AIM: To implement Rail fence Technique

In transposition cipher the alphabets of the plain text are rearranged in a different & usually quite complex order.

Rail fence technique is an example of transposition technique. It involves writing plain text as sequence of diagonals & then reading them row by row to produce cipher text

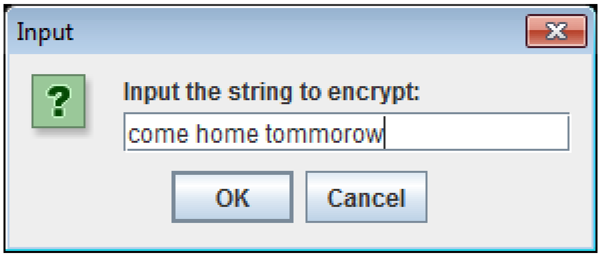
In the program we are implementing Rail fence technique which is a type of transposition cipher. Program consist of two methods encrypt and decrypt to implement the concept.

Program

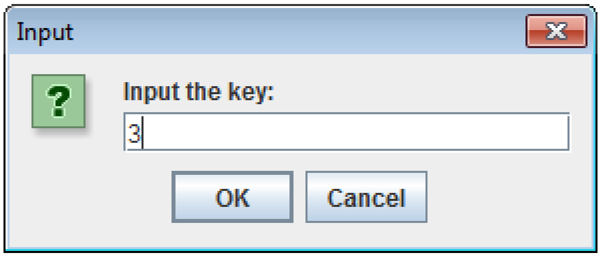
```
importjavax.swing.JOptionPane;
public class RailFence
{
public static void main(String[] args)
   {
      // TODO code application logic here
      String plain_text;
                    int key;
                    String cipher1,cipher2;
                    plain_text=JOptionPane.showInputDialog("Input the string to encrypt:");
                    key = Integer.parseInt(JOptionPane.showInputDialog("Input the key:"));
                    cipher1=encrypt(plain_text,key);
                    JOptionPane.showMessageDialog (null, "Cipher Text is   " + cipher1, "Encryption
Process", JOptionPane.PLAIN_MESSAGE);
                    cipher2=decrypt(cipher1,key);
                    JOptionPane.showMessageDialog (null, "Plain Text is  " + cipher2, "Decryption Process",
JOptionPane.PLAIN_MESSAGE);
   }
public static String encrypt(String plainText,int depth)
          {
                    int r=depth,len=plainText.length();
                    int c=len/depth;
                    char mat[][]=new char[r][c];
                    int k=0;
                    String cipherText="";
                    for(int i=0;i<c;i++)
                    {
                              for(int j=0;j<r;j++)
                              {
                                        if(k!=len)
                                                  mat[j][i]=plainText.charAt(k++);
                                        else
                                                  mat[j][i]='X';
                              }
                    }
                    for(int i=0;i<r;i++)
                    {
                              for(int j=0;j<c;j++)
                              {
                                        cipherText+=mat[i][j];
                              }
                    }
                    returncipherText;
          }

          public static String decrypt(String cipherText,int depth)
          {
```

```
        int r=depth,len=cipherText.length();
        int c=len/depth;
        char mat[][]=new char[r][c];
        int k=0;
        String plainText="";
        for(int i=0;i<r;i++)
        {
            for(int j=0;j<c;j++)
            {
                mat[i][j]=cipherText.charAt(k++);
            }
        }
        for(int i=0;i<c;i++)
        {
            for(int j=0;j<r;j++)
            {
                plainText+=mat[j][i];
            }
        }
        returnplainText;
    }
}
```

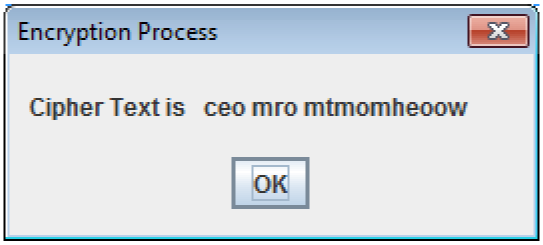
Output:



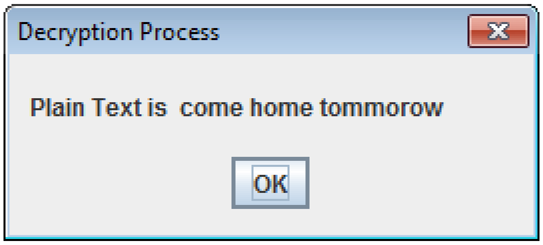
Input the string to encrypt: come home tommorow



Input the key: 3



Cipher Text is   ceo mro mtmomheoow



Plain Text is come home tommorow