```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
//package cifrado_claveprivadades;

/**
 *
 * @author IMCG
 */
import java.security.*; //JCA
import javax.crypto.*; //JCE
import java.io.*; //ficheros

//Programa que encripta y desencripta un fichero
//mediante clave privada o simétrica utilizando el algoritmo DES
public class Main {

    public static void main(String[] Args) {
        //declara e incializa objeto tipo clave secreta
        SecretKey clave = null;

        //llama a los métodos que encripta/desencripta un fichero
        try {
            //Llama al método que encripta el fichero que se pasa como parámetro
            clave = cifrarFichero("original.txt");
            //Llama la método que desencripta el fichero pasado como primer
parámetro
            descifrarFichero("original.txt.cifrado",
clave,"original.txt.descifrado");
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    //método que encripta el fichero que se pasa como parámetro
    //devuelve el valor de la clave privada utilizada en encriptación
    //El fichero encriptado lo deja en el archivo de nombre fichero.cifrado
    //en el mismo directorio
    private static SecretKey cifrarFichero(String file) throws
NoSuchAlgorithmException, NoSuchPaddingException, FileNotFoundException,
IOException, IllegalBlockSizeException, BadPaddingException, InvalidKeyException
{

        FileInputStream fe = null; //fichero de entrada
        FileOutputStream fs = null; //fichero de salida
        int bytesLeidos;

        //1. Crear e inicializar clave
        System.out.println("1.-Genera clave DES");
        //crea un objeto para generar la clave usando algoritmo DES
        KeyGenerator keyGen = KeyGenerator.getInstance("DES");
        keyGen.init(56); //se indica el tamaño de la clave
        SecretKey clave = keyGen.generateKey(); //genera la clave privada

        System.out.println("Clave");
        mostrarBytes(clave.getEncoded()); //muestra la clave
        System.out.println();


        //Se Crea el objeto Cipher para cifrar, utilizando el algoritmo DES
        Cipher cifrador = Cipher.getInstance("DES");
        //Se inicializa el cifrador en modo CIFRADO o ENCRIPTACIÓN
        cifrador.init(Cipher.ENCRYPT_MODE, clave);
```

```
        System.out.println("2.- Cifrar con DES el fichero: " + file
                + ", y dejar resultado en " + file + ".cifrado");
        //declaración  de objetos
        byte[] buffer = new byte[1000]; //array de bytes
        byte[] bufferCifrado;
        fe = new FileInputStream(file); //objeto fichero de entrada
        fs = new FileOutputStream(file + ".cifrado"); //fichero de salida
        //lee el fichero de 1k en 1k y pasa los fragmentos leidos al cifrador
        bytesLeidos = fe.read(buffer, 0, 1000);
        while (bytesLeidos != -1) {//mientras no se llegue al final del fichero
            //pasa texto claro al cifrador y lo cifra, asignándolo a
bufferCifrado
            bufferCifrado = cifrador.update(buffer, 0, bytesLeidos);
            fs.write(bufferCifrado); //Graba el texto cifrado en fichero
            bytesLeidos = fe.read(buffer, 0, 1000);
        }
        bufferCifrado = cifrador.doFinal(); //Completa el cifrado
        fs.write(bufferCifrado); //Graba el final del texto cifrado, si lo hay
        //Cierra ficheros
        fe.close();
        fs.close();
        return clave;
    }

    //método que desencripta el fichero pasado como primer parámetro file1
    //pasándole también la clave privada que necesita para desencriptar, key
    //y deja el fichero desencriptado en el tercer parámetro file2

    private static void descifrarFichero(String file1, SecretKey key, String
file2) throws NoSuchAlgorithmException, NoSuchPaddingException,
FileNotFoundException, IOException, IllegalBlockSizeException,
BadPaddingException, InvalidKeyException {

        FileInputStream fe = null; //fichero de entrada
        FileOutputStream fs = null; //fichero de salida
        int bytesLeidos;
        Cipher cifrador = Cipher.getInstance("DES");
        //3.- Poner cifrador en modo DESCIFRADO o DESENCRIPTACIÓN
        cifrador.init(Cipher.DECRYPT_MODE, key);
        System.out.println("3.- Descifrar con DES el fichero: " + file1
            + ", y dejar en  " + file2);
        fe = new FileInputStream(file1);
        fs = new FileOutputStream(file2);
        byte[] bufferClaro;
        byte[] buffer = new byte[1000]; //array de bytes
        //lee el fichero de 1k en 1k y pasa los fragmentos leidos al cifrador
        bytesLeidos = fe.read(buffer, 0, 1000);
        while (bytesLeidos != -1) {//mientras no se llegue al final del fichero
            //pasa texto cifrado al cifrador y lo descifra, asignándolo a
bufferClaro
            bufferClaro = cifrador.update(buffer, 0, bytesLeidos);
            fs.write(bufferClaro); //Graba el texto claro en fichero
            bytesLeidos = fe.read(buffer, 0, 1000);
        }
        bufferClaro = cifrador.doFinal(); //Completa el descifrado
        fs.write(bufferClaro); //Graba el final del texto claro, si lo hay
        //cierra archivos
        fe.close();
        fs.close();
    }

    //método que muestra bytes
    public static void mostrarBytes(byte[] buffer) {
        System.out.write(buffer, 0, buffer.length);   }}
```