# Linked List

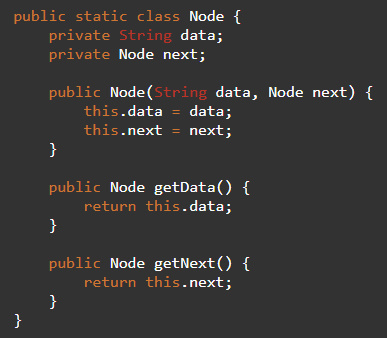
## Problem Statement 1:

Find if a list is circular.

### Description:

Given the head of a singly linked list, find if the linked list is circular or not. A linked list is circular if it is not NULL-terminated and nodes connected are in the form of a cycle.

Node class: Example:



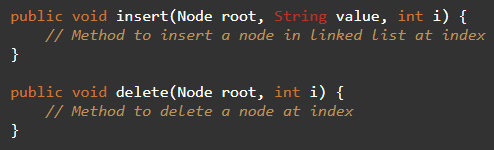
## Problem Statement 2:

Insertion/deletion in the doubly linked list.

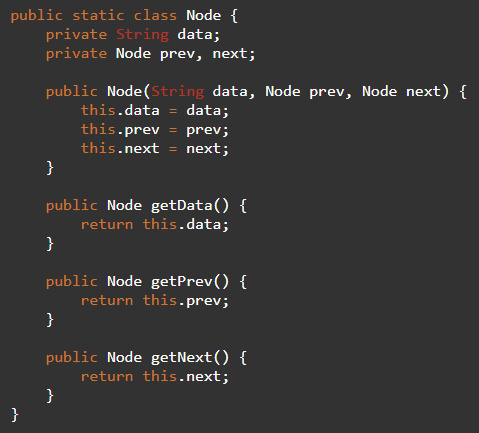
### Description:

Given the head of a doubly linked list and respective parameters, implement methods to insert/delete the node at the provided index.

Methods to implement:



Node class:



## Problem Statement 3:

Merge 2 sorted LinkedLists.

### Description:

Given the head of two singly linked lists both in sorted increasing order, write a function which returns the head of a new single linked list, created from the nodes from the 2 lists with values in sorted order.

## Problem Statement 4:

Reversing a doubly LinkedList.

### Description:

Given the head of singly/doubly linked list, write a function that returns the head of a new linked list created from the same nodes but reverse ordered w.r.t the original list.