

Practical no 1

AIM: WAP in Java to implement the following Substitution Cipher Techniques.

CODE**1) Caesar cipher :-**

```
package caesarcipher;
import java.util.Scanner;
public class CaesarCipher {
    public static void main(String[] args) {
        CaesarCipher c = new CaesarCipher();
        Scanner s = new Scanner(System.in);
        System.out.println("input data to encrypt");
        String str = s.nextLine();
        System.out.println("input key : ");
        int key = s.nextInt();
        String encrypt = c.encrypt(str,key);
        System.out.println("ency data : "+ encrypt);
        String decrypted = c.decrypt(encrypt, key);
        System.out.println("decy data "+ decrypted);
        System.out.println("performed by krunal 713");
    }
    String encrypt(String str, int key){
        String encrypted = "";
        for(int i =0 ; i< str.length(); i++)
        {
            int c =str.charAt(i);
            //      System.out.println("c:" + c);
```

```
        if(Character.isUpperCase(c)){
            c = (c+key);
//            System.out.println("c: " + c);
            if(c > 'Z'){
                c = c-26;
            }
        }
        else
        {
            c= c+key;
//            System.out.println("c:" +c);
            if(c > 'z'){
                c = c-26;
            }
        }
        encrypted += (char) c;
    }
    return encrypted;
}

String decrypt(String str, int key){
    String decrypted = "";
    for(int i =0 ; i< str.length(); i++)
    {
        int c =str.charAt(i);
//        System.out.println("c:" + c);
        if(Character.isUpperCase(c)){
            c = (c-key);
//            System.out.println("c: " + c);
            if(c < 'A'){
```

```
        c = c+26;
    }
}
else
{
    c= c-key;
//    System.out.println("c:" +c);
    if(c < 'a'){
        c = c+26;
    }
}
    decrypted += (char) c;
}
return decrypted;
}
}
```

Output - CaesarCipher (run) X

```
run:
input data to encrypt
krunal
input key :
3
ency data : nuxqdo
decy data krunal
performed by krunal 713
BUILD SUCCESSFUL (total time: 9 seconds)
```

2) Monoalphabetic Cipher

```
package caesarcipher;
import java.util.Scanner;
public class CaesarCipher {
    public static void main(String[] args) {
        CaesarCipher c = new CaesarCipher();
        Scanner s = new Scanner(System.in);
        System.out.println("input data to encrypt");
        String str = s.nextLine();
        System.out.println("input key : ");
        int key = s.nextInt();
        String encrypt = c.encrypt(str,key);
        System.out.println("ency data : "+ encrypt);
        String decrypted = c.decrypt(encrypt, key);
        System.out.println("decy data "+ decrypted);
        System.out.println("performed by rohan 703");
    }

    String encrypt(String str, int key){
        String encrypted = "";
        for(int i =0 ; i< str.length(); i++)
        {
            int c =str.charAt(i);
            if(Character.isUpperCase(c)){
                c = (c+key);
                if(c > 'Z'){
                    c = c-26;
                }
            }
            else
            {
                c= c+key;
                if(c > 'z'){
                    c = c-26;
                }
            }
            encrypted += (char) c;
        }
        return encrypted;
    }
    String decrypt(String str, int key){
        String decrypted = "";
```

```
for(int i =0 ; i< str.length(); i++)  
{  
    int c =str.charAt(i);  
    if(Character.isUpperCase(c)){  
        c = (c-key);  
        if(c < 'A'){  
            c = c+26;  
        }  
    }  
    else  
    {  
        c= c-key;  
        if(c < 'a'){  
            c = c+26;  
        }  
    }  
    decrypted += (char) c;  
}  
return decrypted;  
}
```

Output - MonoalphabeticCipher (run) X

```
run:  
Performed by krunal 713  
Enter the message:  
krunal dhavle  
Encrypted message:AKXFQS  
Decrypted message:krunal  
BUILD SUCCESSFUL (total time: 10 seconds)
```