Title: Notes, Week 11
Page 1 of 2

Name: Zach Caldwell
Class: CS 112-01
Date: March 21, 2017

Cues:

shutil.copy()

shutil.copytree()

shutil.move()

shutil.rmtree()

os.unlink()

os.rmdir()

os.walk()

send2trash.send2trash()

zipfile.ZipFile()

ZipFile.close()

Notes:

1. Organizing files:
    a. Using the **shutil** (shell utilities) module:
        i. `shutil.copy(sourcePath, destPath)` will copy the given source file to the destination folder or file path. It returns the final path of the file copy.

        ii. `shutil.copytree()` functions much like `shutil.copy()`, except that it is called on a folder path and will copy it recursively.

        iii. `shutil.move()` will move the given source file or folder path to the given destination (possibly overwriting it), and will return the final path of the new location.

        iv. `shutil.rmtree()` will recursively delete the given folder and its contents.

    b. Using the **os** module:
        i. `os.unlink()` will delete (or technically *unlink* the filename from) the given file path.

        ii. `os.rmdir()` will delete an *empty* folder at the given path.

        iii. `os.walk()` will traverse (‘walk’) the given directory tree, returning a current folder name, a list of subfolders, and a list of files on each call.

    c. Using the third-party **send2trash** module:
        i. `send2trash.send2trash()` will send the given file or folder path to the computer’s trash or recycle bin.

    d. Using the **zipfile** module:
        i. The `zipfile.ZipFile()` function will create a ZipFile object using the given ZIP file path.

        ii. The `ZipFile.close()` method will close the ZipFile object it is called on, if the object is open.

Title: Notes, Week 11

Name: Zach Caldwell
Class: CS 112-01
Date: March 21, 2017

Cues:	**Notes:**
ZipFile.namelist()	iii. The `ZipFile.namelist()` method will return a list of relative path strings for everything in the ZIP file.
ZipFIle.getinfo()	iv. The `ZipFile.getinfo()` method can take a file string (contained in `.namelist()`) and return a ZipInfo object with information about the file.
ZipFile.extract()	v. The `ZipFile.extract()` method will attempt to extract a file from the archive into either the current working directory or a given (possibly non-existent) folder path.
ZipFile.extractall()	vi. The `ZipFile.extractall()` method will attempt to extract *everything* from the ZIP archive into either the current working directory or a given (possibly non-existent) folder path.
ZipFile.write()	vii. If the ZIP archive was opened in write or append mode, the `ZipFile.write()` method will add the given file to the archive, using the compression algorithm optionally given using the `compress_type` named parameter.

Summary/Reflection: