Introduction to the Course
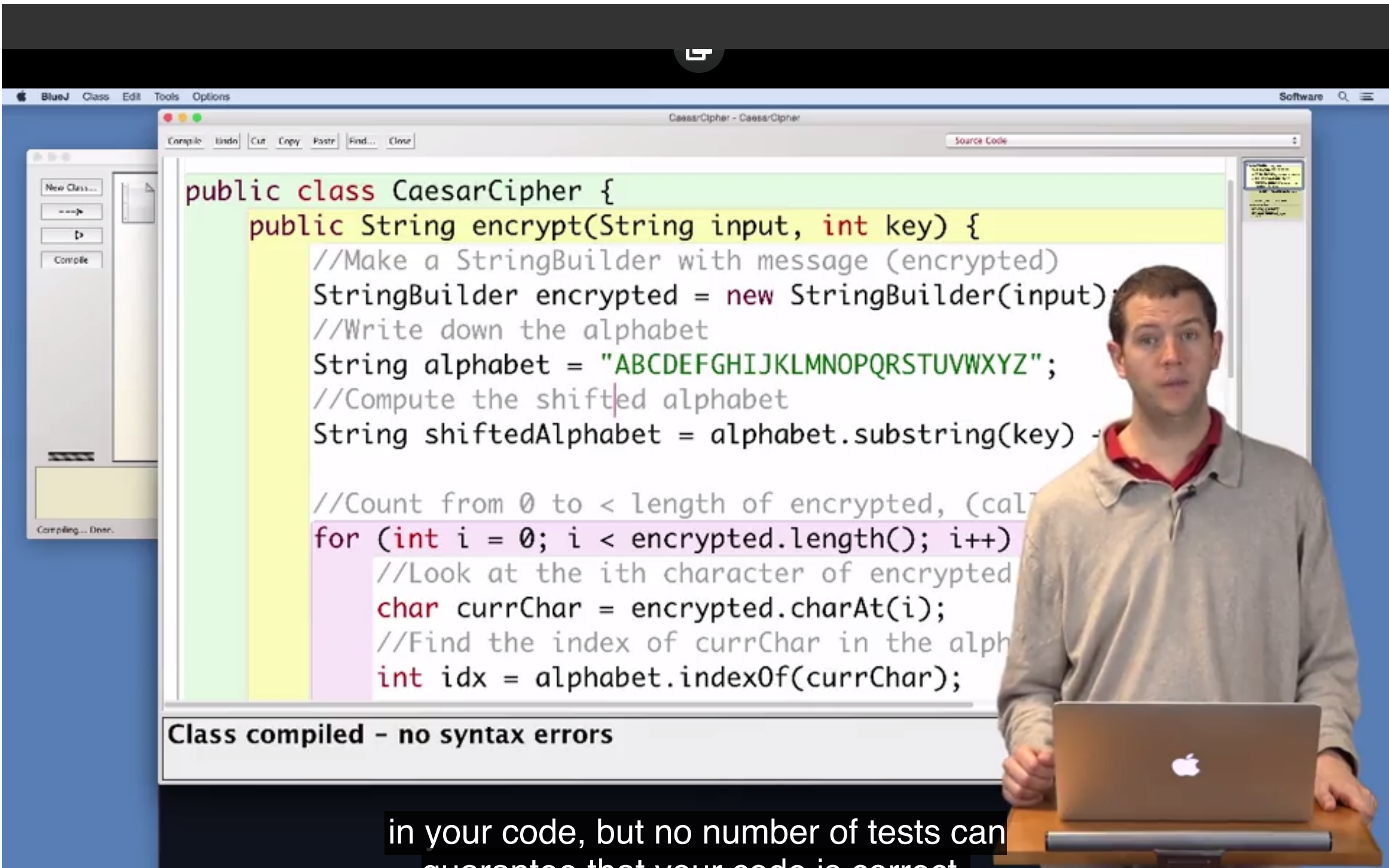
Implementing the Caesar Cipher

Breaking the Caesar Cipher

Object Oriented Caesar Cipher

Review


```java
public class CaesarCipher {
    public String encrypt(String input, int key) {
        //Make a StringBuilder with message (encrypted)
        StringBuilder encrypted = new StringBuilder(input);
        //Write down the alphabet
        String alphabet = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
        //Compute the shifted alphabet
        String shiftedAlphabet = alphabet.substring(key)

        //Count from 0 to < length of encrypted, (cal
        for (int i = 0; i < encrypted.length(); i++)
            //Look at the ith character of encrypted
            char currChar = encrypted.charAt(i);
            //Find the index of currChar in the alph
            int idx = alphabet.indexOf(currChar);
```

Class compiled - no syntax errors

in your code, but no number of tests can guarantee that your code is correct.

Testing and Debugging

👍 👎 🚩

Have a question? Discuss this lecture in the week forums. ›

Downloads

Lecture Video mp4

Subtitles (English) WebVTT

Transcript (English) txt

Would you like to help us translate the transcript and subtitles into additional languages?

Interactive Transcript

Search Transcript | English ▾

0:03
Now, we've written and tested our CaesarCipher class. We tested it out on one message and it looked like it worked. Does this mean we can be sure our code is correct? Of course not. Remember, when you're testing software each test case makes you more confident in your code, but no number of tests can guarantee that your code is correct. You want more and more tests to be more and more confident and one test case generally isn't enough.

0:30
So, here I have a different method. Dear Owen, no matter what you may have heard, thee's no cake in the conference room. The cake is a lie. Please keep working on Coursera videos. So I wanna encrypt this, and send an encrypted message to Owen. To fend him off from the cake in case he intercepted my message to Robert.

0:50
I'm going to use my test Caesar. And now I'm going to do message2.txt. And when I look to at this, I'm going to see that the D in dear turned into a U. That's good. But then e, a, and r were left the same. Then the O in Owen turned into an F. But then w, e, n were all left the same. And in fact, most of this is unchanged.

1:16
So my code isn't quite right. If I look back at it, I'm gonna see that it's only going to work with capital letters and not with lower case letters. This was the problem in our first test case. It only had capital letters. It didn't have lower case letters, so we didn't really test all the cases enough. We're gonna leave this to you to fix. There are a couple of different ways you can do it, and we hope you can figure one of them out. So good luck, and happy fixing of your code.