**Introduction to Python**

**Day Four Cheatsheet: File I/O and Modules**

**File input/output**

# Read a file using the .read() method

file\_handle = open("file.txt", "r")

file\_string = file\_handle.read() # Read file as a single string

file\_handle.close()

# Read a file using the .readlines() method

file\_handle = open("file.txt", "r")

file\_line\_list = file\_handle.readlines() # Read file lines into a list

file\_handle.close()

# Read a file using the .readlines() method and with control-flow

with open("file.txt", "r") as file\_handle:

file\_line\_list = file\_handle.readlines()

# Write to a file

file\_handle = open("file.txt", "w")

file\_handle.write("This sentence is now in the file.\n")

file\_handle.write("And so is this sentence, on the next line!")

file\_handle.close()

# Write to a file using with control-flow

with open("file.txt", "w") as file\_handle:

file\_handle.write("This sentence is now in the file.\n")

file\_handle.write("And so is this sentence, on the next line!")

**Modules**

To import a module, use either of the following commands (example uses module os)

import os # use as os.<function>, e.g. os.listdir()

from os import \* # use as <function>, e.g. listdir()

**The** sys **module**

|  |  |  |  |
| --- | --- | --- | --- |
| Module | Command | Description | Example |
| sys | sys.path | Contains a list of directories in the python path | sys.path.append("another/path/to/access/") |
| sys | sys.argv | Contains a list of command-line arguments (all strings!) | save\_me = sys.argv[1] |
| sys | sys.exit() | Immediately exit out of a script | sys.exit() |

**The** os **and** shutil **modules**

|  |  |  |  |
| --- | --- | --- | --- |
| Module | Command | UNIX equivalent | Example |
| os | os.listdir | ls | files = os.listdir("directory/with/files") |
| os | os.remove | rm | os.remove("file.txt") |
| os | os.mkdir | mkdir | os.mkdir("new\_directory") |
| os | os.chdir | cd | os.chdir("go/to/this/directory/") |
| shutil | shutil.copy | cp | shutil.copy("old\_file", "new\_file") |
| shutil | shutil.move | mv | shutil.move("old\_file", "new\_file") |