Coding:

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/Anubhav/Desktop/DemoData.txt";

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

String decryptedData = "C:/Users/Anubhav/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:

Screenshot 2025-02-02 133646

