report_p_03.md 5/3/2020

P-03

CSCI - P436 | Brandon Young

Source Code

```
# Imports
import sys
import fileinput
import argparse
import fileinput
# Global variables
version = "1.0.0"
# Methods
def parseInput():
    parser = argparse.ArgumentParser(add_help=False)
    parser.add_argument("-v", "--version", action="store_true",help="Display
current version info")
    parser.add_argument('-h','-H', '--help',"-?", action='help',
default=argparse.SUPPRESS, help='Show this help message and exit.')
    parser.add_argument("-f","--file", type=str, help="The file to output")
    args = parser.parse_args()
    return args
def evalInfo(argsData):
    if not sys.stdin.isatty():
        data = fileinput.input()
        printStdIn(data)
        data.close()
    if argsData.file != None:
        datafile = open(argsData.file)
        printFile(datafile)
    if argsData.version:
        print("shell.py version --> " + version)
def printFile(file):
    count = 0
    for line in file:
            if count > 1:
                sys.stdout.write(line[3:])
            count += 1
def printStdIn(stdIn):
    for line in stdIn:
            if stdIn.lineno() > 2:
                sys.stdout.write(line[3:])
```

report p 03.md 5/3/2020

```
# Main

parsed = parseInput()
evalInfo(parsed)
```

Execution

```
# usage: shell.py [-v] [-h] [-f FILE]

# optional arguments:
# -v, --version Display current version info
# -h, -H, --help, -? Show this help message and exit.
# -f FILE, --file FILE The file to output

# or alternatively
cat <filename> | python .\shell.py
```

Outputs

```
python .\shell.py -h
# usage: shell.py [-v] [-h] [-f FILE]
# optional arguments:
# -v, --version
     Display current version info
# -h, -H, --help, -? Show this help message and exit.
# -f FILE, --file FILE The file to output
python ./shell.py -v
# shell.py version --> 1.0.0
python ./shell.py -f ./docs/formatted_disk.txt
```

5/3/2020 report p 03.md

cat ./docs/formatted_disk | python ./shell.py