

crypto.java

```
System.out.println("Encrypted message: " + encryptedData);

JOptionPane.showMessageDialog(null, "Encrypted Data\n" + encryptedData);

byte[] decryptedBytes = decrypt(key, encryptedBytes);
decryptedMessage = new String(decryptedBytes, StandardCharsets.UTF_8);
System.out.println("Decrypted message: " + decryptedMessage);

JOptionPane.showMessageDialog(null, "Decrypted Data\n" + decryptedMessage);
} catch (Exception e) {
    System.out.println("Error: " + e.getMessage());
}
}

private void generateSymmetricKey() {
    try {
        KeyGenerator kgen = KeyGenerator.getInstance("DES");
        SecureRandom sr = new SecureRandom();
        kgen.init(56, sr);
        SecretKey key = kgen.generateKey();
        key = skey.getEncoded();
        System.out.println("DES Symmetric key: " +
Base64.getEncoder().encodeToString(key));
    } catch (Exception e) {
        System.out.println("Error generating symmetric key: " + e.getMessage());
    }
}

private byte[] encrypt(byte[] key, byte[] clear) throws Exception {
    SecretKeySpec keySpec = new SecretKeySpec(key, "DES");
    Cipher cipher = Cipher.getInstance("DES");
    cipher.init(Cipher.ENCRYPT_MODE, keySpec);
    return cipher.doFinal(clear);
}

private byte[] decrypt(byte[] key, byte[] encrypted) throws Exception {
    SecretKeySpec keySpec = new SecretKeySpec(key, "DES");
    Cipher cipher = Cipher.getInstance("DES");
    cipher.init(Cipher.DECRYPT_MODE, keySpec);
    return cipher.doFinal(encrypted);
}

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

Ln 61, Col 32 2,563 characters 100% Windows (CRLF) UTF-8

C:\Windows\System32\cmd.e

C:\Users\termi\OneDrive\JAVA>javac cryp-6.java
cryp-6.java:10: error: class DES is public, should be declared in a file nam
ed DES.java
public class DES {
 ^
1 error

C:\Users\termi\OneDrive\JAVA>javac cryp-6.java
cryp-6.java:10: error: '{' expected
public class cryp-6 {
 ^
cryp-6.java:14: error: invalid method declaration; return type required
 public DES() {
 ^
2 errors

C:\Users\termi\OneDrive\JAVA>javac cryp-6.java
cryp-6.java:9: error: '{' expected
 cryp-6 {
 ^
cryp-6.java:14: error: <identifier> expected
 cryp-6() {
 ^
2 errors

C:\Users\termi\OneDrive\JAVA>javac crypto.java
crypto.java:70: error: cannot find symbol
 DES des = new DES();
 ^
symbol: class DES
location: class crypto

crypto.java:70: error: cannot find symbol
 DES des = new DES();
 ^
symbol: class DES
location: class crypto
2 errors

C:\Users\termi\OneDrive\JAVA>javac crypto.java

C:\Users\termi\OneDrive\JAVA>java crypto
DES Symmetric key: PrPQA1rhjE=

Input

Enter message to encrypt

COMPUTER

OK Cancel

crypto.java

```
System.out.println("Encrypted message: " + encryptedData);

JOptionPane.showMessageDialog(null, "Encrypted Data\n" + encryptedData);

byte[] decryptedBytes = decrypt(key, encryptedBytes);
decryptedMessage = new String(decryptedBytes, StandardCharsets.UTF_8);
System.out.println("Decrypted message: " + decryptedMessage);

JOptionPane.showMessageDialog(null, "Decrypted Data\n" + decryptedMessage);
} catch (Exception e) {
    System.out.println("Error: " + e.getMessage());
}
}

private void generateSymmetricKey() {
    try {
        KeyGenerator kgen = KeyGenerator.getInstance("DES");
        SecureRandom sr = new SecureRandom();
        kgen.init(56, sr);
        SecretKey skey = kgen.generateKey();
        key = skey.getEncoded();
        System.out.println("DES Symmetric key: " +
Base64.getEncoder().encodeToString(key));
    } catch (Exception e) {
        System.out.println("Error generating symmetric key: " + e.getMessage());
    }
}

private byte[] encrypt(byte[] key, byte[] clear) throws Exception {
    SecretKeySpec skeySpec = new SecretKeySpec(key, "DES");
    Cipher cipher = Cipher.getInstance("DES");
    cipher.init(Cipher.ENCRYPT_MODE, skeySpec);
    return cipher.doFinal(clear);
}

private byte[] decrypt(byte[] key, byte[] encrypted) throws Exception {
    SecretKeySpec skeySpec = new SecretKeySpec(key, "DES");
    Cipher cipher = Cipher.getInstance("DES");
    cipher.init(Cipher.DECRYPT_MODE, skeySpec);
    return cipher.doFinal(encrypted);
}

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

Ln 61, Col 32 2,563 characters 100% Windows (CRLF) UTF-8

C:\Windows\System32\cmd.e

```
crp-6.java:10: error: class DES is public, should be declared in a file nam
ed DES.java
public class DES {
    ^
1 error

C:\Users\termi\OneDrive\JAVA>javac crp-6.java
crp-6.java:10: error: '{' expected
public class crp-6 {
    ^
crp-6.java:14: error: invalid method declaration; return type required
public DES() {
    ^
2 errors


C:\Users\termi\OneDrive\JAVA>javac crp-6.java
crp-6.java:10: error: '{' expected
public class crp-6 {
    ^
crp-6.java:14: error: <identifier> expected
public DES() {
    ^
2 errors

C:\Users\termi\OneDrive\JAVA>javac crypto.java
crypto.java:70: error: cannot find symbol
    DES des = new DES();
    ^
symbol:   class DES
location: class crypto
crypto.java:70: error: cannot find symbol
    DES des = new DES();
    ^
symbol:   class DES
location: class crypto
2 errors

C:\Users\termi\OneDrive\JAVA>javac crypto.java

C:\Users\termi\OneDrive\JAVA>java crypto
DES Symmetric key: PrPQAl1rhjE=
Encrypted message: PZ3Wuw2HryHMqt0Jw317Aw==
```

Message

 Encrypted Data
PZ3Wuw2HryHMqt0Jw317Aw==

OK

crypto.java

```
System.out.println("Encrypted message: " + encryptedData);

JOptionPane.showMessageDialog(null, "Encrypted Data\n" + encryptedData);

byte[] decryptedBytes = decrypt(key, encryptedBytes);
decryptedMessage = new String(decryptedBytes, StandardCharsets.UTF_8);
System.out.println("Decrypted message: " + decryptedMessage);

JOptionPane.showMessageDialog(null, "Decrypted Data\n" + decryptedMessage);
} catch (Exception e) {
    System.out.println("Error: " + e.getMessage());
}
}

private void generateSymmetricKey() {
    try {
        KeyGenerator kgen = KeyGenerator.getInstance("DES");
        SecureRandom sr = new SecureRandom();
        kgen.init(56, sr);
        SecretKey skey = kgen.generateKey();
        key = skey.getEncoded();
        System.out.println("DES Symmetric key: " +
Base64.getEncoder().encodeToString(key));
    } catch (Exception e) {
        System.out.println("Error generating symmetric key: " + e.getMessage());
    }
}

private byte[] encrypt(byte[] key, byte[] clear) throws Exception {
    SecretKeySpec skeySpec = new SecretKeySpec(key, "DES");
    Cipher cipher = Cipher.getInstance("DES");
    cipher.init(Cipher.ENCRYPT_MODE, skeySpec);
    return cipher.doFinal(clear);
}

private byte[] decrypt(byte[] key, byte[] encrypted) throws Exception {
    SecretKeySpec skeySpec = new SecretKeySpec(key, "DES");
    Cipher cipher = Cipher.getInstance("DES");
    cipher.init(Cipher.DECRYPT_MODE, skeySpec);
    return cipher.doFinal(encrypted);
}

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

Ln 61, Col 32 2,563 characters 100% Windows (CRLF) UTF-8

C:\Windows\System32\cmd.e

```
ed DES.java
public class DES {
    ^
1 error

C:\Users\termi\OneDrive\JAVA>javac cryp-6.java
cryp-6.java:10: error: '{' expected
public class cryp-6 {
    ^
cryp-6.java:14: error: invalid method declaration; return type required
public DES() {
    ^
2 errors


C:\Users\termi\OneDrive\JAVA>javac cryp-6.java
cryp-6.java:10: error: '{' expected
public class cryp-6 {
    ^
cryp-6.java:14: error: <identifier> expected
    cryp-6() {
        ^
2 errors

C:\Users\termi\OneDrive\JAVA>javac crypto.java
crypto.java:70: error: cannot find symbol
    DES des = new DES();
    ^
symbol:   class DES
location: class crypto
crypto.java:70: error: cannot find symbol
    DES des = new DES();
    ^
symbol:   class DES
location: class crypto
2 errors

C:\Users\termi\OneDrive\JAVA>javac crypto.java

C:\Users\termi\OneDrive\JAVA>java crypto
DES Symmetric key: PrPQAl1rhjE=
Encrypted message: PZ3Wuw2HryHMqt0Jw317Aw==
Decrypted message: COMPUTER
```

Message

 Decrypted Data
COMPUTER

OK