

```

import javax.crypto.Cipher;
import javax.crypto.KeyGenerator;
import javax.crypto.SecretKey;
import javax.crypto.spec.SecretKeySpec;
import java.util.Base64;
import java.util.Scanner;

public class DESExample {

    // Encrypts plain text using DES algorithm
    public static String encrypt(String plainText, SecretKey key) throws Exception {
        Cipher cipher = Cipher.getInstance("DES"); // DES algorithm
        cipher.init(Cipher.ENCRYPT_MODE, key); // Initialize cipher for encryption
        byte[] encryptedBytes = cipher.doFinal(plainText.getBytes());
        return Base64.getEncoder().encodeToString(encryptedBytes); // Return
        Base64 encoded string
    }

    // Decrypts encrypted text using DES algorithm
    public static String decrypt(String encryptedText, SecretKey key) throws
    Exception {
        Cipher cipher = Cipher.getInstance("DES");
        cipher.init(Cipher.DECRYPT_MODE, key);
        byte[] decryptedBytes =
        cipher.doFinal(Base64.getDecoder().decode(encryptedText));
        return new String(decryptedBytes);
    }

    public static void main(String[] args) {
        try {
            Scanner scanner = new Scanner(System.in);

            // Generate a DES key
            KeyGenerator keyGenerator = KeyGenerator.getInstance("DES");
            SecretKey secretKey = keyGenerator.generateKey();

            // Input message
            System.out.print("Enter the message to encrypt: ");
            String message = scanner.nextLine();

            // Encrypt and decrypt
            String encrypted = encrypt(message, secretKey);
            String decrypted = decrypt(encrypted, secretKey);

            System.out.println("\nEncrypted Message: " + encrypted);
            System.out.println("Decrypted Message: " + decrypted);

        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

```