

AIM:

3. Write a program to perform Encryption / Decryption using Mono alphabetic techniques.

IMPLEMENTATION

```
import java.io.*;
class GFG {
    public static char normalChar[]
        = { 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i',
            'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r',
            's', 't', 'u', 'v', 'w', 'x', 'y', 'z' };

    public static char codedChar[]
        = { 'Q', 'W', 'E', 'R', 'T', 'Y', 'U', 'I', 'O',
            'P', 'A', 'S', 'D', 'F', 'G', 'H', 'J', 'K',
            'L', 'Z', 'X', 'C', 'V', 'B', 'N', 'M' };

    // Function which returns encrypted string
    public static String stringEncryption(String s)
    {
        // initializing an empty String
        String encryptedString = "";

        // comparing each character of the string and
        // encoding each character using the indices
        for (int i = 0; i < s.length(); i++) {
            for (int j = 0; j < 26; j++) {

                // comparing the character and
                // adding the corresponding char
                // to the encryptedString
                if (s.charAt(i) == normalChar[j])
                {
                    encryptedString += codedChar[j];
                    break;
                }

                // if there are any special characters
                // add them directly to the string
                if (s.charAt(i) < 'a' || s.charAt(i) > 'z')
                {
                    encryptedString += s.charAt(i);
                    break;
                }
            }
        }
    }
}
```

```

        }
    }
}

// return encryptedString
return encryptedString;
}

// Function which returns decryptedString
public static String stringDecryption(String s)
{
    // Initializing the string
    String decryptedString = "";

    // Run the for loop for total string
    for (int i = 0; i < s.length(); i++)
    {
        for (int j = 0; j < 26; j++) {

            // compare each characters and decode them
            // using indices
            if (s.charAt(i) == codedChar[j])
            {
                decryptedString += normalChar[j];
                break;
            }

            // Add the special characters directly to
            // the String
            if (s.charAt(i) < 'A' || s.charAt(i) > 'Z')
            {
                decryptedString += s.charAt(i);
                break;
            }
        }
    }

    // return the decryptedString
    return decryptedString;
}

public static void main(String args[])
{
    String str = "Welcome to geeksforgeeks";

```

```
// print plain text
System.out.println("Plain text: " + str);

// Changing whole string to lower case
// function call to stringEncryption and storing in
// encryptedString
String encryptedString = stringEncryption(str.toLowerCase());

// printing encryptedString
System.out.println("Encrypted message: "
                  + encryptedString);

// function call to stringDecryption and printing
// the decryptedString
System.out.println("Decrypted message: "
                  + stringDecryption(encryptedString));
    }
}
```

OUTPUT: