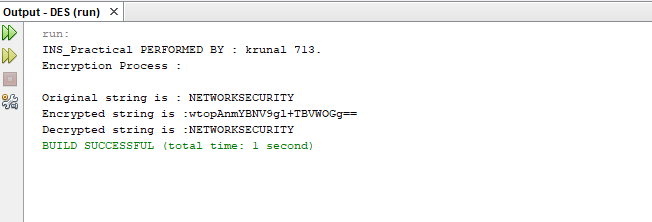
**Practical no 4**

**AIM:** Write program to encrypt and decrypt strings using

1) DES Algorithm 2) AES Algorithm

**CODE**

|  |
| --- |
| package des;  import java.util.logging.Level;  import java.util.logging.Logger;  import javax.crypto.Cipher;  import javax.crypto.KeyGenerator;  import javax.crypto.SecretKey;  public class DES {    public static SecretKey getSecretEncryptionKey() throws Exception  {  KeyGenerator generator=KeyGenerator.getInstance("DES");  SecretKey secKey=generator.generateKey();  return secKey;  }    public String encrypt(SecretKey key,String Plaintext) throws Exception{  byte[] utf8=Plaintext.getBytes("UTF8");  Cipher ecipher=Cipher.getInstance("DES");  ecipher.init(Cipher.ENCRYPT\_MODE, key);  byte[] enc=ecipher.doFinal(utf8);  return new sun.misc.BASE64Encoder().encode(enc);  }    public String decrypt(SecretKey key , String Ciphertext) throws Exception{  byte[] dec=new sun.misc.BASE64Decoder().decodeBuffer(Ciphertext);  Cipher dcipher =Cipher.getInstance("DES");  dcipher.init(Cipher.DECRYPT\_MODE,key);  byte[] utf8 =dcipher.doFinal(dec);  return new String(utf8,"UTF8");  }    public static void main(String[] args) {  // TODO code application logic here  try{  System.out.println("INS\_Practical PERFORMED BY : krunal 713.");  System.out.println("Encryption Process :");  System.out.println();  String message="NETWORKSECURITY";  DES d=new DES();  SecretKey key =getSecretEncryptionKey();  String Encrypted=d.encrypt(key, message);  String Decrypted=d.decrypt(key, Encrypted);  System.out.println("Original string is : " +message);  System.out.println("Encrypted string is :"+Encrypted);  System.out.println("Decrypted string is :"+Decrypted);    }catch(Exception ex){  Logger.getLogger(DES.class.getName()).log(Level.SEVERE,null,ex);  }  }  } |

****

**b) AES CODE**

|  |
| --- |
| package aes;  import java.util.logging.Logger;  import java.util.logging.Level;  import javax.crypto.Cipher;  import javax.crypto.KeyGenerator;  import javax.crypto.SecretKey;  public class AES {  public static SecretKey getSecretEncryptionKey() throws Exception{  KeyGenerator generator = KeyGenerator.getInstance("AES");  SecretKey secKey= generator.generateKey();  return secKey;  }    public String encrypt(SecretKey key,String Plaintext)throws Exception{  byte[] utf8= Plaintext.getBytes("UTF8");  Cipher ecipher= Cipher.getInstance("AES");  ecipher.init(Cipher.ENCRYPT\_MODE,key);  byte[] enc= ecipher.doFinal(utf8);  return new sun.misc.BASE64Encoder().encode(enc);  }    public String decrypt(SecretKey key,String Ciphertext) throws Exception{  byte[] dec= new sun.misc.BASE64Decoder().decodeBuffer(Ciphertext);  Cipher dcipher= Cipher.getInstance("AES");  dcipher.init(Cipher.DECRYPT\_MODE,key);  byte[] utf8= dcipher.doFinal(dec);  return new String(utf8, "UTF8");  }    public static void main (String[]args) throws Exception  {  try{  System.out.println("Performed by : krunal ,713");  System.out.println("Encryption using AES");  String message="NETWORK SECURITY";  AES d= new AES();  SecretKey key= getSecretEncryptionKey();  String Encrypted= d.encrypt(key, message);  String decrypted = d.decrypt(key,Encrypted);  System.out.println("Original string is:" +message);  System.out.println("Encrypted string is:" + Encrypted);  System.out.println("Decrypted string is:" +decrypted);  }  catch(Exception ex){  Logger.getLogger(AES.class.getName()).log(Level.SEVERE,null,ex) ;  }  }  } |

