The ConfigParser module

This module reads configuration files.

The files should be written in a format similar to Windows INI files. The file contains one or more sections, separated by section names written in brackets. Each section can contain one or more configuration items.

Here's an example:

```
[book]
title: The Python Standard Library
author: Fredrik Lundh
email: fredrik@pythonware.com
version: 2.0-001115

[ematter]
pages: 250

[hardcopy]
pages: 350
```

Example: Using the ConfigParser module

```
# File: configparser-example-1.py

import ConfigParser
import string

config = ConfigParser.ConfigParser()

config.read("samples/sample.ini")

# print summary
print
print string.upper(config.get("book", "title"))
print "by", config.get("book", "author"),
print  "(" + config.get("book", "email") + ")"
print
print config.get("ematter", "pages"), "pages"
print

# dump entire config file
for section in config.sections():
    print section
    for option in config.options(section):
        print " ", option, "=", config.get(section, option)


THE PYTHON STANDARD LIBRARY
by Fredrik Lundh (fredrik@pythonware.com)

250 pages

book
  title = Python Standard Library
  email = fredrik@pythonware.com
  author = Fredrik Lundh
  version = 2.0-010504
  __name__ = book
ematter
  __name__ = ematter
  pages = 250
hardcopy
  __name__ = hardcopy
  pages = 300
```

In Python 2.0, this module also allows you to write configuration data to a file.

Example: Using the ConfigParser module to write configuration data

```
# File: configparser-example-2.py

import ConfigParser
import sys

config = ConfigParser.ConfigParser()

# set a number of parameters
config.add_section("book")
config.set("book", "title", "the python standard library")
config.set("book", "author", "fredrik lundh")

config.add_section("ematter")
config.set("ematter", "pages", 250)

# write to screen
config.write(sys.stdout)


[book]
title = the python standard library
author = fredrik lundh

[ematter]
pages = 250
```