Week 4 ) Write a Java program to implement the DES algorithm logic.

import java.util.\*;

import java.io.BufferedReader;

import java.io.InputStreamReader;

import java.security.spec.KeySpec;

import javax.crypto.Cipher;

import javax.crypto.SecretKey;

import javax.crypto.SecretKeyFactory;

import javax.crypto.spec.DESedeKeySpec;

import sun.misc.BASE64Decoder;

import sun.misc.BASE64Encoder;

public class DES{

private static final String UNICODE\_FORMAT = "UTF8";

public static final String DESEDE\_ENCRYPTION\_SCHEME = "DESede";

private KeySpec myKeySpec;

private SecretKeyFactorymySecretKeyFactory;

private Cipher cipher;

byte[] keyAsBytes;

private String myEncryptionKey;

private String myEncryptionScheme;

key;

static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

public DES()throws Exception{

// TODO code application logic here my

myEncryptionKey= "ThisIsSecretEncryptionKey";

myEncryptionScheme = DESEDE\_ENCRYPTION\_SCHEME;

keyAsBytes=myEncryptionKey.getBytes(UNICODE\_FORMAT);

myKeySpec= new DESedeKeySpec(keyAsBytes);

mySecretKeyFactory = SecretKeyFactory.getInstance(myEncryptionScheme);

cipher = Cipher.getInstance(myEncryptionScheme);

key = mySecretKeyFactory.generateSecret(myKeySpec);

}

public String encrypt(String unencryptedString)

{

String encryptedString = null;

try {

cipher.init(Cipher.ENCRYPT\_MODE, key);

byte[] plainText = unencryptedString.getBytes(UNICODE\_FORMAT);

byte[] encryptedText = cipher.doFinal(plainText);

BASE64Encoder base64encoder = new BASE64Encoder();

encryptedString= base64encoder.encode(encryptedText); }

catch(Exception e)

{

e.printStackTrace();

}

return encryptedString;

}

public String decrypt(String encryptedString)

{

String decryptedText=null;

try

{

cipher.init(Cipher.DECRYPT\_MODE, key);

BASE64Decoder base64decoder = new BASE64Decoder(); byte[]

encryptedText = base64decoder.decodeBuffer(encryptedString); byte[] plainText =

cipher.doFinal(encryptedText); decryptedText= bytes2String(plainText); }

catch (Exception e)

{

e.printStackTrace();

}

return decryptedText;

}

private static String bytes2String(byte[] bytes)

{ StringBufferstringBuffer =new StringBuffer(); for (int i

= 0; i <bytes.length;

i++) { stringBuffer.append((char) bytes[i]); }

return stringBuffer.toString();

}

public static void main(String args []) throws Exception

{ System.out.print("Enter the string: "); DES

myEncryptor= new DES();

String stringToEncrypt = br.readLine();

}

}

String encrypted = myEncryptor.encrypt(stringToEncrypt);

String decrypted= myEncryptor.decrypt(encrypted); System.out.println("\nString To Encrypt: " +stringToEncrypt);

System.out.println("\nEncrypted Value : "+encrypted);

System.out.println("\nDecrypted Value : " +decrypted); System.out.println("");

OUTPUT:

Enter the string:Welcome

String To Encrypt: Welcome

Encrypted Value : BPQMwc0wKvg=

Decrypted Value: Welcome