

# More Regular Expressions

April 9, 2013

# Today

- Using match iterators
- More special characters
- Match groups

# Good News!



# Good News!



# Good News!



# Calendar

| Sun  | Mon  | Tues   | Wed   | Thurs  | Fri  | Sat  |
|------|------|--|---|--|------|------|
| 4/7  | 4/8  | 4/9  | 4/10  | 4/11   | 4/12 | 4/13 |
|      |      | <b>Today</b>                                     | Project 2 work in progress due to <a href="mailto:cs0931handin@gmail.com">cs0931handin@gmail.com</a> by midnight! | In-class peer review of Project 2 in progress. |      |      |
| 4/14 | 4/15 | 4/16   | 3/27  | 3/28   | 3/29 | 3/30 |
|      |      | <b>Project 2 due.</b><br>Start on final project. |   |  |      |      |

# Match Iterators

- Download `re_tester.py`.

```
def showMatches(pattern, text):
    print('Text: ' + text)
    print('Patt: ' + pattern)
    match_iterator = re.finditer(pattern, text)
    for match in match_iterator:
        print('----- New Match -----')
        print('  match.start() = ' + str(match.start()))
        print('  match.end()   = ' + str(match.end()))
        print('  Whole match: ' + match.group(0))
        if match.lastindex is not None:
            print('  Match groups:')
            for i in range(1, match.lastindex+1):
                print('    match.group(' + str(i) + '): '
                    + match.group(i))
```

| Pattern                    | Description  |
|----------------------------|--|
| <code>\s</code>            | Match a whitespace character (except newline, if you use the "re.MULTILINE" option).                 |
| <code>\w</code>            | Match a letter, number, or underscore character.   |
| <code>\d</code>            | Match a digit.   |
| <code>^</code>             | Matches beginning of string (or beginning of line with the "re.MULTILINE" option).                   |
| <code>\$</code>            | Matches end of string (or end of line with the "re.MULTILINE" option).                               |
| <code>.</code>             | Matches any single character except newline (also including newline with the "re.MULTILINE" option). |
| <code>[aeiou]</code>       | Matches any single character in brackets.  |
| <code>[A-Z]</code>         | Matches any single character in the range indicated in the brackets.                                 |
| <code>[^aeiou]</code>      | Matches any single character not in brackets.  |
| <code>[aeiou]*</code>      | Matches 0 or more occurrences of preceding expression.   |
| <code>[aeiou]+</code>      | Matches 1 or more occurrence of preceding expression.  |
| <code>[aeiou]?</code>      | Matches 0 or 1 occurrence of preceding expression.   |
| <code>[aeiou]{n}</code>    | Matches exactly n number of occurrences of preceding expression.                                     |
| <code>[aeiou]{n,}</code>   | Matches n or more occurrences of preceding expression.   |
| <code>[aeiou]{n, m}</code> | Matches at least n and at most m occurrences of preceding expression.                                |
| <code>a b</code>           | Matches either a or b.   |
| <code>([aeiou]+)</code>    | Groups regular expressions and remembers matched text.   |
| ...                        | ...  |



# Match Iterators

- Download `re_tester.py`.

```
def showMatches(pattern, text):
    print('Text: ' + text)
    print('Patt: ' + pattern)
    match_iterator = re.finditer(pattern, text)
    for match in match_iterator:
        print('----- New Match -----')
        print('  match.start() = ' + str(match.start()))
        print('  match.end()   = ' + str(match.end()))
        print('  Whole match: ' + match.group(0))
        if match.lastindex is not None:
            print('  Match groups:')
            for i in range(1, match.lastindex+1):
                print('    match.group(' + str(i) + '): '
                    + match.group(i))
```

# Let's Investigate...

- Download `zombify.py` and `pnp.txt`.
- Open `pnp.txt` in Notepad++.
- Open `zombify.py` in IDLE.
- Enter this command:

```
zombify_text('pnp.txt', 'pnpnz.txt', 'pnp.csv')
```

- Open `pnpnz.txt` in Notepad++.
- Now let's look at the code.

# Remember:

Project 2 (and the website) are now due on Tuesday, 4/16.

**Submit your work in progress by Wednesday night (4/10) at midnight!**