Two class inherit one interface

```
/*
C#: The Complete Reference
by Herbert Schildt

Publisher: Osborne/McGraw-Hill (March 8, 2002)
ISBN: 0072134852
*/
using System;

// An encryption interface.
public interface ICipher {
string encode(string str);
string decode(string str);
}

/* A simple implementation of ICipher that codes
a message by shifting each character 1 position
higher. Thus, A becomes B, and so on. */
class SimpleCipher : ICipher {

// Return an encoded string given plaintext.
public string encode(string str) {
string ciphertext = “”;

for(int i=0; i < str.Length; i++) ciphertext = ciphertext +
(char) (str[i] + 1); return ciphertext; } // Return an decoded
string given ciphertext. public string decode(string str) {
string plaintext = ""; for(int i=0; i < str.Length; i++)
plaintext = plaintext + (char) (str[i] - 1); return plaintext;
} } /* This implementation of ICipher uses bit manipulations
and key. */ class BitCipher : ICipher { ushort key; // Specify
a key when constructing BitCiphers. public BitCipher(ushort k)
{ key = k; } // Return an encoded string given plaintext.
public string encode(string str) { string ciphertext = ""; for(int i=0; i < str.Length; i++) ciphertext = ciphertext + (char) (str[i] ^ key); return ciphertext; } // Return an decoded string given ciphertext. public string decode(string str) { string plaintext = ""; for(int i=0; i < str.Length; i++) plaintext = plaintext + (char) (str[i] ^ key); return plaintext; } } // Demonstrate ICipher. public class ICipherDemo { public static void Main() { ICipher ciphRef; BitCipher bit = new BitCipher(27); SimpleCipher sc = new SimpleCipher(); string plain; string coded; // first, ciphRef refers to the simple cipher ciphRef = sc; Console.WriteLine("Using simple cipher."); plain = "testing"; coded = ciphRef.encode(plain); Console.WriteLine("Cipher text: " + coded); plain = ciphRef.decode(coded); Console.WriteLine("Plain text: " + plain); // now, let ciphRef refer to the bitwise cipher ciphRef = bit; Console.WriteLine(" Using bitwise cipher."); plain = "testing"; coded = ciphRef.encode(plain); Console.WriteLine("Cipher text: " + coded); plain = ciphRef.decode(coded); Console.WriteLine("Plain text: " + plain); } } [/csharp]
```