

Files and Regular Expressions

```
In [1]: import re
        if re.match('a.*b', 'alphabet'):
            print('match found!')
```

match found!

```
In [5]: # match() only matches at beginning of string
        if re.match('l.*b', 'alphabet'):
            print('match found!')
```

```
In [6]: o = re.search('l.*e', 'alphabet')
        if o:
            print('match found!')
```

match found!

```
In [7]: o.re.pattern, o.string
```

```
Out[7]: ('l.*e', 'alphabet')
```

```
In [5]: o.start(), o.end()
```

```
Out[5]: (1, 7)
```

```
In [6]: o.string[o.start():o.end()]
```

```
Out[6]: 'lphabe'
```

```
In [8]: !cat poem.txt
```

```
TWO roads diverged in a yellow wood,
And sorry I could not travel both
And be one traveler, long I stood
And looked down one as far as I could
To where it bent in the undergrowth;
```

```
Then took the other, as just as fair,
And having perhaps the better claim,
Because it was grassy and wanted wear;
Though as for that the passing there
Had worn them really about the same,
```

```
And both that morning equally lay
In leaves no step had trodden black.
Oh, I kept the first for another day!
Yet knowing how way leads on to way,
I doubted if I should ever come back.
```

```
I shall be telling this with a sigh
Somewhere ages and ages hence:
Two roads diverged in a wood, and I-
I took the one less traveled by,
And that has made all the difference.
```

```
In [13]: import re
        linenum = 0
        with open('poem.txt') as poem:
            for line in poem:
                linenum += 1
                if re.search('the', line):
                    print(f'{linenum}: {re.sub('the', '---', line)}", end='')
```

```
5: To where it bent in --- undergrowth;
7: Then took --- o---r, as just as fair,
8: And having perhaps --- better claim,
10: Though as for that --- passing ---re
11: Had worn ---m really about --- same,
15: Oh, I kept --- first for ano---r day!
22: I took --- one less traveled by,
23: And that has made all --- difference.
```

- When opening a file, can set a variable to result of open()
- Requires explicit close() to correctly manage files
- Use the with structure (as outlined above) to automatically close
- Use read() method to read file in its entirety as a string
- Provide a size "hint" to read to absorb file contents in "chunks"

```
In [10]: with open('poem.txt') as file:
        data = file.read()
        print(data)
```

```
TWO roads diverged in a yellow wood,
And sorry I could not travel both
And be one traveler, long I stood
```

```
And looked down one as far as I could
To where it bent in the undergrowth;
```

```
Then took the other, as just as fair,
And having perhaps the better claim,
Because it was grassy and wanted wear;
Though as for that the passing there
Had worn them really about the same,
```

```
And both that morning equally lay
In leaves no step had trodden black.
Oh, I kept the first for another day!
Yet knowing how way leads on to way,
I doubted if I should ever come back.
```

```
I shall be telling this with a sigh
Somewhere ages and ages hence:
Two roads diverged in a wood, and I—
I took the one less traveled by,
And that has made all the difference.
```

```
In [11]: with open('poem.txt') as file:
         while (chunk := file.read(500)):
             print(chunk, end = '')
```

```
TWO roads diverged in a yellow wood,
And sorry I could not travel both
And be one traveler, long I stood
And looked down one as far as I could
To where it bent in the undergrowth;
```

```
Then took the other, as just as fair,
And having perhaps the better claim,
Because it was grassy and wanted wear;
Though as for that the passing there
Had worn them really about the same,
```

```
And both that morning equally lay
In leaves no step had trodden black.
Oh, I kept the first for another day!
Yet knowing how way leads on to way,
I doubted if I should ever come back.
```

```
I shall be telling this with a sigh
Somewhere ages and ages hence:
Two roads diverged in a wood, and I—
I took the one less traveled by,
And that has made all the difference.
```

Exercise One

- Write a Python program to read a file and count the number of occurrences of each word in the file
- Use a dictionary indexed by word, to count the occurrences
- Treat The and the as the same word when counting
- EXTRA: remove punctuation, so Hamlet, == Hamlet
- Test on Shakespeare's Hamlet (<http://bit.ly/BillShak>)

Exercise Two

- Let's write a function which takes a word as an argument and outputs the plural of that word
- The program should follow these rules:
 - If the word ends in 's', 'x', or 'z', the plural adds 'es', e.g., ax => axes, loss => losses
 - If the word ends in an 'h', which is not preceded by a vowel or 'd', 'g', 'k', 'p', 'r', or 't', the plural adds 'es', e.g., moth => moths, but match => matches
 - If the word ends in a 'y' which is not preceded by a vowel, then the plural strips the 'y' and adds 'ies', e.g., baby => babies, but boy => boys
 - Otherwise just add 's'
- Prompt the user for a string and pass through function
- Output results to verify