New York University
School of Continuing and Professional Studies
Division of Programs in Information Technology

Introduction to Python

Exercise Solutions, Session 8

Ex. 8.1 Allow your Program to Accept Arguments. Accept two arguments at the command line, using the default list object sys.argv. Sum the values and print out a formatted string with an "addition" formula.

```
import sys

arg1 = sys.argv[1]
arg2 = sys.argv[2]

arg1 = int(arg1)
arg2 = int(arg2)

print("{} + {} = {}".format(arg1, arg2, (arg1 + arg2)))
```

Ex. 8.2 Validate Arguments. Extend the above program to validate user input.

```
import sys

usage = 'Usage:  validate_addargs.py [int1] [int2]'

try:
    arg1 = sys.argv[1]
    arg2 = sys.argv[2]
except IndexError:
    exit(usage)

try:
    this_sum = int(arg1) + int(arg2)
except ValueError:
    exit(usage)

print("{} + {} = {}".format(arg1, arg2, this_sum))
```

Ex. 8.3 os.path.getsize(). Pick a file in the same directory as your Python script (it can even be this very same Python script!). Write a script that prints the size of the file.

```
import os

filename = '../python_data/student_db.txt'      # this is the name of an existing file

print("{}:  {} bytes".format(filename, os.path.getsize(filename)))
```

Ex. 8.4 Validate filename. Continuing the previous program, take a filename from the user through the command line. Use os.path.isfile() to see if the submitted file is a file in the current directory. If it is a file, then print the filename and size. If it is not a file, then print an error message.

```
import sys
import os

filename = sys.argv[1]

if os.path.isfile(filename):
    print("{}:  {} bytes".format(filename, os.path.getsize(filename)))

else:
    exit('error:  {} is not a file in this directory'.format(filename))
```

Ex. 8.5 List a Directory. Accept a string argument that is the pathname of a directory. Print out all items in the directory listing using os.listdir(). Identify whether the listing is a file or directory.

```
import sys
import os

try:
    dirname = sys.argv[1]
except IndexError:
    exit('error:  please provide an argument')

try:
    files = os.listdir(dirname)
except IOError:                   # in Windows, a WindowsError
    exit('error:  directory does not exist or is not readable')

for filename in files:
    full_path = os.path.join(dirname, filename)
    if os.path.isfile(full_path):
        this_type = 'file'
    else:
        this_type = 'dir'

    print("{} ({})".format(filename, this_type))
```

Ex. 8.6 List files and sizes in another directory. Continuing the previous exercise, output the file name and byte size (using os.path.getsize()) of each file.

```
import sys
import os

try:
    dirname = sys.argv[1]
except IndexError:
    exit('error:  please provide an argument')

try:
    files = os.listdir(dirname)
except IOError:                   # in Windows, a WindowsError
    exit('error:  directory does not exist or is not readable')

for filename in files:
    full_path = os.path.join(dirname, filename)
    if os.path.isfile(full_path):
        this_type = 'file'
    else:
        this_type = 'dir'

    bytesize = os.path.getsize(full_path)
    print("{} ({}):  {}".format(filename, this_type, bytesize))
```