

Python Walk-Through

```
#!/usr/bin/python
```

Tells the shell which interpreter to use

```
import random, sys
from optparse import OptionParser
```

Import statements, similar to include statements
Import OptionParser class from optparse module

```
class randline:
    def __init__(self, filename):
        f = open (filename, 'r')
        self.lines = f.readlines()
        f.close ()

    def chooseline(self):
        return random.choice(self.lines)
```

The beginning of the class statement: randline
The constructor
Creates a file handle
Reads the file into a list of strings called lines
Close the file

```
def main():
    version_msg = "%prog 2.0"
    usage_msg = "%prog [OPTION]..."
    FILE Output randomly selected lines from
    FILE."
```

The beginning of a function belonging to randline
Randomly select a number between 0 and the size of lines
and returns the line corresponding to the randomly
selected number
The beginning of main function
version message
usage message

Python Walk-Through

```
parser = OptionParser(version=version_msg,
                      usage=usage_msg)
parser.add_option("-n", "--numlines",
                  action="store", dest="numlines",      default=1,
                  help="output NUMLINES lines (default 1)")

options, args = parser.parse_args(sys.argv[1:])

try:
    numlines = int(options.numlines)
except:
    parser.error("invalid NUMLINES: {0}".
                format(options.numlines))

if numlines < 0:
    parser.error("negative count: {0}".
                format(numlines))

if len(args) != 1:
    parser.error("wrong number of operands")
input_file = args[0]

try:
    generator = randline(input_file)
    for index in range(numlines):
        sys.stdout.write(generator.chooseline())
except IOError as (errno, strerror):
    parser.error("I/O error({0}): {1}".format(errno,
        strerror))

if __name__ == "__main__":
    main()
```

Creates OptionParser instance

Start defining options, action "store" tells optparse to take next
argument and store to the right destination which is "numlines".
Set the default value of "numlines" to 1 and help message.

options: an object containing all option args
args: list of positional args leftover after parsing options

Try block
get numline from options and convert to integer

Exception handling
error message if numlines is not integer type, replace {0} w/
input

If numlines is negative
error message

If length of args is not 1 (no file name or more than one file name)
error message

Assign the first and only argument to variable input_file
Try block

instantiate randline object with parameter input_file
for loop, iterate from 0 to numlines - 1
print the randomly chosen line

Exception handling
error message in the format of "I/O error (errno):strerror"

In order to make the Python file a standalone program