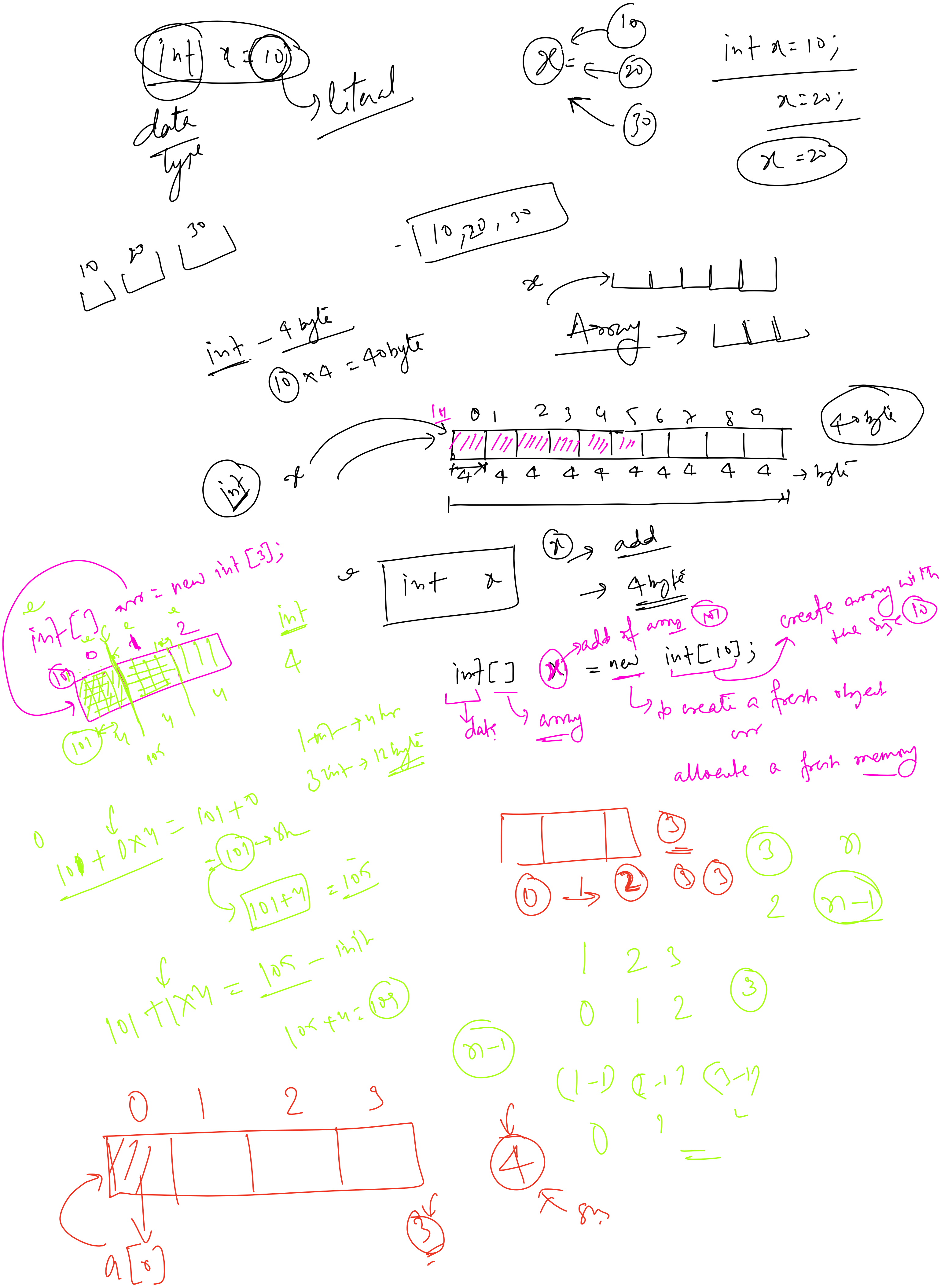
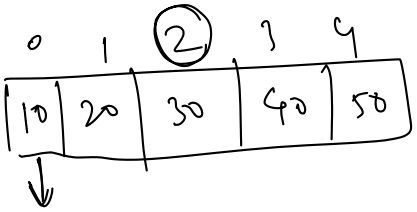


int x = 10
data Type
literal
int x=10;
x=20;
x =20
10 20 30
10, 20, 30
Array
int - 4 byte
10 x4 = 40byte
0 1 2 3 4 5 6 7 8 9
40 byte
4 4 4 4 4 4 4 4 4 4 → byte
int x
x → add
→ 4 byte
int[] arr = new int[3];
int[] x = new int[10];
add of array 101
create array with the size 10
data
array
to create a fresh object
or
allocate a fresh memory
1 int → 4 byte
3 int → 12 byte
101 + 0x4 = 101+0
101+4 = 105
101 + 1x4 = 105 - 1th
105+4 = 109
n-1
1 2 3
0 1 2
0 1 2 3
a[0]

index



① int[] x = new int[5];

Size = 5

0, 1, 2, 3, 4 → index

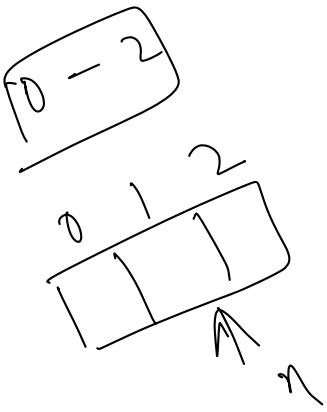
① fetch, update
↳ index

• x[2]; → output will be value stored at the index 2.

① x to (n-1) 0 to (n-1)

-ve

① <0 / >n-1 // ArrayIndexOutOfBound Exception



x[3] → runtime

x[-1] =

① A[] = {1, 2, 3, 4, 5, 6};
print all elements of this array.

① access / fetch

eg | int[] arr = {1, 2, 3, 4, 5, 6}; // Syntax
only at the time of declaration.

→ int[] arr = new int[6]; // Error

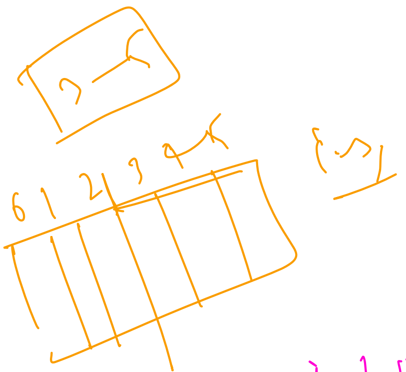
→ arr = {1,2,3,4,5,6}; X

arr[0] = 1;

arr[1] = 2;

⋮

arr[5] = 6;



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

// 6-1 = 5

index 0 — 5   i <= 5   i < 6

```
for (int i=0; i<6; i++) { // index
    int v = arr[i];
    Sop(v); // 2
}
```

i = 0 → 1 → ... 5

① arr[0]

② arr[1]

Size of array → arr.length (8)

Arrays.toString(arr)

Arrays.toString(arr);

Sop(Arrays.toString(arr));

Arrays.toString(arr)

Sop(Arrays.deepToString(arr));

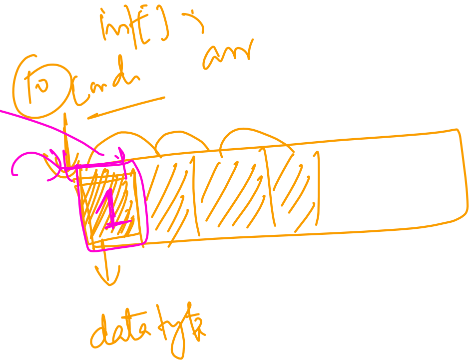
```
for(int i=0; i<arr.length; i++){
    sop(arr[i]);
}
```

arr.length

.length()

.length

for-each // Iterator




```
for(int x : arr){
    sop(x);
}
```

add

index

int

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

int

4

```
for(int x : arr){   // 4 times
}
```

1
2
3
4

```
int[] arr = {19, 21, 30, 31, 35}
```

5
6

```
for(int i : arr){
    sop(i);   // 19, 21 30
}
```

i = 19 21 30

index

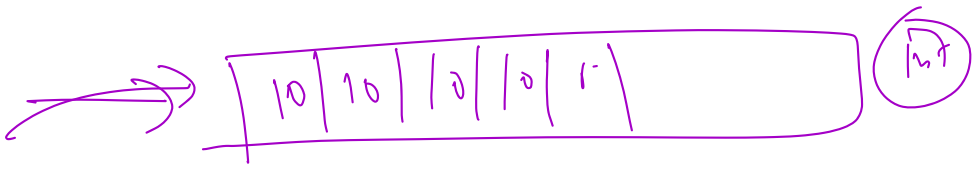
content

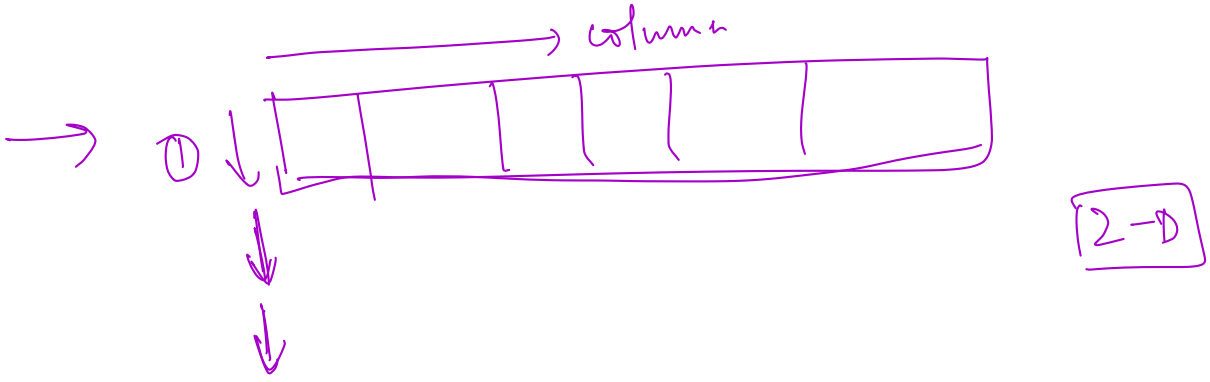
index

Array Index Out of Bound

// Array 0 index

0 → 5





1D

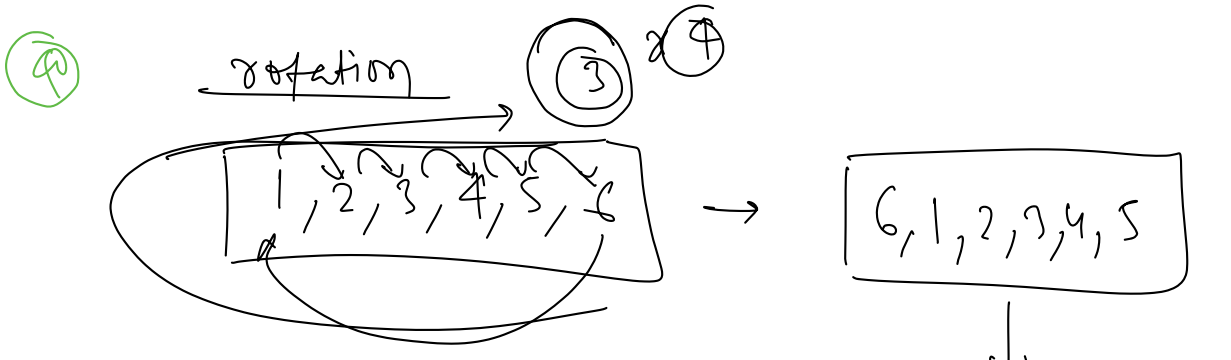
```
int[][] arr = new int[6][];
```

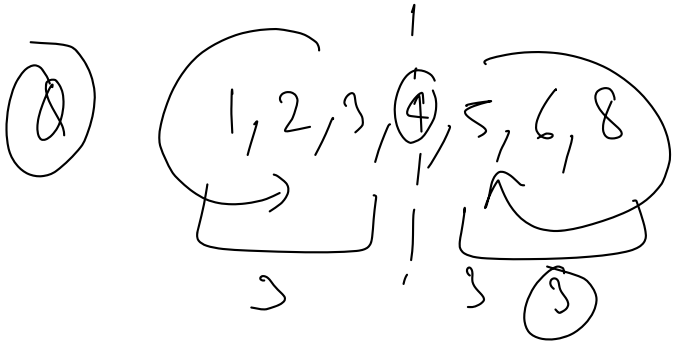
2D

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

row = 2

colm. = 3





Hash array

1 2 3 4 | 5 6 7 8

e e e

0 1 2 3 4 5 6 7 8

(0) 1 2 3 4 5 6 7 8 9

(3) - odd number

odd ~~option~~

0 1 2 3 4 5 6 7 8

1 8 3 2 5 4 7 6 9

1 6 3 8 5 2 7 4 9