**Practical no – 4(a) Roll no- 05,46**

import javax.crypto.Cipher;

import javax.crypto.KeyGenerator;

import javax.crypto.SecretKey;

public class desexample {

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

KeyGenerator myGenerator=KeyGenerator.getInstance("DES");

SecretKey key=myGenerator.generateKey();

Cipher desCipher=Cipher.getInstance("DES");

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

byte[] bytes="This message is important".getBytes();

byte[] myEncyptedBytes=desCipher.doFinal(bytes);

System.out.println(myEncyptedBytes);

}

}

**Output:**

**D:\TYCS\INS>javac desexample.java**

**D:\TYCS\INS>java desexample**

**[B@4b6995df**

**Practical no – 4(b) Roll no- 05,46**

import java.io.UnsupportedEncodingException;

import java.security.MessageDigest;

import java.security.NoSuchAlgorithmException;

import java.util.Arrays;

import java.util.Base64;

import javax.crypto.Cipher;

import javax.crypto.spec.SecretKeySpec;

public class AES {

private static SecretKeySpec secretKey;

private static byte[] key;

public static void setKey(String myKey)

{

MessageDigest sha = null;

try {

key = myKey.getBytes("UTF-8");

sha = MessageDigest.getInstance("SHA-1");

key = sha.digest(key);

key = Arrays.copyOf(key, 16);

secretKey = new SecretKeySpec(key, "AES");

}

catch (NoSuchAlgorithmException e) {

e.printStackTrace();

}

catch (UnsupportedEncodingException e) {

e.printStackTrace();

}

}

public static String encrypt(String strToEncrypt, String secret)

{

try

{

setKey(secret);

Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");

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

return Base64.getEncoder().encodeToString(cipher.doFinal(strToEncrypt.getBytes("UTF-8")));

}

catch (Exception e)

{

System.out.println("Error while encrypting: " + e.toString());

}

return null;

}

public static String decrypt(String strToDecrypt, String secret)

{

try

{

setKey(secret);

Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5PADDING");

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

return new String(cipher.doFinal(Base64.getDecoder().decode(strToDecrypt)));

}

catch (Exception e)

{

System.out.println("Error while decrypting: " + e.toString());

}

return null;

}

public static void main(String[] args)

{

final String secretKey = "ssshhhhhhhhhhh!!!!";

String originalString = "akhil";

String encryptedString = AES.encrypt(originalString, secretKey) ;

String decryptedString = AES.decrypt(encryptedString, secretKey) ;

System.out.println(originalString);

System.out.println(encryptedString);

System.out.println(decryptedString);

}

}

**Output :**

**D:\TYCS\INS>javac AES.java**

**D:\TYCS\INS>java AES**

**akhil**

**Y/KM8Mqf9+wk88XTf4jClQ==**

**akhil**