PROJECT REPORT

on

Image Cryptography

(CSE IV Semester Mini Project PCS-404)

2021-2022

Submitted to:

Dr. Prateek Srivastava

(CC-CSE-B-IV -Sem)

Submitted by:

Manmohan Rajpal

Roll. No.: 2018471

CSE-B-IV-Sem

Session: 2021-2022

DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

GRAPHIC ERA HILL UNVERSITY, DEHRADUN

CERTIFICATE

Certified that Mr. Manmohan Rajpal(Roll No.- 2018471) has developed mini project on "**Image Cryptography**" for the CS IV Semester Mini Project Lab (PCS-404) in Graphic Era Hill University, Dehradun. The project carried out by Students is their own work as best of my knowledge.

Dr. Prateek Srivastava

Class Co-ordinator

CSE-B-IV-Sem

(CSE Department)

GEHU Dehradun

ACKNOWLEDGMENT

We would like to express our gratitude to The Almighty, the most Beneficent and the most Merciful, for completion of project.

I wish to thank our parents for their continuing support and encouragement. We also wish to thank them for providing us with the opportunity to reach this far in our studies.

I would like to thank particularly our project Co-ordinator Dr. Prateek Srivastava for his patience, support, and encouragement throughout the completion of this project and having faith in us.

We also acknowledge them who helped me in developing the project.

At last but not the least we are greatly indebted to all other persons who directly or indirectly helped us during this work.

Mr. Manmohan Rajpal

Roll No.- 2018471

Student ID: 20011019

CSE-B-IV-Sem

Session: 2021-2022

GEHU, Dehradun

TABLE OF CONTENTS

INTRODUCTION

1.1 OBJECTIVE

In the current trends, the technologies have been advanced. Most of the individuals prefer using the internet as the primary medium to transfer data from one end to another across the internet. There are many possible ways to transmit data using the internet like: via e-mails, sending text and images, etc. In the present communication world, images are widely in use. However, one of the main problems with sending data over the Internet is the 'security' and authenticity.

Encryption is one of the techniques for the information security. Image encryption is a technique that convert original image to another form that is difficult to understand. No one can access the content without knowing a decryption key. Image encryption has applications in corporate world, health care, military operations, and multimedia systems. Encryption is the process of encoding plain text message into cipher text message whereas reverse process of transforming cipher text to plain text is called as decryption. Cryptography consists of encryption and decryption techniques.

1.2 OVERVIEW

In this project, the user has to select the they want to do encryption or decryption and then click on the "encryption or decryption" button. It will shows "**done**" below when the file perfectly encrypts or decrypt. The user can also choose the file from the location.

- "Encryption" is the process of encoding plain text message into cipher text message
- "Decryption" is reverse process of transforming cipher text to plain text

and if the user clicks on the "Cut" button, then it will close automatically.

PROJECT

2.1 REQUIREMENTS OF PROJECT

2.1.1 IDE Software:

IDE stands for integrated development environment; it is a software that combines all the feature and tools needed by software developers. In this project, we have used Visual Studio Code for compiling the programs.

2.1.2 Programming Language:

The Java programming language was developed by Sun Microsystems in the early 1990s. Although it is primarily used for Internet-based applications, Java is a simple, efficient, general-purpose language. Java was originally designed for embedded network applications running on multiple platforms. It is a portable, object-oriented, interpreted language.

Java is extremely portable. The same Java application will run identically on any computer, regardless of hardware features or operating system, as long as it has a Java interpreter. Besides portability, another of Java's key advantages is its set of security features which protect a PC running a Java program not only from problems caused by erroneous code but also from malicious programs (such as viruses). You can safely run a Java applet downloaded from the Internet, because Java's security features prevent these types of applets from accessing a PC's hard drive or network connections. An *applet* is typically a small Java program that is embedded within an HTML page.

2.2 SNAPSHOTS:

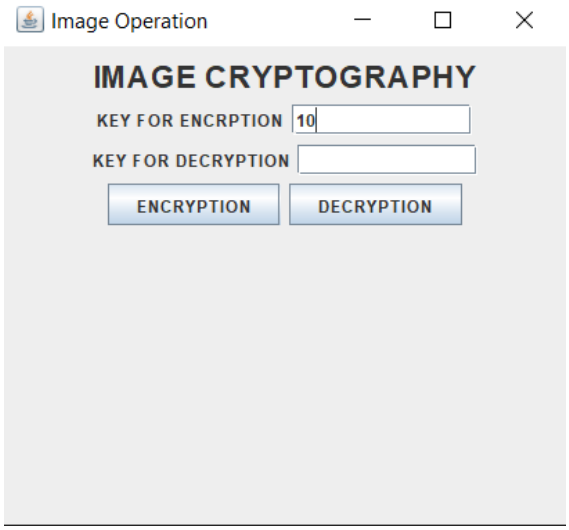


Fig. 1: Click "Encryption"

Fig. 2: Click "Decryption"

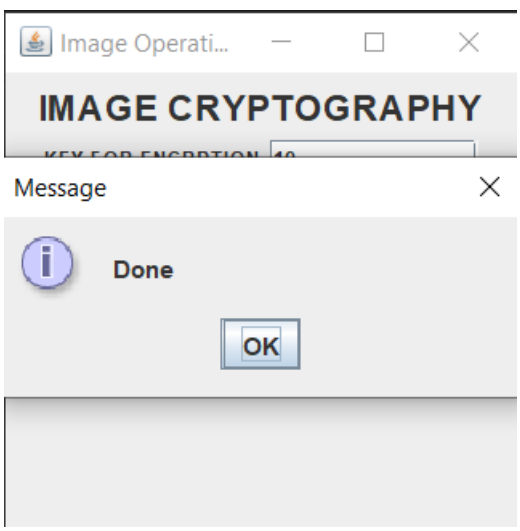


Fig. 3: Message after Encryption or Decryption Process done

and if file is not Encrypted then they show file is not Encrypted.

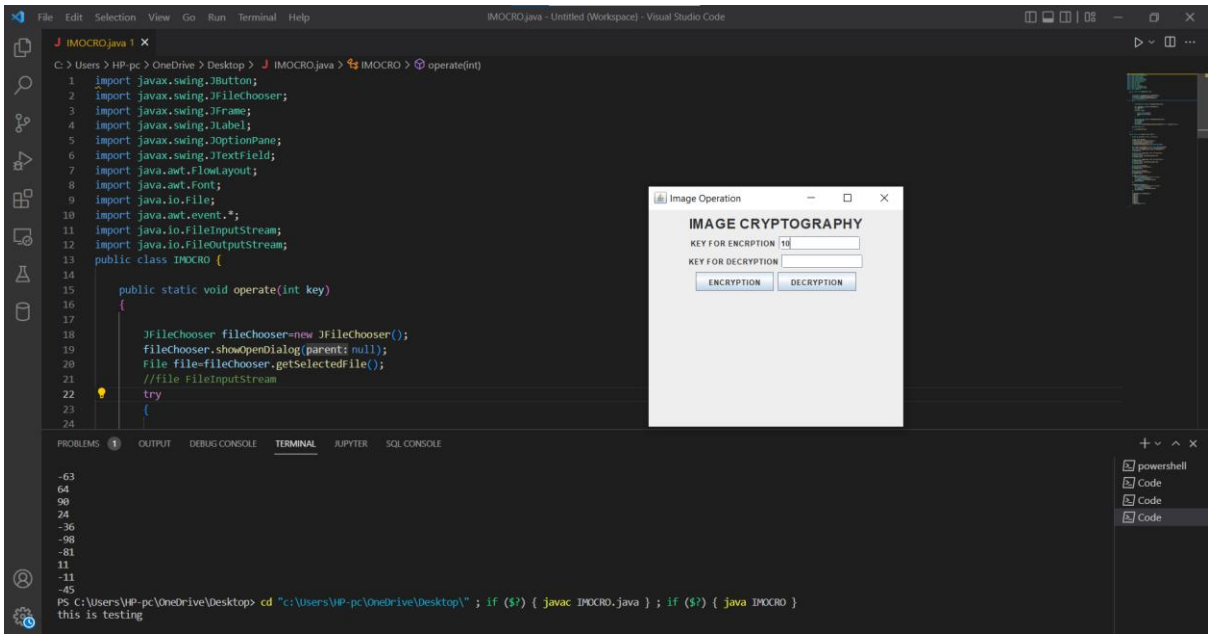


Fig. 4: Running application with Source Code

CONCLUSION

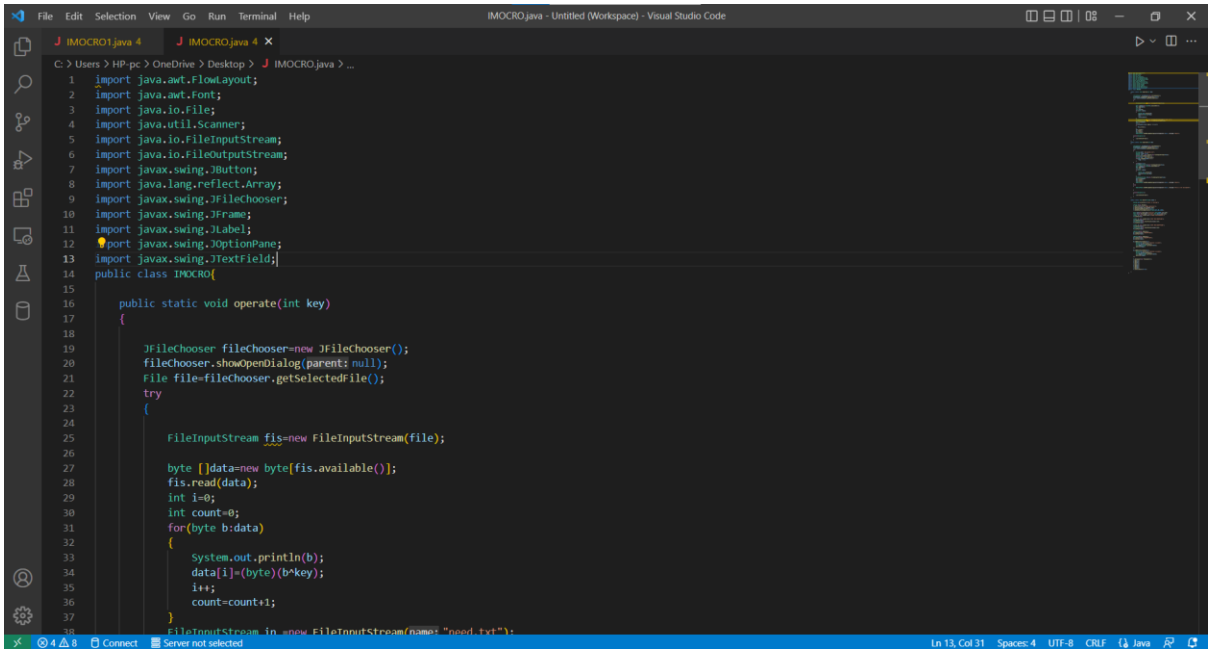
3.1 Summary:

Image Cryptography is a Encryption or Decryption program which is use to encrypt or decrypt the image ,as well as files.

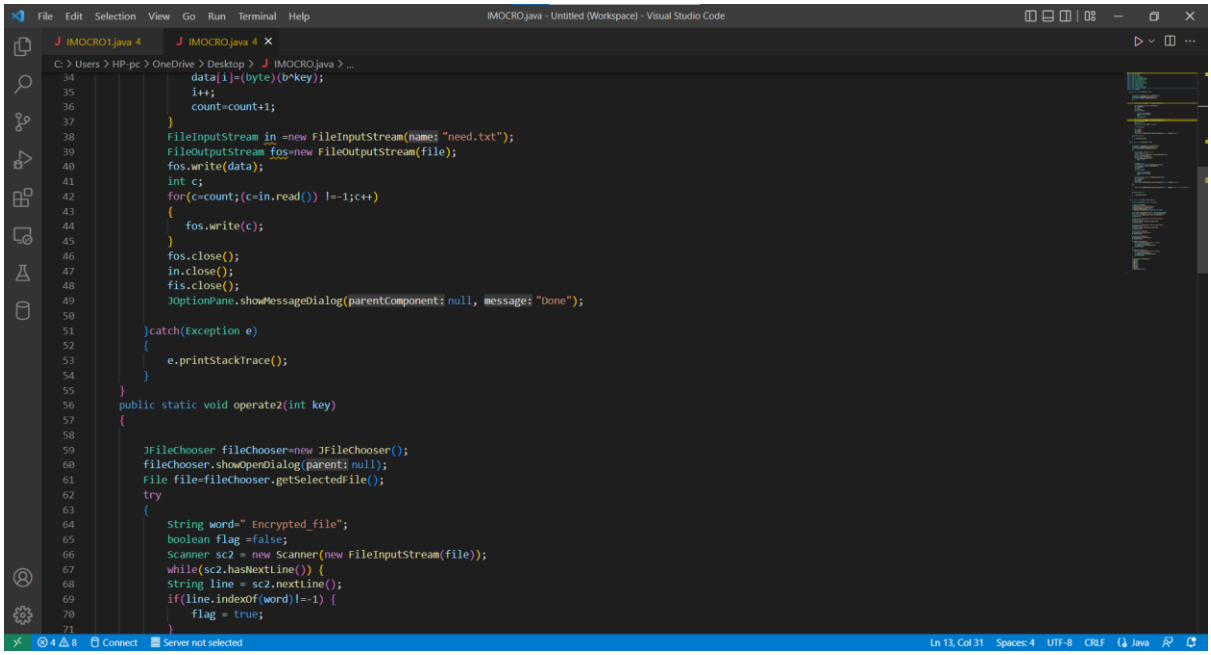
3.2 Future Goals:

Future research directions related to image encryption strategies were examined. It was found that the development of image encryption approaches is still an open area for researchers. This project encourages researchers to understand the challenges involved in image encryption approaches.

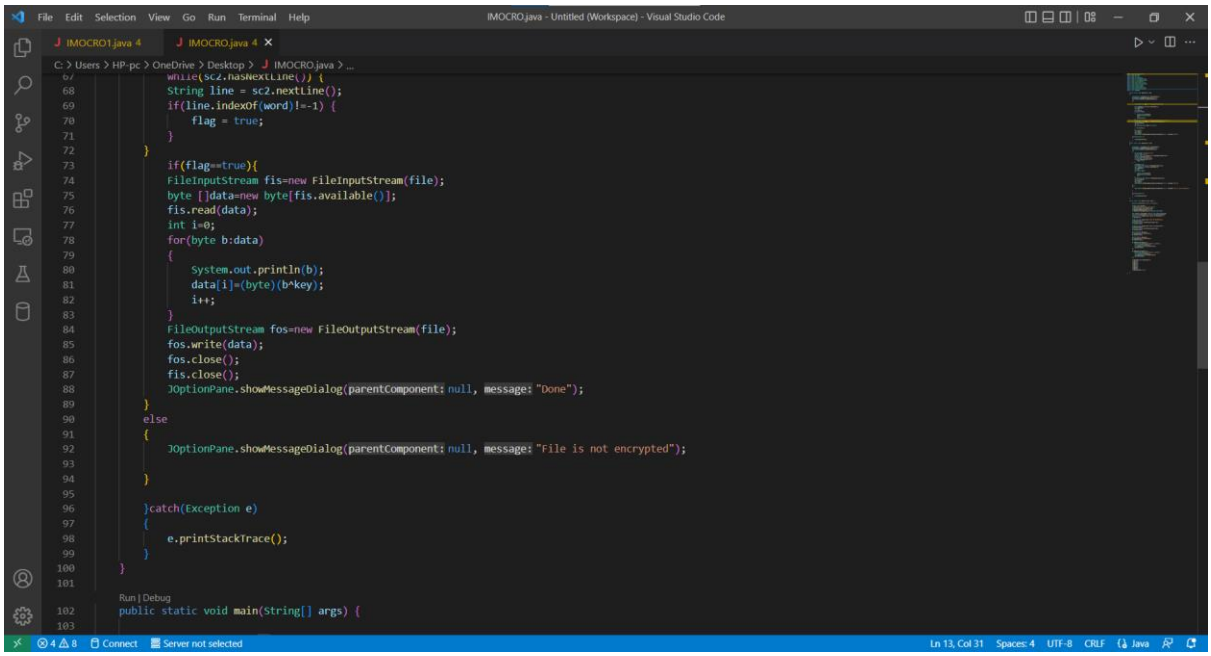
Source Code

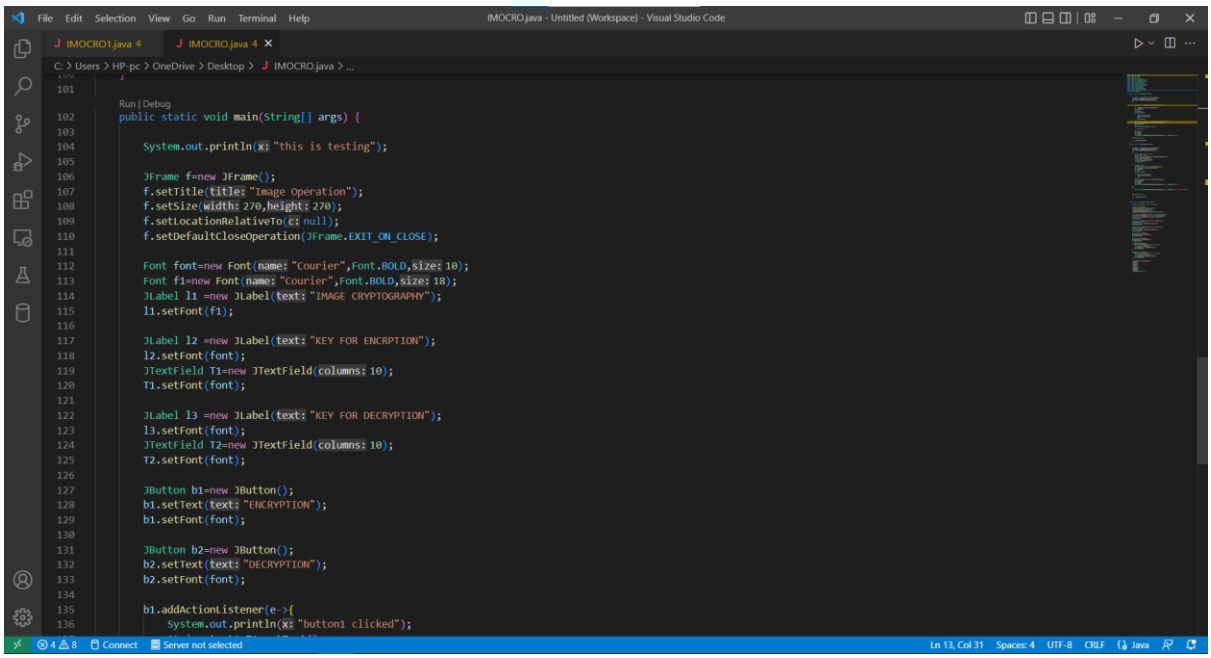


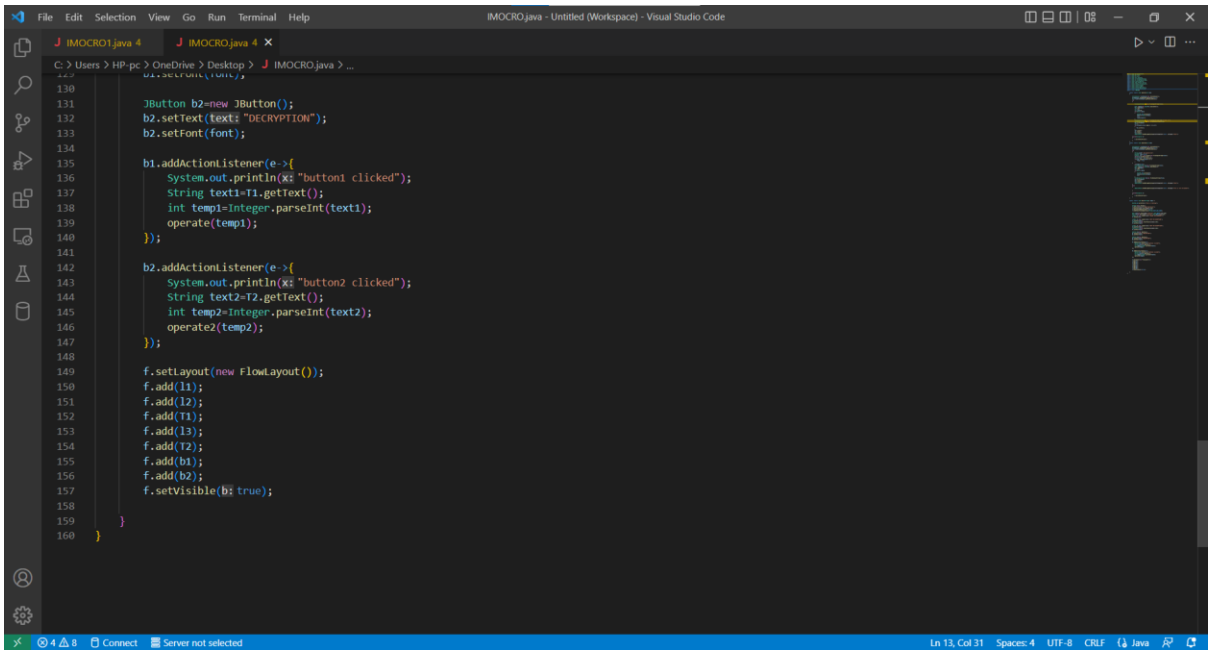
```java
import java.awt.FlowLayout;
import java.awt.Font;
import java.io.File;
import java.util.Scanner;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import javax.swing.JButton;
import java.lang.reflect.Array;
import javax.swing.JFileChooser;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JTextField;
public class IMOCRO{

    public static void operate(int key)
    {

        JFileChooser fileChooser=new JFileChooser();
        fileChooser.showOpenDialog(parent: null);
        File file=fileChooser.getSelectedFile();
        try
        {

            FileInputStream fis=new FileInputStream(file);

            byte []data=new byte[fis.available()];
            fis.read(data);
            int i=0;
            int count=0;
            for(byte b:data)
            {
                System.out.println(b);
                data[i]=(byte)(b^key);
                i++;
                count=count+1;
            }
            FileInputStream in =new FileInputStream(name: "need.txt");
```



```java
                data[i]=(byte)(b^key);
                i++;
                count=count+1;
            }
            FileInputStream in =new FileInputStream(name: "need.txt");
            FileOutputStream fos=new FileOutputStream(file);
            fos.write(data);
            int c;
            for(c=count;(c=in.read()) !=-1;c++)
            {
               fos.write(c);
            }
            fos.close();
            in.close();
            fis.close();
            JOptionPane.showMessageDialog(parentComponent: null, message: "Done");

        }catch(Exception e)
        {
            e.printStackTrace();
        }
    }
    public static void operate2(int key)
    {

        JFileChooser fileChooser=new JFileChooser();
        fileChooser.showOpenDialog(parent: null);
        File file=fileChooser.getSelectedFile();
        try
        {
            String word=" Encrypted_file";
            boolean flag =false;
            Scanner sc2 = new Scanner(new FileInputStream(file));
            while(sc2.hasNextLine()) {
            String line = sc2.nextLine();
            if(line.indexOf(word)!=-1) {
                flag = true;
            }
```


File Edit Selection View Go Run Terminal Help
IMOCRO.java - Untitled (Workspace) - Visual Studio Code
IMOCRO1.java 4
IMOCRO.java 4
C: > Users > HP-pc > OneDrive > Desktop > IMOCRO.java > ...
while(sc2.hasNextLine()) {
String line = sc2.nextLine();
if(line.indexOf(word)!=-1) {
flag = true;
}
}
if(flag==true){
FileInputStream fis=new FileInputStream(file);
byte []data=new byte[fis.available()];
fis.read(data);
int i=0;
for(byte b:data)
{
System.out.println(b);
data[i]=(byte)(b^key);
i++;
}
FileOutputStream fos=new FileOutputStream(file);
fos.write(data);
fos.close();
fis.close();
JOptionPane.showMessageDialog(parentComponent: null, message: "Done");
}
else
{
JOptionPane.showMessageDialog(parentComponent: null, message: "File is not encrypted");
}catch(Exception e)
{
e.printStackTrace();
}
}
Run | Debug
public static void main(String[] args) {
Connect
Server not selected
Ln 13, Col 31
Spaces: 4
UTF-8
CRLF
Java


File Edit Selection View Go Run Terminal Help
IMOCRO.java - Untitled (Workspace) - Visual Studio Code
IMOCRO1.java 4
IMOCRO.java 4
C: > Users > HP-pc > OneDrive > Desktop > IMOCRO.java > ...
}
Run | Debug
public static void main(String[] args) {
System.out.println(x: "this is testing");
JFrame f=new JFrame();
f.setTitle(title: "Image Operation");
f.setSize(width: 270,height: 270);
f.setLocationRelativeTo(c: null);
f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
Font font=new Font(name: "Courier",Font.BOLD,size: 10);
Font f1=new Font(name: "Courier",Font.BOLD,size: 18);
JLabel l1 =new JLabel(text: "IMAGE CRYPTOGRAPHY");
l1.setFont(f1);
JLabel l2 =new JLabel(text: "KEY FOR ENCRPTION");
l2.setFont(font);
JTextField T1=new JTextField(columns: 10);
T1.setFont(font);
JLabel l3 =new JLabel(text: "KEY FOR DECRYPTION");
l3.setFont(font);
JTextField T2=new JTextField(columns: 10);
T2.setFont(font);
JButton b1=new JButton();
b1.setText(text: "ENCRYPTION");
b1.setFont(font);
JButton b2=new JButton();
b2.setText(text: "DECRYPTION");
b2.setFont(font);
b1.addActionListener(e->{
System.out.println(x: "button1 clicked");
Connect
Server not selected
Ln 13, Col 31
Spaces: 4
UTF-8
CRLF
Java


File Edit Selection View Go Run Terminal Help
IMOCRO.java - Untitled (Workspace) - Visual Studio Code
IMOCRO1.java 4
IMOCRO.java 4
C: > Users > HP-pc > OneDrive > Desktop > IMOCRO.java > ...
JButton b2=new JButton();
b2.setText(text: "DECRYPTION");
b2.setFont(font);
b1.addActionListener(e->{
System.out.println(x: "button1 clicked");
String text1=T1.getText();
int temp1=Integer.parseInt(text1);
operate(temp1);
});
b2.addActionListener(e->{
System.out.println(x: "button2 clicked");
String text2=T2.getText();
int temp2=Integer.parseInt(text2);
operate2(temp2);
});
f.setLayout(new FlowLayout());
f.add(l1);
f.add(l2);
f.add(T1);
f.add(l3);
f.add(T2);
f.add(b1);
f.add(b2);
f.setVisible(b: true);
}
}
Connect
Server not selected
Ln 13, Col 31
Spaces: 4
UTF-8
CRLF
Java

REFERENCES

- Get Programming, Learn to code with java from college
- Geeks For Geeks
- YouTube
- javatpoint.com
- Reference Book – Java The Complete Reference Eleventh Edition by Herbet Schildt