Best Friends: Strings and Lists

abc = ‘With three words’

- stuff = abc.split()
  print(stuff)
  [‘With’, ‘three’, ‘words’]
  - print(len(stuff))
    3
    - print(stuff[0])
    - With
      - print(stuff)
      - [With’, ‘three’, ‘words’]
        - for w in stuff:
             print(w)
          With
          Three
          Words

split() breaks a string into parts and produces a List. We think of these as words. We can access a particular word or loop through all the words.

- The split() function takes away the complication that loops bring

When you do not specify a delimiter, multiple spaces are treated like one delimiter.

- line = ‘A lot of   spaces’
  etc = line.split()
  print(etc)
  [‘A’, ‘lot’, ‘of’, ‘spaces’]

You can specify what delimiter character to use in the splitting.

- line = ‘first;second;third’
  thing = line.split(‘;’)
  print(thing)
  [‘first’, ‘second’, ‘third’]

Best Friends: Strings and Lists (Part 2)

```
From stephen.marquard@utc.ac.za Sat Jan 5 09:14:16 2008
```

```
fhand = open(‘mbox-short.txt’)
for line in fhand:
    line = line.rstrip()
    if not line.startswith(‘From ’): continue
    words = line.split()
    print(words[2])
```

```
line = ‘From stephen.marquard@utc.ac.za Sat Jan 5 09:14:16 2008’
words = line.split()
print(words)
[‘From’, ‘stephen.marquard@utc.ac.za’, ‘Sat’, ‘Jan’, ‘5’, ‘09:14:16’, ‘2008’]
```

The Double Split Pattern

Sometimes we split a line one way, and then grab one of the pieces of the line and split that piece up again.

```
line = ‘From stephen.marquard@utc.ac.za Sat Jan 5 09:14:16 2008’

words = line.split()
email = words[1]
pieces = email.split(‘@’)
print(pieces[1])
‘uct.ac.za’
```