

How can i extract the info between " " ? C#

Asked 11 years, 5 months ago Modified 11 years, 5 months ago Viewed 312 times



Can anyone tell me how to extract the following info to a variable in a simple way from this output.

-5

Output:



```
AT+CMGL="ALL"
+CMGL: 0,"REC READ", "+40728072005", "12/06/29,13:04:26+12"
password,1,ON
```



i want to extract from some info from the above output like below into variables:

```
msisdn="+40728072005"
passwd="password"
itemno="1"
command="ON"
```

--

```
string split = ",";
string[] substrings = Regex.Split(buff, split);
foreach (string match in substrings)
{
    Console.WriteLine("{0}", match);
}
```

image -> [1]: <https://i.stack.imgur.com/zl6Nj.png>

Is there a way to exclude the unwanted output from the start? like a grep -v, i am not that familiar with C#.

So far i got to this point:

```
string split = ",";
string split1 = "\"";
string split2 = "OK";
string[] substrings = Regex.Split(buff, split);
string[] substrings1 = Regex.Split(substrings[2], split1);
Console.WriteLine(substrings1[1]);
string[] substrings2 = Regex.Split(substrings[5], split1);

Console.WriteLine(substrings2[1]);
Console.WriteLine(substrings[6]);
string[] substrings3 = Regex.Split(substrings[7], split2);
Console.WriteLine(substrings3[0]);
```




```
foreach (Match m in matches)
{
    Console.WriteLine(m.Groups["msisdn"].Value);
    Console.WriteLine(m.Groups["date"].Value);
    Console.WriteLine(m.Groups["passwd"].Value);
    Console.WriteLine(m.Groups["itemno"].Value);
    Console.WriteLine(m.Groups["command"].Value);
}
Console.ReadKey();
```

Basically, regular expression searches for a defined pattern inside text. Meaning of the expression:

\"" - that's how a quote sign is defined, it is escaped with backslash and written twice so it doesn't cause compiler error.

(?+\d*) - generally, bracketed expression means that you are capturing a group, this here is a named group and it's name is specified by ?, this group is going to capture string that begins with + sign (it is escaped by backslash) followed by numbers which are represented by backslash and letter "d" ("d" comes from "digit"). Backslash is used as an escape character for characters with special meaning. Asterisk * means that character that comes before it can occur 0 or more times.

\"" - quote, this is closing quote for "msisdn" number.

, - means comma.

. * - dot means any character, and asterisk that follows it means that any character can occur 0 or more times.

(?\d{2}/\d{2}/\d{2}:\d{2}:\d{2}+\d{2}) - named group that captures date, expression \d{2} has \d which means digit and {2} which tells regex engine that digit should occur two times, then comes / - it means forwardslash (it is escaped by a backslash). Afterwards come digits separated by colons and in the end plus sign and two digits after it, that's how the date is extracted.

\n+ - \n means newline, + is a quantifier which means that newline appears one or more times.

(?[^,]*) - is a named group that captures password part, [^,] means that any character except comma should be captured, square brackets mean range of characters, ^ means negation, asterisk means that character can occur 0 or more times.

(?\d*) - is a named group that captures numbers as defined by \d*

(?\w*) - is a named group that captures word characters (letters, digits, and underscores)

Share Improve this answer Follow

edited Jun 29, 2012 at 17:40

answered Jun 29, 2012 at 11:02



Ivan Golović


8,752 3 25 31

Hi, it returns 5 blank lines, the output contains also some empty lines and OK at the end, like in here:

i.stack.imgur.com/zl6Nj.png – Alexis Cimpu Jun 29, 2012 at 11:08

Hi, I changed it, you can try it now – Ivan Golović Jun 29, 2012 at 11:31

I'm glad to help, is it giving you the output that you needed? I also changed Singleline to Multiline mode so that it returns all occurrences of such pattern (not only first one that was found). – Ivan Golović Jun 29, 2012 at 11:49

Ivan can you please help me with another REGEX or better explain me how is this one done so i can modify it and make another one :D. – Alexis Cimpu Jun 29, 2012 at 15:26 

Sure, I'll add more comments to the answer. Best thing would be to go through one of the tutorials about regular expressions – Ivan Golović Jun 29, 2012 at 17:20
