

```
1 import javax.crypto.Cipher;
2 import javax.crypto.KeyGenerator;
3 import javax.crypto.SecretKey;
4 import javax.xml.bind.DatatypeConverter;
5
6 public class AESEncryption {
7
8     public static void main(String[] args) throws Exception {
9         String plainText = "Hello World";
10        SecretKey secKey = getSecretEncryptionKey();
11        byte[] cipherText = encryptText(plainText, secKey);
12        String decryptedText = decryptText(cipherText, secKey);
13
14        System.out.println("Original Text:" + plainText);
15        System.out.println("AES Key (Hex Form):"+bytesToHex(secKey.getEncoded()));
16        System.out.println("Encrypted Text (Hex Form):"+bytesToHex(cipherText));
17        System.out.println("Decrypted Text:"+decryptedText);
18    }
19
20
21    public static SecretKey getSecretEncryptionKey() throws Exception{
22        KeyGenerator generator = KeyGenerator.getInstance("AES");
23        generator.init(128); // The AES key size in number of bits
24        SecretKey secKey = generator.generateKey();
25        return secKey;
26    }
27
28
29    public static byte[] encryptText(String plainText, SecretKey secKey) throws Exception{
30
31        Cipher aesCipher = Cipher.getInstance("AES");
32        aesCipher.init(Cipher.ENCRYPT_MODE, secKey);
33        byte[] byteCipherText = aesCipher.doFinal(plainText.getBytes());
34        return byteCipherText;
35    }
36
37
38    public static String decryptText(byte[] byteCipherText, SecretKey secKey) throws Exception{
39
40        Cipher aesCipher = Cipher.getInstance("AES");
41        aesCipher.init(Cipher.DECRYPT_MODE, secKey);
42        byte[] bytePlainText = aesCipher.doFinal(byteCipherText);
43        return new String(bytePlainText);
44    }
45
46
47    private static String bytesToHex(byte[] hash) {
48        return DatatypeConverter.printHexBinary(hash);
49    }
50 }
```