Introduction to Python
More functions, ...

Christopher Barker

UW Continuing Education / Isilon

June 27, 2012

Table of Contents

title

Lab from end of last class?

LAB

`def count_them(letter):`

- prompts the user to input a letter
- counts the number of times the given letter is input
- prompts the user for another letter
- continues until the user inputs "x"
- returns the count of the letter input

`def count_letter_in_string(string, letter):`

- counts the number of instances of the letter in the string
- ends when a period is encountered
- if no period is encountered – prints "hey, there was no period!"

Questions?

Any Questions about:

- Last class ?
- Reading ?
- Homework ?

Homework review

Homework notes

subprocesses

Subprocesses

```
#easy:
os.popen('ls').read()

#even easier:
os.system('ls')

# but for anything more oomplicated:
pipe = \
  subprocess.Popen("ls", stdout=subprocess.PIPE).stdout
```

reload

module importing and reloading

```
In [190]: import module_reload

In [191]: module_reload.print_something()
I'm printing something

# change it...
In [196]: reload(module_reload)
Out[196]: <module 'module_reload' from 'module_reload.py'>

In [193]: module_reload.print_something()
I'm printing something else
```

Module Reloading

```
In [194]: from module_reload import this

# change it...

In [196]: reload(module_reload)
Out[196]: <module 'module_reload' from 'module_reload.py'>

In [197]: module_reload.this
Out[197]: 'this2'

In [198]: this
Out[198]: 'this'
```

repr vs. str

`repr()` vs `str()`

```
In [200]: s = "a string\nwith a newline"

In [203]: print str(s)
a string
with a newline

In [204]: print repr(s)
'a string\nwith a newline'
```

repr vs. str

```
eval(repr(something)) == something
```

```
In [205]: s2 = eval(repr(s))

In [206]: s2
Out[206]: 'a string\nwith a newline'
```

Default Parameters

Sometimes you don't need the user to specify everything every time

```
def fun(x,y,z=5):
    print x,y,z
```

Building Strings

The string format operator: %

```
In [178]: "a string"
Out[178]: 'a string'

In [179]: str(34.5)
Out[179]: '34.5'

In [180]: `34.56`
Out[180]: '34.56'

In [181]: "the number %s is %i"%('five', 5)
Out[181]: 'the number five is 5'
```

String formatting

Gotcha

```
In [127]: "this is a string with %i formatting item"%1
Out[127]: 'this is a string with 1 formatting item'

In [128]: "string with %i formatting %s: "%2, "items"
TypeError: not enough arguments for format string

# Done right:
In [131]: "string with %i formatting %s"%(2, "items")
Out[131]: 'string with 2 formatting items'

In [132]: "string with %i formatting item"%(1,)
Out[132]: 'string with 1 formatting item'
```

LAB

Format operators:

-

String methods

bunch of...

Sequence API

full API

http://docs.python.org/library/stdtypes.html#
sequence-types-str-unicode-list-tuple-bytearray-buffer-xra

Text File Notes

Text is default

- newlines are translated: `\r\n -> \n`
- reading and writing!
- Always use *nux-style in your code: `\n`
- Open text files with `'U'` "Universal" flag

Gotcha:

- no difference between text and binary on *nix
    - breaks on Windows