

CSIS 4135

Regular Expressions

Regular Expressions

Consist of two types of characters

- Literals
An exact character to be matched in target string
- Metacharacters
Symbol with special meaning to parser

Example: `A\d{5}B`

Matches strings with A followed by exactly 5 digits followed by B

Regular Expressions

- Five digit zip code: `[0-9]{5}`
- Zip + 4: `[0-9]{5}-[0-9]{4}`
- Either: `[0-9]{5}|[0-9]{5}-[0-9]{4}`
- Dates: `\d{1,2}/\d{1,2}/\d{4}`
- Capture the year:
`\d{1,2}/\d{1,2}/(\d{4})`
- Metacharacter reference:
<http://weblogs.asp.net/roshero/rover/articles/6863.aspx>

Options and Repetitions

- | | |
|---------------------|----------------------|
| • <code>ca?t</code> | optional a |
| • <code>ca*t</code> | 0 or more a's |
| • <code>ca+t</code> | 1 or more a's |
| • <code>c.t</code> | any single character |
| • <code>c.*t</code> | 0 or more characters |

Whole Lines and Anchors

- Match a line starting with Yes:
`^Yes`
- Match a line starting with Yes and ending with No
`^Yes.*No$`
- Match a line containing only digits:
`^\d+$`

Regular Expressions in C#

- Import:
`using System.Text.RegularExpressions;`

```
Regex pattern1 = new Regex("[0-9]+");
if (pattern1.IsMatch("123456"))
    Response.Write("Found a number");
```

Found a number

Regular Expressions in C#

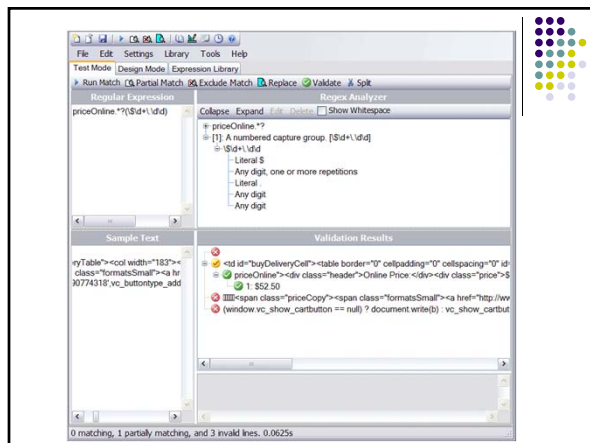
```
Match m1 = pattern1.Match("zzzzz123abc");
if (m1.Success)
    Response.Write(
        "Contains a number starting at: " + m1.Index);
```

Contains a number starting at: 5

Regular Expressions in C#

```
Regex pattern2 = new Regex("ab([0-9]+)xy");
Match m2 = pattern2.Match("xxxab123xyz");
foreach (Group match in m2.Groups)
{
    Response.Write(match + <br />);
}
```

ab123xy
123



Accessing Other Web Sites

```
using System.Net;
using System.IO;
```

```
WebRequest request = WebRequest.Create(url);
HttpWebResponse response =
    (HttpWebResponse)request.GetResponse();
Stream dataStream = response.GetResponseStream();
StreamReader reader = new StreamReader(dataStream);
string responseFromServer = reader.ReadToEnd();
reader.Close();
response.Close();
// Process the content of the response
```