Credit Name: Chapter 13
Assignment Name: Queue List

Name: Grayson Ardron

Reflection log

Firstly I copied and pasted the linked list over from class demo, Changing needed strings to Objects for the task in need of being performed. I also removed the AddAtEnd function and simplified remove for the task at hand.

```
package StackList;


public class LinkedList
{
        private Node head;

        public LinkedList()
        {
                head = null;
        }

        public void addAtFront(Object str)
        {
                Node newNode = new Node(str);
                newNode.setNext(head);
                head = newNode;
        }



        public Object remove()
        {
                Node current = head;
                head = current.getNext();


                return(current.getData());
                        }
```

I then finally added getHead to get the data on the top of the list.

```
public String toString()
{
        Node current = head;
        String listString;

        if (current != null) {
                listString = current.getData() + "\n";
                while (current.getNext() != null) {
                        current = current.getNext();
                        listString += current.getData() + "\n";
                }
                return(listString);
        } else {
                return("There are no items in list.");
        }
}

public int size()
{
        Node current = head;
        int count = 0;

        if (current != null) {
                count += 1;
                while (current.getNext() != null) {
                        current = current.getNext();
                        count += 1;
                }
                return(count);
        } else {
                return(0);
        }
}

public void makeEmpty()
{
        head = null;
}


public Object getHead()
{
        return head.getData();
}
```

I copied Node from class demo

```java
package StackList;

public class Node
{
	private Object data;
	private Node next;


	public Node(Object newData) {
		data = newData;
		next = null;
	}


	public Node getNext() {
		return(next);
	}



	public void setNext(Node newNode) {
		next = newNode;
	}


	public Object getData() {
		return(data);
	}


}
```

For StackList I copied the code from class demo and added/removed when needed

```
package StackList;

public class StackList
{
	private LinkedList data;
	private int top;

	public StackList()
	{
		data = new LinkedList();
		top = -1;

	}

	public Object top()
	{
		return(data.getHead());
	}

	public Object pop()
	{

		return(data.remove());
	}

	public void push(Object item)
	{
		data.addAtFront(item);
	}

	public boolean isEmpty()
	{
		if(data.size() == 0){
			return(true);

		} else {

			return(false);  }

	}

	public int size()
	{
		return data.size();

	}

	public void makeEmpty()
	{
		data.makeEmpty();
	}

}
```

And lastly copied the Stack test code from class demo for testing

```
package StackList;

public class StackListTest
{

        public static void main(String[] args)
        {
                StackList s2 = new StackList();

                System.out.println("Adding \"red\" and \"yellow \" to stack. ");
                s2.push("red");
                s2.push("yellow");
                System.out.print("Top of stack: "+ s2.top() + "\n");
                System.out.print("Items in stack: "+ s2.size() + "\n");
                System.out.println("Removing top item.");
                s2.pop();
                System.out.print("Top of stack: "+ s2.top());
                System.out.print("Items in stack: "+ s2.size());



        }

}
```