import javax.crypto.Cipher;

import javax.crypto.KeyGenerator;

import javax.crypto.SecretKey;

import java.util.Base64;

class DESExample {

Cipher ecipher;

Cipher dcipher;

DESExample(SecretKey key) throws Exception {

ecipher = Cipher.getInstance("DES");

dcipher = Cipher.getInstance("DES");

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

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

}

public String encrypt(String str) throws Exception {

// Encode the string into bytes using utf-8

byte[] utf8 = str.getBytes("UTF8");

// Encrypt

byte[] enc = ecipher.doFinal(utf8);

// Encode bytes to base64 to get a string

return Base64.getEncoder().encodeToString(enc);

}

public String decrypt(String str) throws Exception {

// Decode base64 to get bytes

byte[] dec = Base64.getDecoder().decode(str);

byte[] utf8 = dcipher.doFinal(dec);

// Decode using utf-8

return new String(utf8, "UTF8");

}

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

final String secretText = "SPPU PUNE";

System.out.println("SecretText: " + secretText);

SecretKey key = KeyGenerator.getInstance("DES").generateKey();

DESExample encrypter = new DESExample(key);

String encrypted = encrypter.encrypt(secretText);

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

String decrypted = encrypter.decrypt(encrypted);

System.out.println("Decrypted: " + decrypted);

}

}

**Output**