Exercise 6

Hariesh R - 23110344

Aim:

1. The program retrieves and displays the character at a given index in a string. It prompts the user for the string and index, checks if the index is valid, and then prints the character at that position. If the index is invalid, it notifies the user.

2. The program is to determine if two given strings are *k-anagrams* of each other. Two strings are considered k-anagrams if they have the same length and can be transformed into anagrams by changing at most k characters in one of the strings. The program should evaluate these conditions and return whether the strings meet the criteria for being k-anagrams.

3. The program is to demonstrate the use of the LinkedList class from the Collections Framework by performing insert, delete, and display operations. The program should showcase how to add elements to the linked list, remove elements from it, and print the current state of the list.

Algorithm:

1) Q1:
    a) Create a Scanner for input.
    b) Read a string from the user.
    c) Read an index from the user.
    d) Check if the index is valid for the string:
        i) Print "Invalid Index" if out of bounds.
        ii) Print the character at the index if valid.
    e) Close the Scanner.

2) Q2:
    a) Define invalid_char.
    b) Check Lengths: Return false if word1 and word2 differ in length.
    c) Match and Mark: Replace matching characters with invalid_char in both words.
    d) Count Unmatched: Count characters in word1 that are not invalid_char.
    e) Return Result: Return true if unmatched count ≤ k, otherwise false.

f) Read Inputs: Get word1, word2, and k from the user.
g) Print Result: Output the result of kAnagrams().
h) Close Scanner.

3) Q3:

a) Initialize: Create a LinkedList and a Scanner.
b) Menu Loop: Continuously display the menu and read user choices.
c) Handle Choices:
d) Choice 1: Add user input to the list.
e) Choice 2: Remove and print the first item from the list.
f) Choice 3: Print all elements in the list.
g) Choice 4: Print the list in its current state.
h) Choice 5: Exit the program and close the scanner.
i) Print Menu: Define and display the menu options.
j) Exit: End the loop and close resources when the user selects exit.

Source Code:

Q1)

```
package Exercise6;

import java.util.Scanner;


public class Q1 {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the string: ");

        String word = scanner.nextLine();


        System.out.print("Enter the index: ");

        int index = scanner.nextInt();

        scanner.nextLine();


        if(index >= word.length() || index < 0)

            System.out.println("Invalid Index");
        else
            System.out.println("Character at index " + index + " is " + word.charAt(index));

        scanner.close();
    }
}
```

Q2)

```
package Exercise6;
import java.util.Scanner;

public class Q2 {
    private static final char invalid_char = '*';

    public static boolean kAnagrams(String word1, String word2, int k){

        if(word1.length() != word2.length()) return false;
        for(int i = 0; i < word1.length(); i++){
            for(int j = 0; j < word2.length(); j++){

                if(word1.charAt(i) == word2.charAt(j)){

                    word1 = word1.substring(0, i) + invalid_char + word1.substring(i+1);
                    word2 = word2.substring(0, j) + invalid_char + word2.substring(j+1);
                }
            }
        }

        int count = 0;

        for(int i = 0; i < word1.length(); i++){

            if(word1.charAt(i) != invalid_char) count++;
        }

    return count <= k;
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the word 1:" );
        String word1 = scanner.nextLine();
        System.out.print("Enter the word 2: ");
        String word2 = scanner.nextLine();

        System.out.print("Enter the value of k: ");
        int k = scanner.nextInt();
        scanner.nextLine();

        System.out.println(kAnagrams(word1, word2, k));

        scanner.close();
    }
```

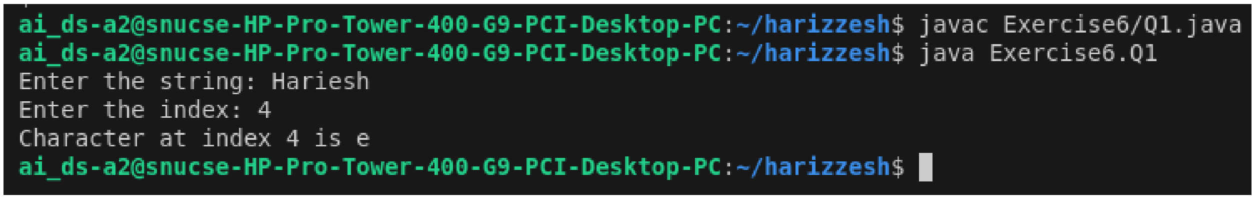
```
    }
```

Q3)

```
package Exercise6;
import java.util.LinkedList;
import java.util.Scanner;

public class Q3 {
    public static void main(String[] args) {
    LinkedList<String> list = new LinkedList<>();
    Scanner scanner = new Scanner(System.in);

    while(true){

        printMenu();
        int choice = scanner.nextInt();
        scanner.nextLine();

        switch(choice){

            case 1:
                System.out.print("Enter the value: ");
                String data = scanner.nextLine();

                list.add(data);
                break;
        case 2:
            System.out.println("Removed: " + list.remove());
        break;

        case 3:
            for(String element: list) System.out.print(element + " ");
            System.out.println();
            break;

        case 4:
            System.out.println(list);
            break;

        case 5:
            System.out.println("Exited !!");
            scanner.close();
            return;

        default:
            System.out.println("Invalid Choice");
            break;
        }
    }

}

public static void printMenu(){
        System.out.println("1. Insert");

        System.out.println("2. Remove");

        System.out.println("3. Display");

        System.out.println("4. Print");

        System.out.println("5. Exit");

        System.out.print("Enter: ");

    }

}
```

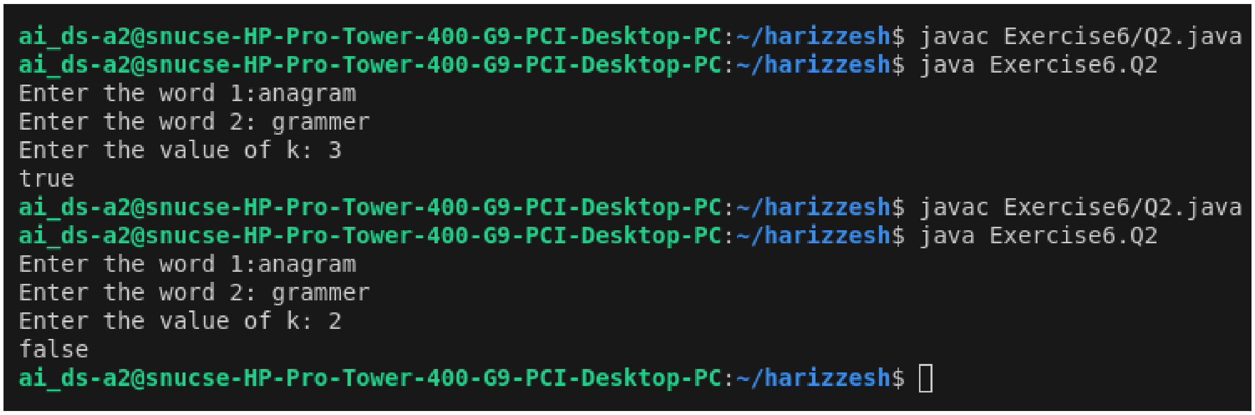
Output:

Q1)



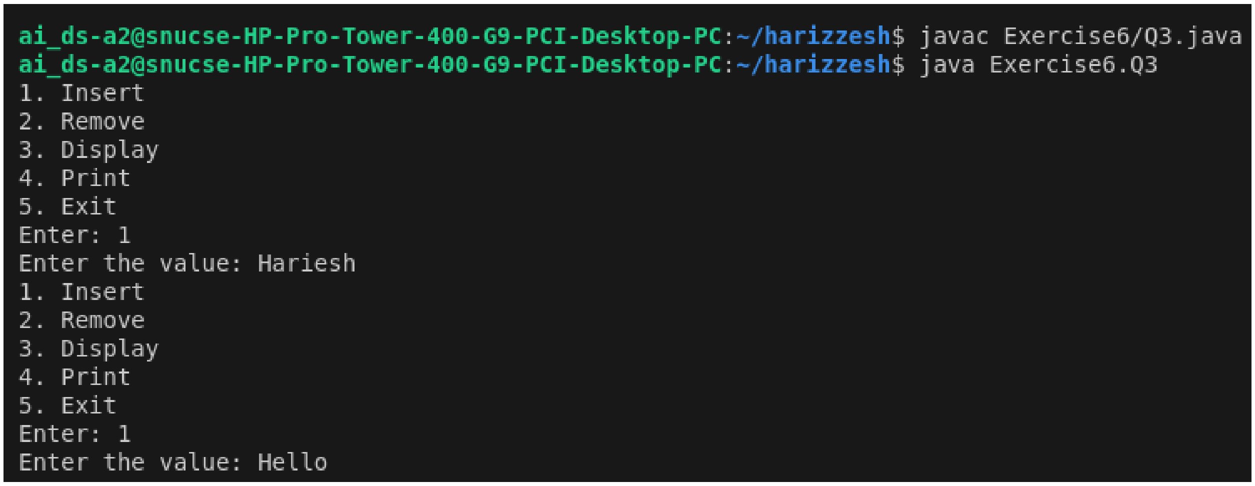
```
ai_ds-a2@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/harizzesh$ javac Exercise6/Q1.java
ai_ds-a2@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/harizzesh$ java Exercise6.Q1
Enter the string: Hariesh
Enter the index: 4
Character at index 4 is e
ai_ds-a2@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/harizzesh$
```

Q2)

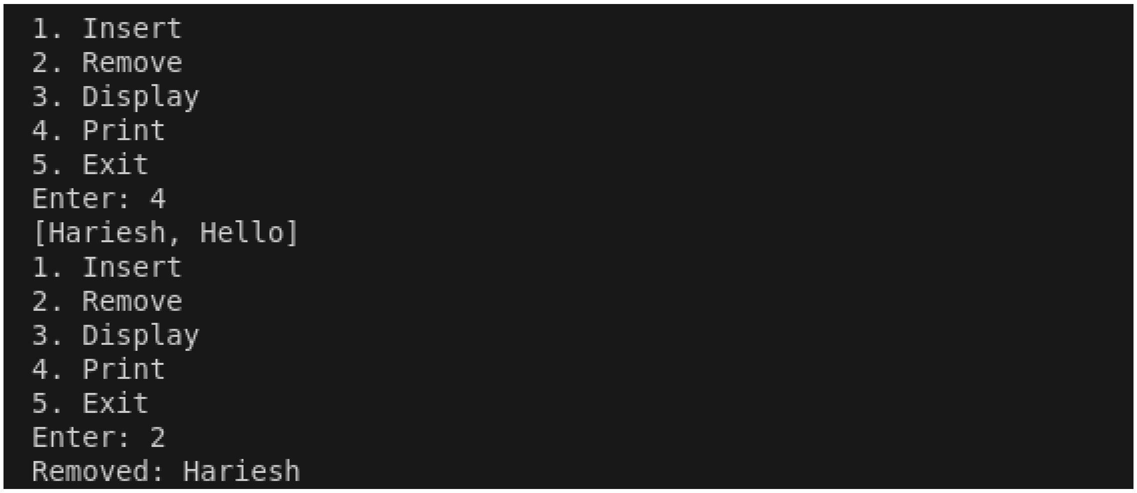


```
ai_ds-a2@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/harizzesh$ javac Exercise6/Q2.java
ai_ds-a2@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/harizzesh$ java Exercise6.Q2
Enter the word 1:anagram
Enter the word 2: grammer
Enter the value of k: 3
true
ai_ds-a2@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/harizzesh$ javac Exercise6/Q2.java
ai_ds-a2@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/harizzesh$ java Exercise6.Q2
Enter the word 1:anagram
Enter the word 2: grammer
Enter the value of k: 2
false
ai_ds-a2@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/harizzesh$
```

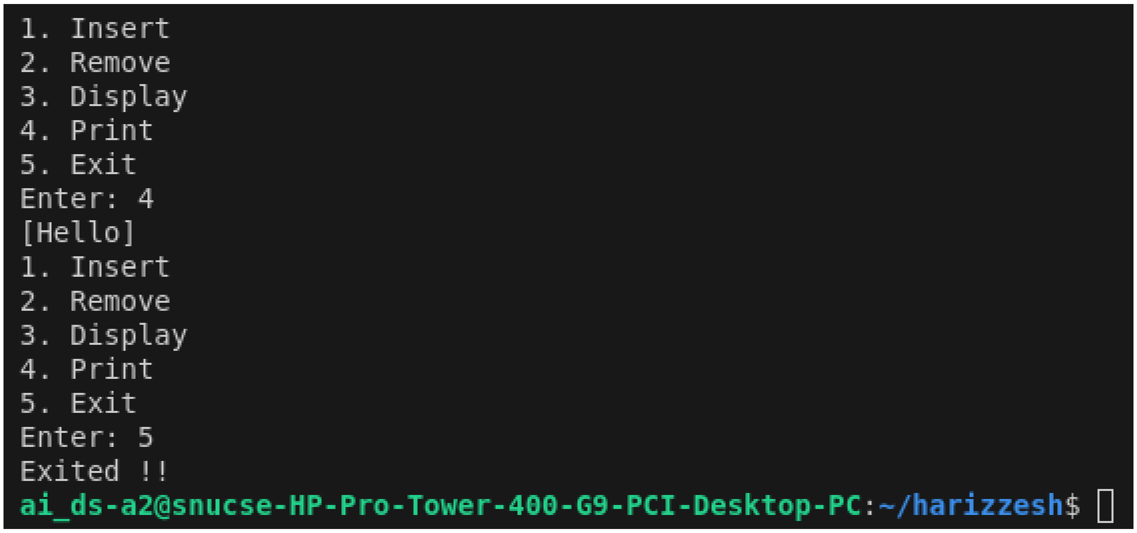
Q3)



```
ai_ds-a2@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/harizzesh$ javac Exercise6/Q3.java
ai_ds-a2@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/harizzesh$ java Exercise6.Q3
1. Insert
2. Remove
3. Display
4. Print
5. Exit
Enter: 1
Enter the value: Hariesh
1. Insert
2. Remove
3. Display
4. Print
5. Exit
Enter: 1
Enter the value: Hello
```



```
1. Insert
2. Remove
3. Display
4. Print
5. Exit
Enter: 4
[Hariesh, Hello]
1. Insert
2. Remove
3. Display
4. Print
5. Exit
Enter: 2
Removed: Hariesh
```



```
1. Insert
2. Remove
3. Display
4. Print
5. Exit
Enter: 4
[Hello]
1. Insert
2. Remove
3. Display
4. Print
5. Exit
Enter: 5
Exited !!
ai_ds-a2@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/harizzesh$
```

Result:

1) The Java program retrieves and displays the character at a specified index within a string. It prompts the user for a string and an index, then checks if the index is valid. If valid, it prints the character at that index; if not, it displays an error message.

2) The Java program determines if two strings are k-anagrams by checking if they have the same length and can be made anagrams by changing at most k characters. It takes two strings and an integer k as input, then calculates the number of changes needed and prints true if the changes are within the allowed limit, otherwise false.

3) The Java program demonstrates operations on a LinkedList by inserting elements, deleting elements, and displaying the list. It shows the list's state after each operation to confirm the changes.