Yash Gunjal CSI Assignment Week 2

Implement a Linked List in Python Using OOP and Delete the Nth Node

Code:-

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


class LinkedList:
    def __init__(self):
        self.head = None

    def add_node(self, data):
        """Adds a node with the given data at the end of the list"""
        new_node = Node(data)
        if not self.head:
            self.head = new_node
            return
        curr = self.head
        while curr.next:
            curr = curr.next
        curr.next = new_node

    def print_list(self):
        """Prints the entire linked list"""
        if not self.head:
            print("List is empty.")
            return
        curr = self.head
        while curr:
            print(curr.data, end=" -> ")
            curr = curr.next
        print("None")

    def delete_nth_node(self, n):
        """Deletes the nth node (1-based index)"""
        if not self.head:
            raise IndexError("Cannot delete from an empty list.")

        if n <= 0:
            raise IndexError("Index must be a positive integer.")

        if n == 1:
            self.head = self.head.next
```

```
            return

        curr = self.head
        count = 1
        prev = None

        while curr and count < n:
            prev = curr
            curr = curr.next
            count += 1

        if not curr:
            raise IndexError("Index out of range.")

        prev.next = curr.next


# 🚀 Testing the implementation
if __name__ == "__main__":
    ll = LinkedList()
    print("Adding nodes: 10, 20, 30, 40, 50")
    for value in [10, 20, 30, 40, 50]:
        ll.add_node(value)

    print("\nOriginal List:")
    ll.print_list()

    try:
        print("\nDeleting 3rd node...")
        ll.delete_nth_node(3)
        ll.print_list()

        print("\nDeleting 1st node...")
        ll.delete_nth_node(1)
        ll.print_list()

        print("\nTrying to delete 10th node (out of range)...")
        ll.delete_nth_node(10)

    except IndexError as e:
        print(f"Exception: {e}")
```

output:-

```
Adding nodes: 10, 20, 30, 40, 50

Original List:
10 -> 20 -> 30 -> 40 -> 50 -> None
```

Deleting 3rd node...
10 -> 20 -> 40 -> 50 -> None

Deleting 1st node...
20 -> 40 -> 50 -> None

Trying to delete 10th node (out of range)...
Exception: Index out of range.