

Re: Trim a multiple line message to a single line

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2007-08/msg00314.html>

- *From:* Jesse Houwing <Jesse.houwing@xxxxxxxxxxxxxxxxxxxx>
 - *Date:* Thu, 2 Aug 2007 21:01:29 +0000 (UTC)
-

Hello Brian,

You can do this quite easily with either a regex or a simple function I'll try to demonstrate both:

basically you need a regex that matches the `\r\n[^\r\n]+\r\n` that is followed by two spaces and replace it with a space

so the regex looks like this (I simplified it a bit, but given the fact that all the lines are very well layed out I don't think you have to make it more difficult :).

```
\r\n[^\r\n]+\r\n [ ]{2}
```

now replace this by " " and you're all set.

So:

```
private static Regex _rx = new Regex(@"\r\n[^\r\n]+\r\n [ ]{2}", RegexOptions.Compiled);
```

```
private string LayoutInput(string input)
{
    return _rx.Replace(input, " ");
}
```

Or you could use a `StringReader/StringBuilder` combination:

```
private string LayoutInput(string input)
{
    StringReader sr = new StringReader(input);
    StringBuilder sb = new StringBuilder(input.Length);
    string line;
    while ((line = sr.ReadLine()) != null)
    {
        if (line.Length < 29) { throw new InvalidOperationException("invalid input"); /* or simply continue; */ }
        if (line[29] != ' ')
        {
            if (sb.Length > 0)
            {
                sb.Append("\r\n");
            }
        }
    }
}
```

Re: Trim a multiple line message to a single line

```
sb.Append(line);
}
else
{
sb.Append(line.Substring(30));
}
}
return sb.ToString();
}
```

I haven't used much error checking to make sure the format is correct. But I think you can go on from here.

Jesse

Sent you an email with a sampling.

"Jesse Houwing" wrote:

Hello Brian,

Could you send a sample file with two of these data blocks f what they should look like and how you receive them directly to my email? I'll reply back to this thread with a solution when I have time to go over it. My newsreader keeps messing up your samples, so it would be easier if I had two text files.

Jesse

Jesse the data inside of the [] is static six digits for each line.

The total number of lines may vary anywhere from one to as much as 8 lines of data.

The line should look like this when pieced together;

```
2007/07/27 11:00:03 [153006]ARES_INDICATION
010.050.016.002
404.2.01
(6511)
RX 74 bytes 65 11 26 02 BC 6C AA 20 76 93 51 53 50 76
13 48 52 00
52
02 02
C7 83 D7 07 07 1B 0B 00 00 00 00 28 0A 06 06 06 06 06 06
06 06 06 06
06 0A 06
06 06 0A 06 06 06 0A 06 06 06 06 06 06 06 06 06 06 06
```

Re: Trim a multiple line message to a single line

06 06 06
06
06 06 06 40

All in one big line without word wrap.

The next entry would be a different header
(ARES_INDICATION for
example).

"Jesse Houwing" wrote:

Hello Brian,

2007/07/27 11:00:03
[153006]

My guess is that these aren't the same length
for every line. If
they were it would be easy:

```
regex = "^.{num}";  
input = "...";  
replacement = "";  
result = Regex.Replace(regex, input,  
replacement,  
RegexOptions.MultiLine);  
If the part between [] is variable in length  
you'd have to choose a  
more advanced way:  
^[^[]+\[ [^\]]+\]
```

It's hard to read. But essentially it says:
^ start at the beginning of a line
[^[]+ get everything that is not a [
\[get the opening [
[^\]]+ get everything that is not a]
\[get the closing]
and replace that with an empty string.
should do the job.

Another option is to construct a
stringbuilder, readline every line
of the original input, use substring of the
first] to chop of the
first part and add the resulting string to the
stringbuilder.

The first solution (regex) is shorter and once

Re: Trim a multiple line message to a single line

you understand the
regex easy
to understand and maintain.
The second solution (stringbuilder) is a bit
more complex in lines
of
code,
easier to understand if you're not used to
regex and probably a tad
faster.
Jesse