Task 01 Explanation :

The first thing I did is to is to find the index of Dammam then I add an if condition to check if "Dammam" is not in the list or if the list is empty. And if not, I will add Tabouk in the index of Dammam and for the AlKhafj I found the position of the last index isby taking the sizeOfTheArray -1, then I added the new city in the sizeOfTheArray -2.

Task 01 code :

```
    int dammamIndex = stringList.indexOf("Dammam");

    if (dammamIndex == -1 || stringList.isEmpty()) { System.out.println("Dammam is not in the list or the list is empty.");}

 else {

        stringList.add(dammamIndex, "Tabouk");

        System.out.println("List after adding Tabouk: " + stringList);}

===

    int size = stringList.size();

    if (stringList.isEmpty()) { System.out.println("The list is empty.");}

 else {

        stringList.set(size - 2 ,"AlKhafj"); // -2 because  the position of the last index is sizeOfArray -1

        System.out.println("List after adding AlKhafj: " + stringList);  }

===
```

Task 01 output :

```
[Running] cd "c:\Users\seabd\OneDrive - KFUPM\Desktop\ICS-202\" && javac LinkedListDriver.java && java
LinkedListDriver
List: [Madinah, Dammam, Riyadh, Dhahraan]
List: [Najraan, Madinah, Dammam, Riyadh, Jubail, Dhahraan]
Removed Element: Najraan
Updated list: [Taif, Madinah, Dammam, Riyadh, Jubail, Abha]
List after adding Tabouk: [Taif, Madinah, Tabouk, Dammam, Riyadh, Jubail, Abha]
List after adding AlKhafj: [Taif, Madinah, Tabouk, Dammam, Riyadh, AlKhafj, Abha]
```

Task 02 Explanation :

The first thing I did is to check if the list is empty and throws an exception if it is true and then I make sure that the index is ok then I created a new node for 'newElem' then I Insert the new node at the specified index, then if the insertion is at the end I update the tail, and for the second method most of the things are as same as the first method so no need for explanation and for the third method the first thing I did is the validations then I search for the second occurrence of e2 and its index then I Inserted e1 after the second occurrence and if there is no if there's no second occurrence of e2 an exception will be raised .

Some of Task 02 code :

```
 SLLNode<T> pred = head;
        for (int i = 0; i < index - 1; i++) {
           pred = pred.next;
        }

        newNode.next = pred.next;
        pred.next = newNode;
```

==

```
  SLLNode<T> pred = head;
        for (int i = 0; i < index - 1; i++) {
```

```
            pred = pred.next;
        }

        deletedElement = pred.next.info;
        pred.next = pred.next.next;

==
SLLNode<T> current = head;
    int count = 0;
    int secondOccurrenceIndex = -1;

    while (current != null) {
        if (current.info.equals(e2)) {
            count++;
            if (count == 2) {
                secondOccurrenceIndex = count;
                break; // Found the second occurrence, exit the loop
            }
        }
        current = current.next;
    }

    if (secondOccurrenceIndex == 2) {
        // Create a new node with data e1
        SLLNode<T> newNode = new SLLNode<>(e1);
```

```
        // Insert the new node after the second occurrence of e2
        current = head;
        count = 0;
        while (current != null) {
            if (current.info.equals(e2)) {
                count++;
                if (count == 2) {
                    newNode.next = current.next;
                    current.next = newNode;
                    break;
                }
            }
            current = current.next;
        }
```

Task 02 output :

There is no output

Task 03 Explanation :

This is a test class for the previous task so there is no need for an explanation.

Task 03 code for one scenario :

```
public class SLL_Driver {
    public static void main(String[] args) {
```

```
        // Create a new singly linked list and populate it with elements
        SLL<Integer> myList = new SLL<>();
        myList.addToTail(7);
        myList.addToTail(5);
        myList.addToTail(3);
        myList.addToTail(50);
        myList.addToTail(7);
        myList.addToTail(9);

        // Print the original list
        System.out.println("Original Integer array: " + myList);

        // Test insertBefore method
        myList.insertBefore(4, 20);
        System.out.println("After inserting 20 before index 4: " + myList);

        // Test delete method
        int deletedElement = myList.delete(4);
        System.out.println("Element deleted from index 4: " + deletedElement);
        System.out.println("After deleting element from index 4: " + myList);

        // Test insertAfterSecondOccurrence method
        myList.insertAfterSecondOccurrence(30, 7);
        System.out.println("After inserting 30 after the second occurrence of 7: " + myList);
    }
}
```

Task 03 output :

```
ver'
Original Integer array: [ 7 5 3 50 7 9 ]
After inserting 20 before index 4: [ 7 5 3 50 20 7 9 ]
Element deleted from index 4: 20
After deleting element from index 4: [ 7 5 3 50 7 9 ]
After inserting 30 after the second occurrence of 7: [ 7 5 3 50 7 30 9 ]
PS C:\Users\seabd\OneDrive - KFUPM\Desktop\ICS-202>
```

```
\ICS-202>  c:; cd 'c:\Users\seabd\OneDrive - KFUPM\Desktop\ICS-202'; & 'C:\Program Files\Java\jdk-19\bin\j
 '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\seabd\AppData\Roaming\Code\User\workspaceStorag
959416b018599a465b60a2f2ea\redhat.java\jdt_ws\ICS-202_4e705663\bin' 'SLL_Driver'
Original String list: [ Apple Banana Cherry Banana Date ]
After inserting 'Grape' before index 2: [ Apple Banana Grape Cherry Banana Date ]
Element deleted from index 3: Cherry
After deleting element from index 3: [ Apple Banana Grape Banana Date ]
After inserting 'Lemon' after the second occurrence of 'Banana': [ Apple Banana Grape Banana Lemon Date ]
PS C:\Users\seabd\OneDrive - KFUPM\Desktop\ICS-202>
```

```
705663\bin' 'SLL_Driver'
Original Mixed list: [ 10 Hello 3.14 World 42 ]
After inserting 'Goodbye' before index 2: [ 10 Hello Goodbye 3.14 World 42 ]
Element deleted from index 1: Hello
After deleting element from index 1: [ 10 Goodbye 3.14 World 42 ]
Exception in thread "main" java.lang.IllegalArgumentException: There is no second occurrence of Hello
        at SLL.insertAfterSecondOccurrence(SLL.java:234)
        at SLL_Driver.main(SLL_Driver.java:81)
PS C:\Users\seabd\OneDrive - KFUPM\Desktop\ICS-202>
```