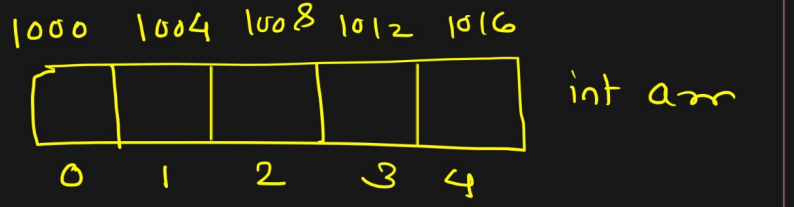
Day-16 Linked List

A) Why we need a Linked List?

Array :- Collection of elements
- it's size is fixed

```
int arr [] = new int [size]
```

size → Fixed → Size=5



Contigeous Memory Allocation
- Linear DS

12, 14, 32, 57, 71, 101

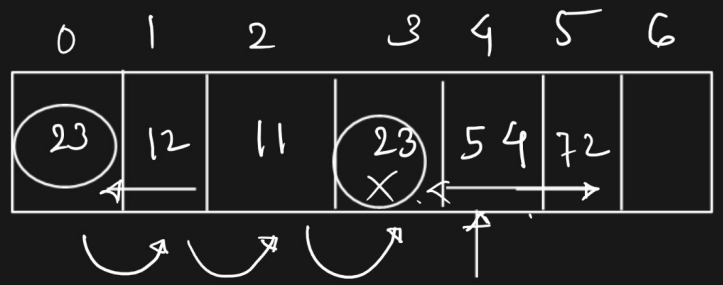
⟹ We can't insert more than the size of array

Array Drawback :- 1) size is fixed

2) Wastage of memory - chances

3) Insertion & Deletion it is very complex in array.

Size = 100 ⇓ insert only 10 ele ⇓ go ele ⇒ Wastage of memory



insert – at last ⇒ very easy

- at first
- in betn element

} very difficult

⇓

Time Consuming

Delete ⇒ any elc ⇒ shifting

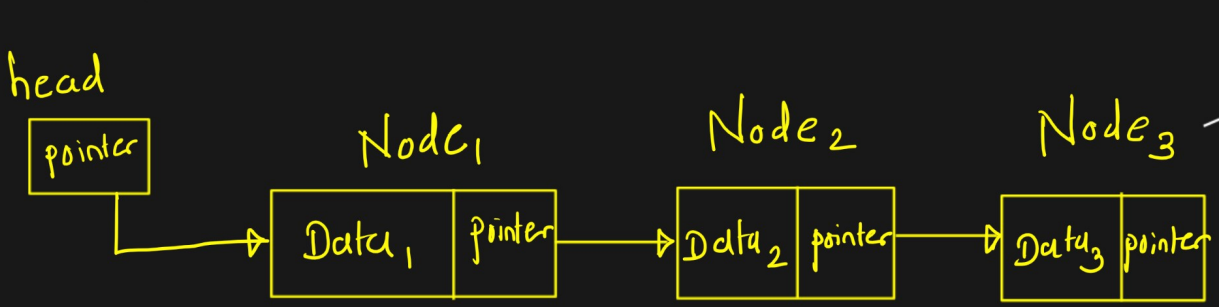
⇑

Difficult

* Linked List :-

Defn :- It is one of the Data Structure, that store the data in the form of chain of nodes.

OR :- It is a chain of nodes.



int x

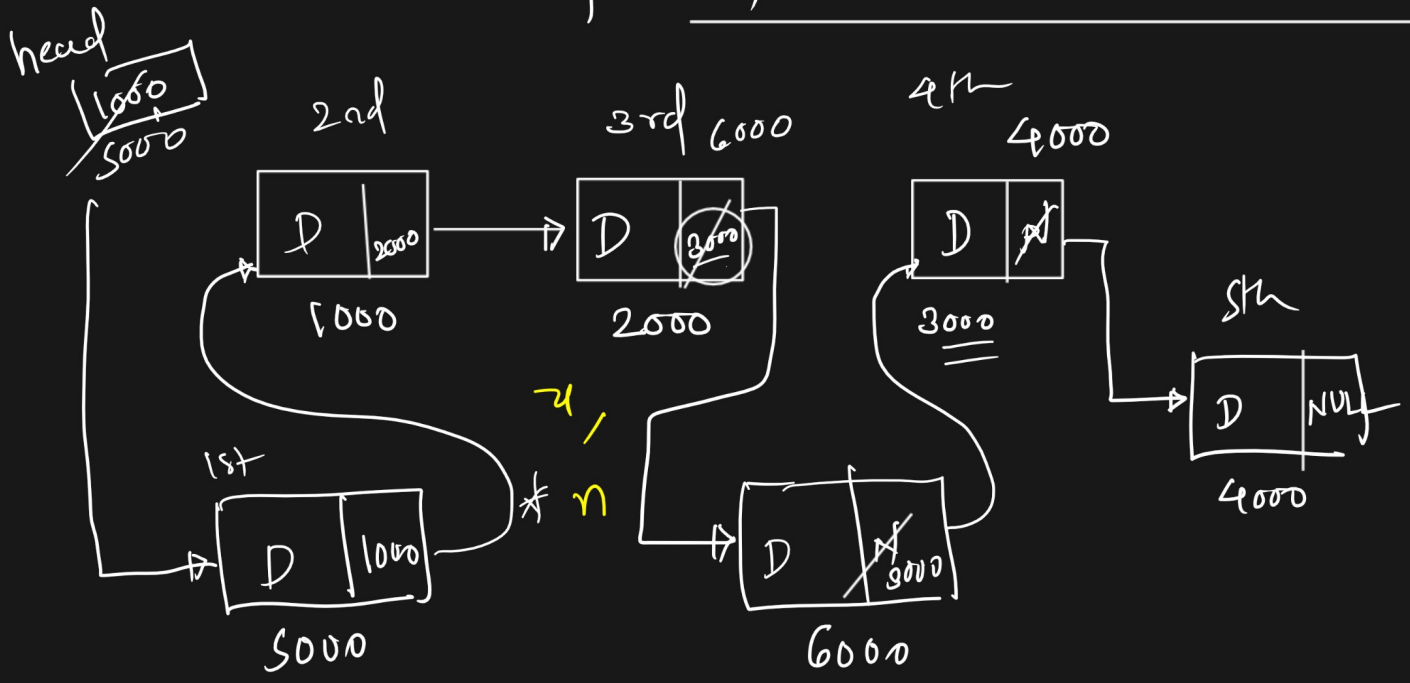
structure / class

user defined { data, fun }

Advantage:- 1) You can insert infinite no. of elements

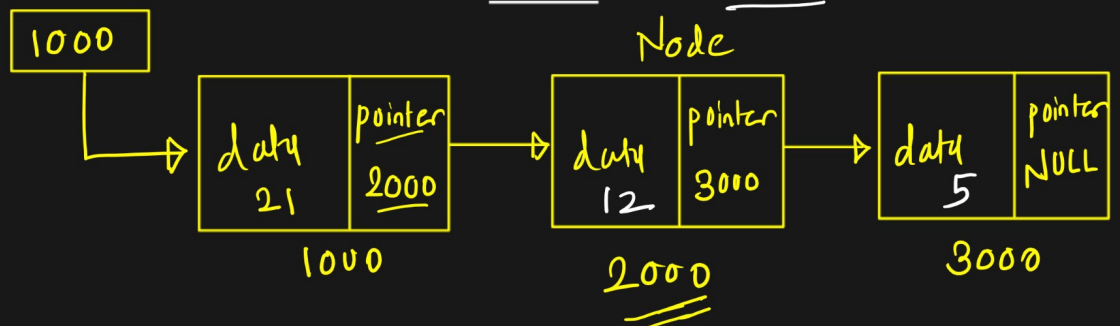
2) And memory create for node when only the new elem is coming.

3) Very easily u can insert / delete any node and fast.



head (pointer)

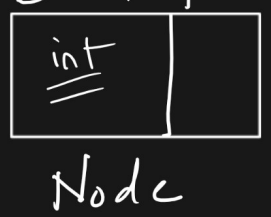
Linked List



Operation on LL

1) Insert
2) Delete
3) Display
4) Search

A) How to create an one single node ?



C / C++

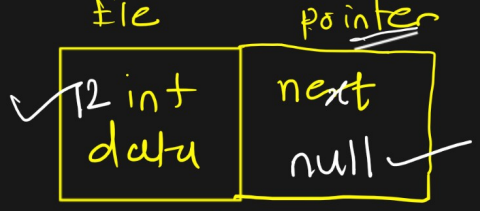
↳ structure

C++ ↳ class

JAVA / python / DS

↳ class

```
class className
{
   data members
   ==========
   ==========
   member fun? / method
}
```

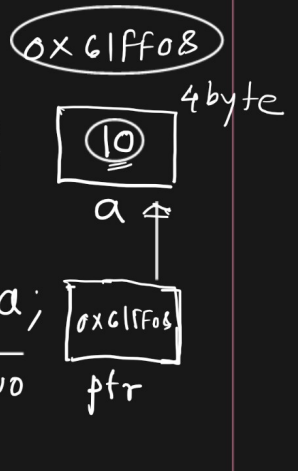


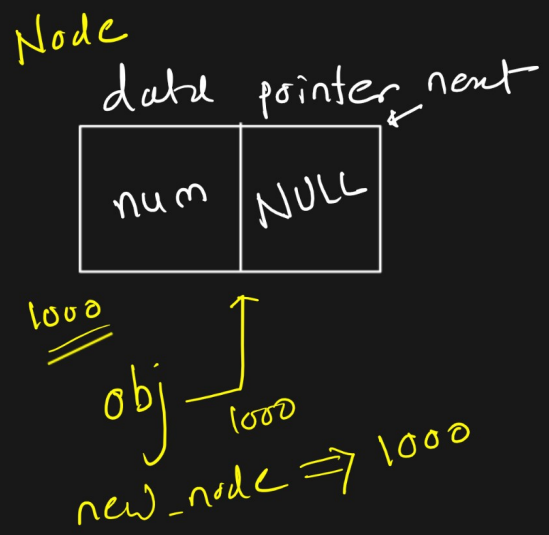

```
class Node
{
    public int data;
    public Node next;

    public Node (int num)
    {
        data = num;
        next = NULL;
    }
}
```

```
Node obj = new Node (12);
```

C & C++






```
Class Node
{
    public int data;
    public Node next;

    public Node (int num)
    {
        data = num;
    }   next = null;
}
```

```
new_node
Node obj = new Node (12)
```

1000