**Explanation of the Code**

1️-:Insert at the Beginning

Creates a new node.

If the list is empty, it points to itself.

Otherwise:

Traverses to the last node.

Adjusts last->next to point to the new head.

2️-:nsert at the End

Creates a new node.

Traverses to the last node.

Attaches new node after the last node and links it to head.

3️-:Insert at Nth Position

If n == 1, insert at the beginning.

Otherwise, traverse to (N-1)th node and insert.

If n is out of bounds, display an error.

4️-:Insert at the Center

Uses the slow & fast pointer technique to find the middle.

Inserts new node after the middle node.

5️-: Display Forward

Starts from head and traverses the list using do-while to ensure at least one iteration.

Stops when temp == head again.

6️-: Display Reverse

Reverse traversal is tricky in a singly circular linked list since there's no prev pointer.

Uses two loops:

First loop finds the last node.

Second loop manually traces back to previous nodes.

