JavaScript: Inserting CRLF into strings

Those familiar with Visual Basic or VB Script have probably seen the vbCrLf constant used to put line breaks into strings:

```
Dim str
str = "Line 1" & vbCrLf & "Line 2"
```

The two control characters carriage return (CR) and line feed (LF) together represent a line break inside strings. In JavaScript you'll use the `"\r"` and `"\n"` *escape characters* to achieve the same thing:

```
var str;
str = "Line 1\r\nLine 2";
```

These are character sequences placed *inside* string literals, escaping means they represent control characters rather than the literal characters you see in the code. If you had two variables that you wanted to separate a CRLF, you would have to place these characters in a string of their own:

```
var line1 = "Line 1";
var line2 = "Line 2";
var str;
str = line1 + "\r\n" + line2;
```

Carriage return (`"\r"` or CR) and line feed (`"\n"` or LF) are not the only escape characters in JavaScript. You might also want to insert a tab (`"\t"`), a quote (`"\""`), or even just a backslash (`"\\"`).

CRLF not the only line break

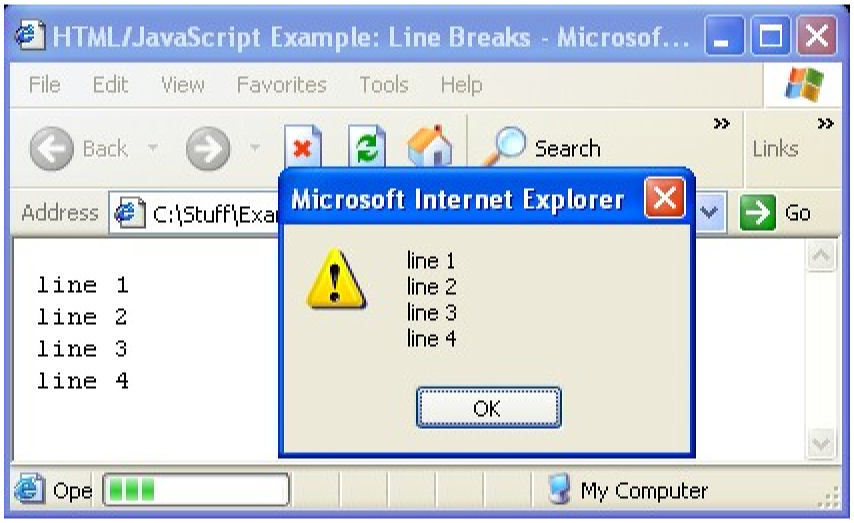
If I don't put this section in here, someone's going to call me on it. It's true that a CRLF sequence isn't the only way to represent a line break. Depending on the situation, you may just use one or the other. Plain text files on Windows use CRLF for line breaks, but Macs use CR, and most other systems (Unix, Linux, BSD, etc.) use just LF.

It's irritating to say the least, but thankfully most programs are forgiving of any of these choices these days (except NotePad - argh!). You may have noticed FTP programs having an option to transfer as binary or *text*. Binary will leave the file bytes alone, but text mode will cause the line breaks to be normalized based on the target system. So if you upload a Windows text file to a Linux server using text mode, it will translate all CRLF's to just LF.

When you're working with client-side JavaScript in a web browser, it's safe to use any of the three line break sequences (CRLF, LR, or CR) as demonstrated in "this example" (http://www.gibdon.com/Examples/HTML/ex-javascript-line-break.html):

```
<PRE>
<SCRIPT LANGUAGE="JavaScript"><!--
    document.write("line 1");
    document.write("\n");
    document.write("line 2");
    document.write("\r\n");
    document.write("line 3");
    document.write("\r");
    document.write("line 4");
    alert("line 1\nline 2\r\nline 3\rline 4");
//--></SCRIPT>
</PRE>
```

The above example should output four lines right on top of each other, both in the browser window and a message box.


HTML/JavaScript Example: Line Breaks - Microsof...
File Edit View Favorites Tools Help
Back
Search
Links
Address C:\Stuff\Exa
Go
Microsoft Internet Explorer
OK
Ope
My Computer

```
line 1
line 2
line 3
line 4
```

```
line 1
line 2
line 3
line 4
```

While you can combine different escape character sequences for line breaks, I suggest sticking to one. In fact, I recommend just using LF (`"\n"`) for client-side JavaScript in a web browser.