Hands-On Activity 2

1. Using the singly-linked class definition in course slides (page 6), **write a method** that finds the sum of all data on the linked list.

```
public void sum(Node head) {

	while (head!=null && head.next != null) {

		this.sum += head.data;
		head = head.next;
	}

	System.out.println(this.sum);
}
```

2. Using the singly-linked class definition in course slides (page 6), **write a Java method** that swaps the reference to the head of the list with the second node (i.e. the node next to the header). Do not swap the nodes' data, but the references to the nodes.

```
public void swap(Node head) {

	Node temp = head.next;
	head.next = temp.next;
	temp.next = head;

}
```

3. Using the doubly-linked list class definition in course slides (page 15), write the following methods:

```
void remove( Node n);
Node search( int k);
Node removeLast();
```

```
public void remove(Node n) {
      n.prev.next = n.next;
      n.next.prev = n.prev;
}

public Node search(int k) {

      while (head != null) {
            if (head.data == k)
                  return head;
            else {
                  head = head.next;
            }
      }

      return null;
}

public Node removeLast() {
      if (head == null)
            return null;

      while (head.next != null) {
            head = head.next;
      }
      head.prev.next = null;
      head.prev = null;

      return head;
}
```