Advanced Encryption Standard

Aim:

To implement Advanced Encryption Standard (AES) Algorithm.

1 Key Generation:

The Key interface represents keys for cryptographic operations. Keys are opaque containers that hold an encoded key, the key's encoding format, and its cryptographic algorithm.

SecretKey secretKey = new SecretKeySpec(keyBytes, "AES");

```
1. key = myKey.getBytes("UTF-8");
2.              System.out.println(key.length);
3.              sha = MessageDigest.getInstance("SHA-1");
4.              key = sha.digest(key);
5.           key = Arrays.copyOf(key, 16); // use only first 128 bit
6.           System.out.println(key.length);
```

To generate 128 bit keys

2 Encryption:

The doFinal() method to perform the encryption or decryption operation.

Get the cipher Instance

Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");

where it use AES in Electronic Code book and it uses publickey cryptography system padding.

SecretKey secretKey = new SecretKeySpec(keyBytes, "AES");

cipher.init(Cipher.ENCRYPT_MODE, secretKey);

cipher.doFinal(message)

3 Decryption:

ENCRYPT_MODE: initialize cipher object to encryption mode

DECRYPT_MODE: initialize cipher object to decryption mode

Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");

SecretKey secretKey = new SecretKeySpec(keyBytes, "AES");

```
 cipher.init(Cipher.DECRYPT_MODE, secretKey);

return cipher.doFinal(encryptedMessage);
```