

Scripts revisited

- the `#!/bin/bash` found at the beginning of a script, for example

```
#!/bin/bash
echo "hello world"
```

- indicate to GNU/Linux & UNIX that this is a bash script and that the script must be run using this interpreter

Effect of #! on GNU/Linux and Cygwin

- under GNU/Linux and Windows (Cygwin) it is treated as a comment by the `bash` interpreter but, critically,
 - the GNU/Linux operating system `exec` routine sees `#!` as a 16 bit magic number indicating that this file must be run with the program found in the string up to the next newline
 - thus should a script be given executable permission it will behave to users like an executable binary
- `#!` means nothing to Windows and so the `bash` interpreter on Windows or (cygwin) simply treats this line as a comment.

Autoftp in python

- consider an `autoftp` file which consists of a number of ftp url's

```
# GNU Troff
j228-gm.comp.glam.ac.uk/pub/groff/groff.tar.gz # this is .
j228-gm.comp.glam.ac.uk/pub/groff/README # another .
```

- a `#` indicates a comment
- step one in creating the `autoftp` script in python is to read in the `autoftp` file a line at a time
- step two, remove the comments (hint examine `split`)

Autoftp in python

- step three, split the ftp url into 3 components
 - site
 - directory
 - filename
- step four, retrieve the file (hint examine the `ftp` module)

Reading a file, a line at a time

```
#!/usr/bin/python
for line in open('autoftprc').readlines():
    print line
```

- removing a comment can be achieved by:

```
uncomment = string.split(line, '#')
print uncomment[0]
```

Extracting the site, directory and filename

- think `split` again

```
words = string.split(uncomment[0], '/')
■ words[0] is the site
■ words[1..n-2] is the directory
■ words[-1] is the filename
■ n is len(words)
```

Putting the components together

- finally here is the `autoftp` program in python so far:

```
#!/usr/bin/python
import sys, string
from ftplib import FTP

for line in open('autoftprc').readlines():
    uncomment = string.split(line, '#')
    words = string.split(uncomment[0], '/')
    i = len(words)
    if i>1:
        print ' is the site', words[0]
        print ' is the file', words[i-1]
        directory=""
        for j in range(1,i-1):
            directory = directory + '/' + words[j]
        site = words[0]
        file = words[-1]

        print 'directory is', directory
```

Tutorial work

- copy the `autoftprc` file as given in these slides (make sure you call your file `autoftprc`)
- now create a Python script called `autoftp.py` and cut and paste the code from stage 1 into this file.
- run this code and satisfy yourself that you understand what is going on
- now move to stage 2 and print all the uncommented lines in `autoftprc`

Tutorial work

- now split the url into the three components: site, directory and filename
 - hint solve the `site` and `filename` first and then solve the directory.
 - you should just print these items to the screen

- now read the ftp module in the global module index (found in the [Python documentation](#) `<../../../../python/html/index.html>`)

- import the ftp module and call the appropriate methods necessary to download the file from the remote site