**Detecting the First Node of a Loop in a Linked List**

class Node {

  constructor(data) {

    this.data = data;

    this.next = null;

  }

}

class Solution {

  detectFirstNodeLoop(head) {

    let tortoise = head;

    let hare = head

    while(hare!==null && tortoise.next !== null){

        tortoise = tortoise.next;

        hare = hare.next.next

        if(tortoise === hare){

           break

        }

    }

    if(hare === null)  return 'No Loop Found'

    let newTortoise = head

    while(hare!==newTortoise){

        newTortoise = newTortoise.next

        hare = hare.next

      }

    return "Loop Found At ===>" + newTortoise.data

  }

}

let head = new Node(1);

let second = new Node(2);

let third = new Node(3);

let fourth = new Node(4);

head.next = second;

second.next = third;

third.next = fourth;

fourth.next = second;

let solution = new Solution();

console.log(solution.detectFirstNodeLoop(head));