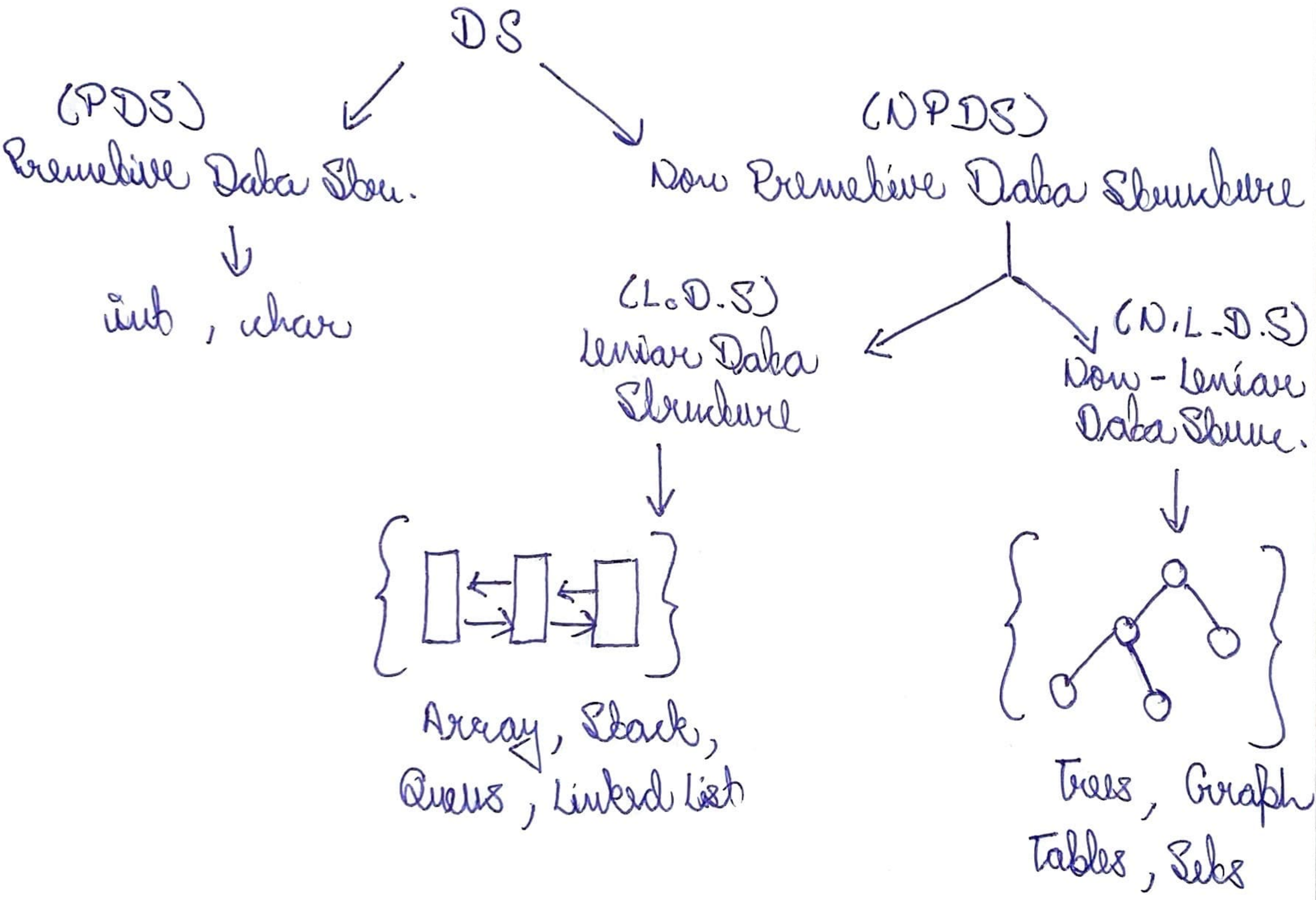
:- Arrays :-

→ Data Structure :

Store Data in an organized Manner.



DS

(PDS) Premetive Data Stru.

↓

int , char

(NPDS) Non Premetive Data Structure

(L.D.S) Leniar Data Structure

↓

Array, Stack, Queus, Linked List

(N.L.D.S) Non-Leniar Data Struc.

↓

Trees, Graph Tables, Sets

→ Arrays :

- Linear Data Structure
- Contigous fashion Data Store
  - Adjacent Memory Location.
- Homogenious
  ↳ Store same type of Data.

Ex :

int 4 bytes

Store Roll No. of 4 Students.

1	2	3	4
↓	↓	↓	↓
Address (3000)	(3004)	(3008)	(3012)

→ Declaration of Array :

int arr[]= new int [4] → Object create

- int → Data Type
- arr → Name
- [4] → Space of an Array-

arr [ ]

- arr → Name of Array
- [ ] → Sign of Array-

arr [4] =

	5	1	2	4
index →	0	1	2	3
Memory →	3000	3004	3008	3012

- Index will be mapped to Memory.

Operation 1 →
Accessing of Array

```
int el = arr [2]
Print (el);
```

Output will be 2.

→ Formula to find location in array

index -> i

Starting Memory Location -> m

Data Size -> 8.

arr[2]

-> $3000 + 2 \times 4$

$[\, m + i \times 8 \,]$

→ Where is an array store

: Heap

→ Insert array data

Pos 1 -> Store 5

$arr[] = 2, 4, 6, \overset{i}{0}, \overset{i+1}{0}$

$\quad\quad\quad\;\; 0 \;\; 1 \;\; 2 \;\; 3 \;\; 4$

$3 \rightarrow n-2 \qquad 4 \rightarrow n-1$

```
{
  arr[i+1] = arr[i]
    i--;
}
  arr[pos] = 5;
```

→ Delete an element in array

i -> n-2.

$arr[] = 2, \underset{i}{\overset{\downarrow}{4}}, \underset{i+1}{8}, \overset{0\,\downarrow}{12}$

```
arr[i] = arr[i+1];
```

→ Dynamic Array

- Size is not fixed
- But can vary.
- Can increase the size of an Array.

- Size
- Capacity :

Store max element.

S/L -> How many elements are there.

- How to increase capacity.

-> 10 - 11th

-> Old capacity + Half of old cap.

Formula. -> nc = OC + OC/2.

Can be changed on new versions of Java.

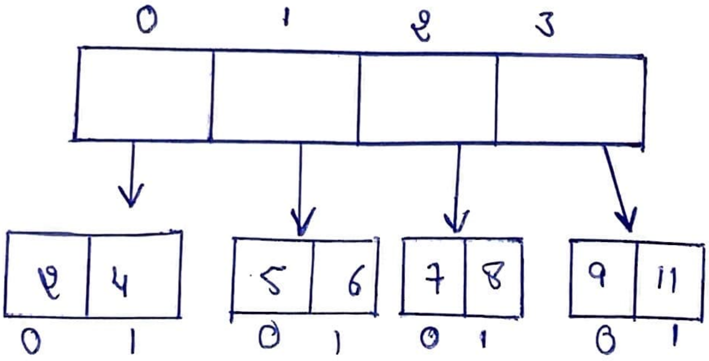
× If the memory is not free then this will throws an error called.

-> Memory Out of Space

(M.O.S.)

→ Multi Dimensional Array (MDA) - 2D Array.

2D → Array inside an array.



→ Represent Multi Dimensional Array.

2D array   int arr [ ] [ ] = new int [5] [2] ;

↓ ↓ ↓ ↓

L1 L2 L1 L2

- Level 1 array will be of 5 size
- Level 2 array will be of 2 size.