**PROGRAM:**

```java
import javax.swing.*;
import javax.crypto.Cipher;
import javax.crypto.KeyGenerator;
import javax.crypto.SecretKey;
import javax.crypto.spec.SecretKeySpec;
import java.security.SecureRandom;
import java.util.Random;

class DES {
    byte[] skey = new byte[1000];
    String skeystring;
    static byte[] raw;
    String inputmessage, encryptedata, decryptedmessage;

    public DES() {
        try {
            generatesymmetrickey();
            inputmessage = JOptionPane.showInputDialog(null, "Enter message to encrypt:");
            byte[] ibyte = inputmessage.getBytes();
            byte[] ebyte = encrypt(raw, ibyte);
            String encrypteddata = new String(ebyte);
            System.out.println("Encrypted message:" + encrypteddata);
            JOptionPane.showMessageDialog(null, "Encrypted Data" + "\n" + encrypteddata);
            byte[] dbyte = decrypt(raw, ebyte);
            String decryptedmessage = new String(dbyte);
            System.out.println("Decrypted message:" + decryptedmessage);
            JOptionPane.showMessageDialog(null, "Decrypted Data " + "\n" +
decryptedmessage);
        } catch (Exception e) {
            System.out.println(e);
        }
    }

    void generatesymmetrickey() {
        try {
            Random r = new Random();
            int num = r.nextInt(10000);
            String knum = String.valueOf(num);
            byte[] knumb = knum.getBytes();
            skey = getRawKey(knumb);
            skeystring = new String(skey);
            System.out.println("DES SymmetricKey=" + skeystring);
        } catch (Exception e) {
```

```java
            System.out.println(e);
        }
    }

    private static byte[] getRawKey(byte[] seed) throws Exception {
        KeyGenerator kgen = KeyGenerator.getInstance("DES");
        SecureRandom sr = SecureRandom.getInstance("SHA1PRNG");
        sr.setSeed(seed);
        kgen.init(56, sr);
        SecretKey skey = kgen.generateKey();
        raw = skey.getEncoded();
        return raw;
    }

    private static byte[] encrypt(byte[] raw, byte[] clear) throws Exception {
        SecretKey seckey = new SecretKeySpec(raw, "DES");
        Cipher cipher = Cipher.getInstance("DES");
        cipher.init(Cipher.ENCRYPT_MODE, seckey);
        byte[] encrypted = cipher.doFinal(clear);
        return encrypted;
    }

    private static byte[] decrypt(byte[] raw, byte[] encrypted) throws Exception {
        SecretKey seckey = new SecretKeySpec(raw, "DES");
        Cipher cipher = Cipher.getInstance("DES");
        cipher.init(Cipher.DECRYPT_MODE, seckey);
        byte[] decrypted = cipher.doFinal(encrypted);
        return decrypted;
    }

    public static void main(String args[]) {
        DES des = new DES();
    }
}
```

# OUTPUT:

```
C:\Windows\System32\cmd.e    Settings

Microsoft Windows [Version 10.0.22621.3296]
(c) Microsoft Corporation. All rights reserved.

C:\Java\jdk1.8.0_202>set path=C:\Java\jdk1.8.0_202\bin;

C:\Java\jdk1.8.0_202>javac DES.java

C:\Java\jdk1.8.0_202>java DES
DES SymmetricKey==dÖ¡?¾n
```

**Input**

? Enter message to encrypt:

`balabharathy`

[ OK ]   [ Cancel ]

```
C:\Windows\System32\cmd.e    Settings

Microsoft Windows [Version 10.0.22621.3296]
(c) Microsoft Corporation. All rights reserved.

C:\Java\jdk1.8.0_202>set path=C:\Java\jdk1.8.0_202\bin;

C:\Java\jdk1.8.0_202>javac DES.java

C:\Java\jdk1.8.0_202>java DES
DES SymmetricKey==dÖ¡?¾n
Encrypted message:?e÷???\z=???ƒd7w
```

**Message**

ℹ **Encrypted Data**
?e÷'‹‡\z=Ãø§ƒd7w

[ OK ]

```
C:\Windows\System32\cmd.e    Settings

Microsoft Windows [Version 10.0.22621.3296]
(c) Microsoft Corporation. All rights reserved.

C:\Java\jdk1.8.0_202>set path=C:\Java\jdk1.8.0_202\bin;

C:\Java\jdk1.8.0_202>javac DES.java

C:\Java\jdk1.8.0_202>java DES
DES SymmetricKey==dÖ¡?¾n
Encrypted message:?e÷???\z=???ƒd7w
Decrypted message:balabharathy
```

**Message**

ℹ **Decrypted Data**
**balabharathy**

[ OK ]