Cleaning up node\_modules files

# Purpose

These notes detail the creation of the cleanup\_node\_modules python program. I find it quite time consuming going through my various projects trying to clean up, that is, remove all the files under all node\_modules folders.

These notes capture all the key python libraries I will use (many for the first time) to implement the program

# Processing Flags

I want the program to be able to accept various flags when the script is invoked via the command line. I want the code to work as:

python cleanup\_node\_modules [-f] -d <top-level-folder> [-o <output-file>]

# Options:  
# -f just check but do not remove any files  
# -d the directory of folder to start the search in  
# -o the output file to write the <path-directory> of any node\_modules found

* f – an optional flag to provide the capability to run the program where it just identifies the non-empty node\_modules folders it found but takes no action
* d – provide the path name to the top-level folder to start the search for the node\_modules folder to empty out
* o – an optional flag that provides the path name of the file to create to list all the location of all the non-empty node\_modules folders found

The module I need to utilize is the argparse.

A blue and white rectangle

AI-generated content may be incorrect.

I need to create an Argument Parser object:

A screen shot of a computer code

AI-generated content may be incorrect.

Add arguments:

1. parser.add\_argument('filename', help='Name of the file to process')  
parser.add\_argument('-v', '--verbose', action='store\_true', help='Enable verbose output')  
parser.add\_argument('-n', '--num-iterations', type=int, default=10, help='Number of iterations to run')