

### 3주차: 단순 연결 리스트 (Singly Linked List)

핵심 개념: 데이터를 담은 '노드'를 기차처럼 연결하여 관리하는 가장 기초적인 비선형 구조

노드(Node)의 구성:

data: 실제 저장할 값.

next: 다음 노드의 주소값을 저장하는 참조(화살표).

주요 포인트:

Head: 리스트의 시작점을 가리키는 포인터. 이걸 잃어버리면 리스트 전체를 잃어버림.

동적 할당: 배열과 달리 실행 중에 크기를 자유롭게 늘리거나 줄일 수 있음.

시간 복잡도:

데이터 검색/맨 끝에 추가:  $O(n)$ (처음부터 끝까지 찾아가야 함)

맨 앞에 추가:  $O(1)$

```
class Node {  
    int data;  
    Node next;  
    Node(int data) { this.data = data; this.next = null; }  
}
```

```
public class C01 {  
    public static void main(String[] args) {  
        // 첫 번째 칸(head) 생성  
        Node head = new Node(10);
```

```
// 두 번째 칸 생성 후 연결  
head.next = new Node(20);  
  
// 출력: head부터 다음 칸이 없을 때까지 이동하며 출력  
Node curr = head;  
while (curr != null) {  
    System.out.print(curr.data + " -> ");  
    curr = curr.next;  
}  
System.out.println("null");  
}  
}
```