* **Program Code:**

int pc1[56] = {

57,49,41,33,25,17,9,

1,58,50,42,34,26,18,

10,2,59,51,43,35,27,

19,11,3,60,52,44,36,

63,55,47,39,31,23,15,

7,62,54,46,38,30,22,

14,6,61,53,45,37,29,

21,13,5,28,20,12,4

};

int pc2[48] = {

14,17,11,24,1,5,

3,28,15,6,21,10,

23,19,12,4,26,8,

16,7,27,20,13,2,

41,52,31,37,47,55,

30,40,51,45,33,48,

44,49,39,56,34,53,

46,42,50,36,29,32

};

package day1;

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.InputStream;

import java.io.OutputStream;

import java.security.InvalidAlgorithmParameterException;

import java.security.InvalidKeyException;

import java.security.NoSuchAlgorithmException;

import java.security.spec.AlgorithmParameterSpec;

import javax.crypto.Cipher;

import javax.crypto.CipherInputStream;

import javax.crypto.CipherOutputStream;

import javax.crypto.KeyGenerator;

import javax.crypto.NoSuchPaddingException;

import javax.crypto.SecretKey;

import javax.crypto.spec.IvParameterSpec;

public class DesProgram {

private static Cipher encrypt;

private static Cipher decrypt;

private static final byte[] initialization\_vector = { 22, 33, 11, 44, 55, 99, 66, 77 };

public static void main(String[] args)

{

String textFile = "C:/Users/NIKITA/Desktop/DemoData.txt";

String encryptedData = "C:/Users/NIKITA/Desktop/encrypteddata.txt";

String decryptedData = "C:/Users/NIKITA/Desktop/decrypteddata.txt";

try

{

SecretKey scrtkey = KeyGenerator.getInstance("DES").generateKey();

AlgorithmParameterSpec aps = new IvParameterSpec(initialization\_vector);

encrypt = Cipher.getInstance("DES/CBC/PKCS5Padding");

encrypt.init(Cipher.ENCRYPT\_MODE, scrtkey, aps);

decrypt = Cipher.getInstance("DES/CBC/PKCS5Padding");

decrypt.init(Cipher.DECRYPT\_MODE, scrtkey, aps);

encryption(new FileInputStream(textFile), new FileOutputStream(encryptedData));

decryption(new FileInputStream(encryptedData), new FileOutputStream(decryptedData));

System.out.println("The encrypted and decrypted files have been created successfully.");

}

catch (NoSuchAlgorithmException | NoSuchPaddingException | InvalidKeyException | InvalidAlgorithmParameterException | IOException e)

{

e.printStackTrace();

}

}

private static void encryption(InputStream input, OutputStream output)

throws IOException

{

output = new CipherOutputStream(output, encrypt);

writeBytes(input, output);

}

private static void decryption(InputStream input, OutputStream output)

throws IOException

{

input = new CipherInputStream(input, decrypt);

writeBytes(input, output);

}

private static void writeBytes(InputStream input, OutputStream output)

throws IOException

{

byte[] writeBuffer = new byte[512];

int readBytes = 0;

while ((readBytes = input.read(writeBuffer)) >= 0)

{

output.write(writeBuffer, 0, readBytes);

}

output.close();

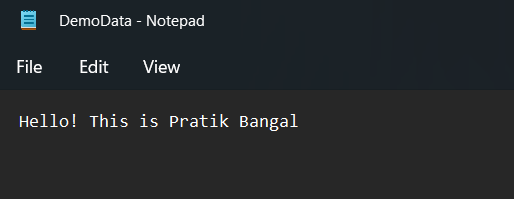
input.close();

}

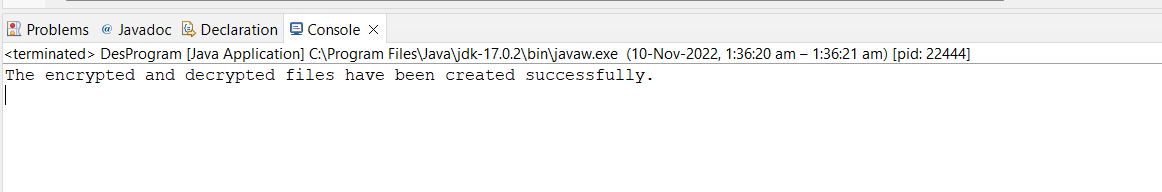
}

* **Output:**

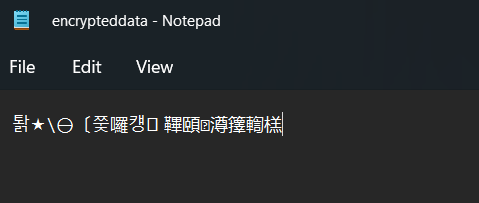
1. DemoData.txt



1. IDE output:



1. encrypteddata.txt :



1. deecrypteddata.txt:

