**16/03/2019 Lecture19**

**#WAP to accept file name from user and print its statistical information like type of file,size of file, permissions. (use stat/fstat system call)**

>>> statobj=os.stat("NEWS.txt")

>>> statobj

nt.stat\_result(st\_mode=33206, st\_ino=0L, st\_dev=0L, st\_nlink=0, st\_uid=0, st\_gid=0, st\_size=493404L, st\_atime=1546743397L, st\_mtime=1525086452L, st\_ctime=1525086452L)

>>>

**#WAP to accept a folder name from user and create zip file out of it.**

**(hint: import shutil)**

**#WAP to accept file name from user and number of lines to copy to another file ( Use command line argumets )**

**#WAP to accept file name from user and number of lines to copy to another file ( Use command line argumets )**

import sys

import argparse

def copyFilewithLine(sourceFile,destinationFile,noofLines):

fd=open(sourceFile)

fd1=open(destinationFile,"w")

count=0

while(count<noofLines):

line=fd.readline()

fd1.write(line)

count+=1

else:

print("line copy completed")

fd.close()

fd1.close()

def main():

parser=argparse.ArgumentParser()

parser.add\_argument("-i",type=str,help="Input/Source File")

parser.add\_argument("-d",type=str,help="Output/Destination File")

parser.add\_argument("-n",type=int,help="Number of lines to copy",default=0)

args=parser.parse\_args()

copyFilewithLine(args.i,args.d,args.n)

if \_\_name\_\_=='\_\_main\_\_':

main()

#WAP to accept file name from user and number of lines to copy to another file ( Use command line argumets and shutil)

import sys

import shutil

import argparse

def copyfile(src,dest,n):

srcfd=open(src)

destfd=open(dest,"w")

count=0

if(n==0):

shutil.copyfileobj(srcfd,destfd)

else:

while(count<n):

line=srcfd.readline()

destfd.write(line)

count+=1

srcfd.close()

destfd.close()

def main():

parser=argparse.ArgumentParser()

parser.add\_argument("-i",type=str,help="Input/Source File")

parser.add\_argument("-d",type=str,help="Output/Destination File")

parser.add\_argument("-n",type=int,help="Number of lines to copy",default=0)

args=parser.parse\_args()

copyfile(args.i,args.d,args.n)

if \_\_name\_\_=='\_\_main\_\_':

main()

**# explore OptionParser library**

**# Pickle - Python pickle** module is used for serializing and de-serializing a **Python** object structure

serialization – process of converting data in the form of stream of bytes

deserialization – convert the stream of bytes back to the original object