Your Ultimate Guide To Landing Top AI roles

DECODE
AiML

Types of Linked Lists

→ Linked Lists are linear data Structure where nodes are Connected using References (pointers)

→ Unlike arrays, the memory is not contigous.

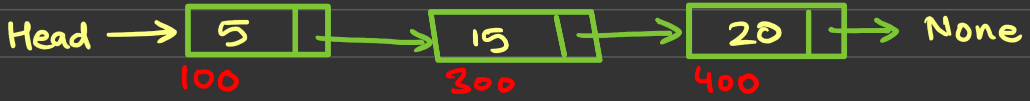
→ Types of Linked Lists

① Singly Linked Lists

→ Each node stores:
① data
② References to the next node

```
Class Node:
    def __init__(self,data):
        Self.data = data
        self. next = None
```

→ only forward direction traversal possible

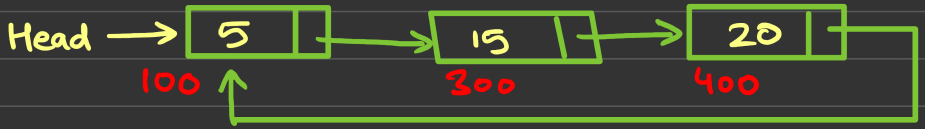


② Circular Linked Lists

→ Each node stores:
① data
② References to the next node

```
Class Node:
    def __init__(self,data):
        Self.data = data
        Self.next = None
```

→ Last Node points back to the first Node.



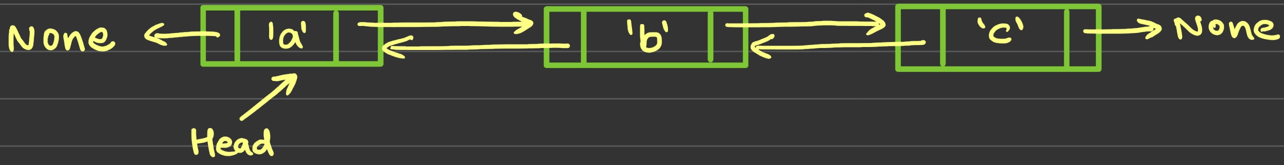
③ Doubly Linked Lists

→ Each node stores:
① data
② References to the next node
③ References to the prev node

```
Class Node:
    def __init__(self,data):
        Self.data = data
        Self. next = None
        Self. prev = None
```

→ Both forward and backward traversal.

→ Extra memory needed for previous pointer.



④ Circular doubly Linked List

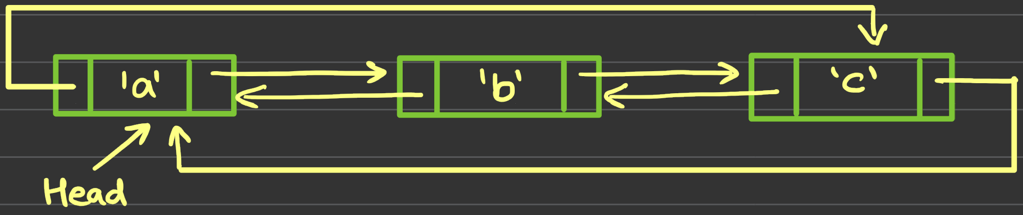
→ Each node stores:
① data
② References to the next node
③ References to the prev node

```
Class Node:
    def __init__(Self,data):
        Self.data = data
        self. next = None
        Self. prev = None
```

→ Combination of doubly and circular Linked Lists.

→ Last node next → First Node

→ First node Prev → Last Node



Like ▶ Subscribe