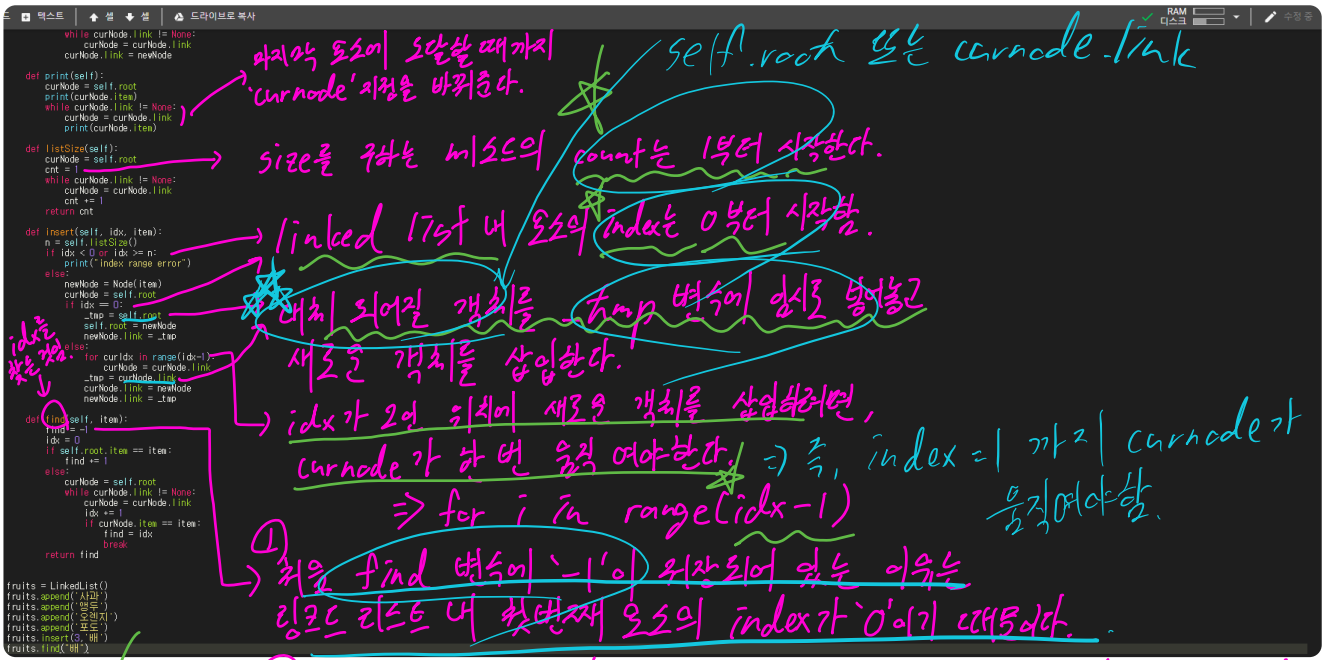

마지막 노드에 도달할 때까지 'curNode' 지정을 바꿔줌.

self.root 또는 curnode.link

size를 구하는 메소드의 count는 1부터 시작한다.

linked list 내 요소의 index는 0부터 시작함.

새로 들어올 객체를 _tmp 변수에 일시 보관하고 새로운 객체를 삽입한다.

idx가 2의 의미 새로운 객체를 삽입하려면, curnode가 한 번 움직여야한다. => 즉, index-1 까지 curnode가 움직여야함.

=> for i in range(idx-1)

idx는 찾을 값

① 최종 find 변수에 '-1'이 저장되어 있는 이유는 링크드 리스트 내 첫번째 노드의 index가 '0'이기 때문이다.

② item 매개변수에 해당 링크드 리스트 내에 존재하지 않는 객체가 들어온다면, '-1'을 return 하기 위해서.

curnode 지점을 계속 바꿔줘야할 때, curnode 변수를 사용한다. => 이런 수행이 필요없을 때는, curnode를 지정할 필요가 없다.

```
def delete(self,item):
    delYN = False
    curNode = self.root
    preNode = curNode
    if curNode.item == item:
        self.root = self.root.link
        delYN = True
    else:
        while curNode.link != None:
            preNode = curNode
            curNode = curNode.link
            if curNode.item == item:
                preNode.link = curNode.link
                delYN = True
    if curNode.item == item:
        preNode.link = None
        delYN = True
    if delYN == False:
        print("delete failed")
```

boolean 객체 생성이 핵심!

preNode 변수를 사용하는 것이 핵심!

① 지울려고 하는 item이 첫번째 노드의 item일 때.

② 지울려고 하는 item이 중간에 있는 노드의 item일때.

마지막 노드까지 감.

③ 지울려고 하는 item이 마지막 노드에 있는 item일때

curnode 움직이는 것이 break 된적이없다.

※ if와 elif 뒤에 나오는 조건은 서로 달라야 한다.

↑ 이전의 조건이 아닌, 또 다른 조건이라면 ...!

※ 파이썬 프로그램은 코드들을 위→아래!

순서로 실행한다!!!