Credit Name:CSE3110
Assignment Name:STackList

How has your program changed from planning to coding to now? Please explain?

Step 1: Define the StackList Class
- Declare the StackList class, which represents a stack implemented using a linked list.
- Initialize the head node as null in the constructor.

Step 2: Implement the push Method
- Create a new Node object with the provided string and set its next node to the current head.
- Update the head to point to the new node, effectively pushing the new item onto the stack.

Step 3: Implement the pop Method
- Update the head to point to the next node, effectively removing the top item from the stack.

Step 4: Implement the getTop Method
- Return the data stored in the head node, representing the top item of the stack.

Step 5: Implement the toString Method
- Traverse the linked list starting from the head and build a string representation of the stack.
- If the head is not null, initialize listString with the first item's data.
- Iterate through the remaining items, appending their data to listString.
- Return the built string representation of the stack.

Step 6: Implement the getSize Method
- Traverse the linked list starting from the head and count the number of items in the stack.
- Return a string indicating the count of items in the stack.

Step 7: Implement the makeEmpty Method
- Set the head to null, effectively emptying the stack.

Step 8: Define the StackedListTester Class
- Declare the StackedListTester class, which contains the main method.

Step 9: Create a StackList Object and Perform Operations
- Create a StackList object named stack.
- Push three items ("Blue", "Green", and "Yellow") onto the stack using stack.push("...").
- Print the stack using System.out.println(stack.toString()).
- Get and print the top item of the stack using System.out.println("The top item in the stack is: " + stack.getTop()).
- Remove the top item from the stack using stack.pop().
- Print the updated stack using System.out.println(stack.toString()).
- Remove all items from the stack using stack.makeEmpty().
- Print the empty stack using System.out.println(stack.toString()).
- Push a new item ("Red") onto the stack using stack.push("...").

- Print the updated stack using System.out.println(stack.toString()).