

# Re: Regex question

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2008-12/msg02607.html>

---

- *From:* Jesse Houwing <[jesse.houwing@xxxxxxxxxxxxxxxxxxxxx](mailto:jesse.houwing@xxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Tue, 30 Dec 2008 14:04:32 +0000 (UTC)
- 

Hello tshad,

I have a text string that has a date in it and I am trying to pull out the date. In my earlier posts I got a couple of patterns that seem correct, but I can't seem to get them to work. I want to account for 1 or 2 digits in the month and day fields and 2 or 4 in the year field.

After not getting the correct result I put it down to the simplest pattern which would match exactly the number of digits and I get nothing.

For example, if I have the following string:

```
valueIn = "At 02/23/2007 DOM";
```

I want to end up with 02/23/2007.

If I use:

```
Match oMatch;  
oMatch = Regex.Match(valueIn, @"\d{2}^\d{2}^\d{4}");  
I get nothing from my oMatch.Groups[1].Value.
```

If I have:

```
oMatch = Regex.Match(valueIn, @"\d{1,2}^\d{1,2}^\d{2,4}");  
strValue = oMatch.Groups[1].Value;  
I get nothing from my oMatch.Groups[1].Value;
```

If I have:

```
oMatch = Regex.Match(valueIn,  
@"\d{1,2}(-|/)\d{1,2}(-|/)(\d{2}|\d{4})");  
strValue = oMatch.Groups[1].Value;  
oMatch.Groups[1].Value = "/"  
oMatch.Groups[2].Value = "/"
```

## Re: Regex question

`oMatch.Groups[3].Value = "20"`  
What is happening here?

I would have expected the 1st one to work which I assume says look for a pattern of: 2 digits, a slash, 2 more digits, another slash and 4 digits which is in this string.

Not sure why it found the slashes and the 20 in the last set up.

Tom, this is due to the fact that `()` create groups. So if your regex contains `()` it will automatically create grouped values in your match object.

In your case there is no need to use groups at all, so you can change your code to:

```
if (oMatch.Success)
{
    string unparsedDate = oMatch.Value;
}
```

Alternatively, if you want to use groups, or want to learn more about groups:

use this regex  
`@"(\d{1,2})[-/](\d{1,2})[-/](\d{2}|\d{4})"`;

Now `oMatch.Groups[1].Value` will contain the month

Now `oMatch.Groups[2].Value` will contain the day

Now `oMatch.Groups[3].Value` will contain the year

Also note that I changed `(-|/)` to `[-/]`, so that it doesn't give me more groups than I actually need...

—

Jesse Houwing  
jesse.houwing at sogeti.nl

.