LAB – 1

Aim : Write a program to implement

1. Additive /Shift/ Caesar Cipher
2. Monoalphabetic Substitution Cipher

1) Additive Cipher :

```
import java.util.*;

public class Shift
{
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      String pText;
      int key;

      System.out.print("Enter plain text : ");
      pText = sc.nextLine();

      for(int k = 1; k <= 25; k++) {
         StringBuilder cText = new StringBuilder();
         StringBuilder dText = new StringBuilder();

         System.out.println("key = " + k);

         for(int i = 0; i < pText.length(); i++) {
            int ascii = ((pText.charAt(i) - 'a') + k) % 26;

            cText.append((char)(ascii + 'a'));
         }

         System.out.println("Encrypted : " + cText);

         for(int i = 0; i < pText.length(); i++) {
            int ascii = ((cText.charAt(i) - 'a') - k) % 26;

            while(ascii < 0) {
               ascii += 26;
            }

            dText.append((char)(ascii + 'a'));
         }

         System.out.println("Decrypted : " + dText);
```

```
        }
    }
}
```

OUTPUT :

```
Enter plain text : welcome
Enter key : 5
Encrypted : bjqhtrj
1 is not Key
2 is not Key
3 is not Key
4 is not Key
K is 5
```

2) Monoalphabatic Cipher :

```
import java.util.*;

public class Mono {
    public static void main (String[] args) {
        ArrayList<Character> visited = new ArrayList<>();
        HashMap<Character, Character> mapping = new HashMap<>();
        HashMap<Character, Character> map = new HashMap<>();

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter text : ");
        String pText = sc.nextLine();

        for(int i = 0; i < pText.length(); i++) {
            if(!visited.contains(pText.charAt(i))) {
                visited.add(pText.charAt(i));
            }
        }

        for(int i = 0; i < 26; i++) {
            char c = (char)('a' + i);
            if(!visited.contains(c)) {
                visited.add(c);
            }
        }
```

```
        for(int i = 0; i < 26; i++) {
            char c = (char)('a' + i);
            mapping.put(c, visited.get(i));
            map.put(visited.get(i), c);
        }

        System.out.println(mapping);
        System.out.println(map);

        StringBuilder encrypted = new StringBuilder();
        StringBuilder decrypted = new StringBuilder();

        for(int i = 0; i < pText.length(); i++) {
            char ch = mapping.get(pText.charAt(i));
            encrypted.append(ch);
        }

        System.out.println("Encrypted : " + encrypted);

        for(int i = 0; i < pText.length(); i++) {
            char ch = map.get(encrypted.charAt(i));
            decrypted.append(ch);
        }

        System.out.println("Decrypted : " + decrypted);
    }
}
```

OUTPUT :

```
Enter text : welcome
{a=w, b=e, c=l, d=c, e=o, f=m, g=a, h=b, i=d, j=f, k=g, l=h, m=i, n=j, o=k, p=n, q=p, r=q, s=r, t=s, u=t, v=u, w=v, x=x, y=y, z=z}
{a=g, b=h, c=d, d=i, e=b, f=j, g=k, h=l, i=m, j=n, k=o, l=c, m=f, n=p, o=e, p=q, q=r, r=s, s=t, t=u, u=v, v=w, w=a, x=x, y=y, z=z}
Encrypted : vohlkio
Decrypted : welcome
```