

## 9)Write Python program

a. To elaborate file operations such as, opening a file, reading from it, writing into it, closing it, and various file methods.

open,read,readline,readlines,write,writelines

```
In [82]: #write operation in a file using mode 'w'
fp = open("test.txt",'w')
fp.write('Your Friendly Neighbourhood SpiderMan\n') # single line statement
fp.writelines(['Peter parker\n','NO Way Home'])# writelines function write multiple lin
fp.close()
print('written successfully')
```

written successfully

```
In [83]: fp = open("test.txt",'r') # make sure the file exists in the directory or folder else y
res=fp.read() # reading a file using mode 'r'
print(res)
fp.seek(0) # modifying file pointer position to origin
res1=fp.read(10) # reading a fixed no. of bytes from the file
print(res1)
fp.seek(0)
print(fp.readline()) # readline, reads one line of file
fp.seek(0)
print(fp.readlines())
fp.seek(0) # readlines, reads multiple lines of file
print(fp.readlines()[1]) # readlines, reads multiple lines of file using indexing
fp.close()
```

Your Friendly Neighbourhood SpiderMan  
 Peter parker  
 NO Way Home  
 Your Frien  
 Your Friendly Neighbourhood SpiderMan

['Your Friendly Neighbourhood SpiderMan\n', 'Peter parker\n', 'NO Way Home']  
 Peter parker

```
In [84]: # with statement : it automatically closes the file no need to explicitly close the fil
with open('test.txt','w') as fp:
    fp.write('SpiderMan is a my Friend')
print('file has been written and also been closed')
```

file has been written and also been closed

```
In [86]: # appending a file using mode 'a'
# open the file.txt in append mode. Create a new file if no such file exists.
fp = open("tommy.txt", "w")
# writing the content to the file
fp.write('Python is the modern day language. It makes things so simple.
It is the fastest-growing programing language')
print("written successfully")
# closing the opened the file
fp.close()
```

written successfully

```
In [87]: # now appending text to above mentioned file
fp = open("tommy.txt", "a")
# appending the content to the file
fp.write(' \nPython is object oriented language')
print("written successfully")
# closing the opened the file
fp.close()
print("appended successfully")

with open('tommy.txt','r') as f:
    print(f.read())
```

written successfully  
appended successfully  
Python is the modern day language. It makes things so simple.  
It is the fastest-growing programming language  
Python is object oriented language

```
In [88]: # read and write the file using mode 'r+' it creates file if not exists
with open('test.txt','r+') as f:
    print(f.tell()) # It returns the current position of the file pointer within the fi
    print(f.read())
    print('now pointer position is at ',f.tell())
    f.write(' hi')
    f.seek(0)
    print(f.read())
```

0  
SpiderMan is a my Friend  
now pointer position is at 24  
SpiderMan is a my Friend hi

```
In [89]: with open('test.txt','w+') as f: # w+ overwrites the existing file
    print(f.tell()) # It returns the current position of the file pointer within the fi
    print(f.read())
    f.write('hi')
    f.seek(0)
    print(f.read())
```

0  
hi

```
In [90]: with open('test.txt','a+') as f: # a+ appends the text the existing file
    print(f.read())
    f.write(' hello')
    f.seek(0)
    print(f.read())
```

hi hello

```
In [95]: # seek function: It modifies the position of the file pointer to a specified offset wit
with open('test.txt','r+') as f: # reading a file in binary format
    print('the pointer is at ',f.tell())
    f.write('hahahah')
```

```
with open('test.txt','rb+') as fp:
    print(fp.read())
    print('the pointer is at ',fp.tell())
    fp.seek(-3,1) # from current position to 4 position before
    print('the pointer is at ',fp.tell())
    fp.seek(5,0) # from beginning to 5th position
    print('the pointer is at ',fp.tell())
    fp.seek(5,2) # from end to last 5th position
    print('the pointer is at ',fp.tell())
```

```
the pointer is at 0
b'hahahaho'
the pointer is at 8
the pointer is at 5
the pointer is at 5
the pointer is at 13
```

```
In [117... # creating a file using mode x
with open('tmtmt.txt','x') as f:
    print(f)
if f:
    print('file created successfully')
```

```
<_io.TextIOWrapper name='tmtmt.txt' mode='x' encoding='cp1252'>
file created successfully
```

b. To elaborate file and directory management such as creating a directory, renaming it, listing all directories and working with them.

```
In [96]: import os
os.getcwd()
```

```
Out[96]: 'c:\\Users\\DELL\\Documents\\5th sem\\Python'
```

```
In [97]: #We can also use the getcwd() method to get it as bytes object.
os.getcwdb()
```

```
Out[97]: b'c:\\Users\\DELL\\Documents\\5th sem\\Python'
```

```
In [98]: os.mkdir('Data') # making a directory
```

```
In [100... os.chdir('Data') # changing the current working directory to data
print(os.getcwd())
```

```
c:\\Users\\DELL\\Documents\\5th sem\\Python\\Data
```

```
In [113... os.chdir('C:\\Users\\DELL\\Documents\\5th sem\\Python')
#listing all directories
print(os.listdir())
print(os.getcwd())
```

```
['.vscode', '19BTRCR018_Pythonhistory.pptx', '19BTRCR018_python_lab-1.ipynb', '19BTRCR018_python_lab-2.html', '19BTRCR018_python_lab-2.ipynb', '19BTRCR018_python_lab-3.html', '19BTRCR018_python_lab-3.ipynb', '19BTRCR018_python_lab-4.ipynb', '19BTRCR018_python_lab-5.html', '19BTRCR018_python_lab-5.ipynb']
```

```
-5.html', '19BTRCR018_python_lab-5.ipynb', '19BTRCR018_python_lab-6.html', '19BTRCR018_p
ython_lab-6.ipynb', '19BTRCR018_python_lab-9.ipynb', 'Data', 'hello.py', 'pywhatkit_dbs.
txt', 'rishab.txt', 'test3.txt', 'whatsapp.py']
C:\Users\DELL\Documents\5th sem\Python
```

```
In [110...
os.rename('test2.txt', 'test3.txt') # renaming a file
print(os.listdir())
```

```
['.vscode', '19BTRCR018_Pythonhistory.pptx', '19BTRCR018_python_lab-1.ipynb', '19BTRCR01
8_python_lab-2.html', '19BTRCR018_python_lab-2.ipynb', '19BTRCR018_python_lab-3.html',
'19BTRCR018_python_lab-3.ipynb', '19BTRCR018_python_lab-4.ipynb', '19BTRCR018_python_lab
-5.html', '19BTRCR018_python_lab-5.ipynb', '19BTRCR018_python_lab-6.html', '19BTRCR018_p
ython_lab-6.ipynb', '19BTRCR018_python_lab-9.ipynb', 'Data', 'hello.py', 'pywhatkit_dbs.
txt', 'rishab.txt', 'test3.txt', 'whatsapp.py']
```

```
In [118...
#removing a file
os.remove('tmtmt.txt')
print('file is removed')
```

file is removed

c. To elaborate exception handing with python built in commands such as try, except, and finally.

```
In [119...
#Syntax error
print('abs')
```

```
File "<ipython-input-119-86584e74b01d>", line 2
    print('abs'
          ^
```

**SyntaxError:** unexpected EOF while parsing

```
In [120...
#logical errors
a = 1
b = 2
print('sum of two numbers is ',a-b)
```

sum of two numbers is -1

```
In [121...
# run time errors
a=int(input('enter a number'))
print(a)
```

```
-----
ValueError                                Traceback (most recent call last)
<ipython-input-121-e74fdadc6cc4> in <module>
      1 # run time errors
----> 2 a=int(input('enter a number'))
      3 print(a)
```

**ValueError:** invalid literal for int() with base 10: 'f'

```
In [124...
#handling exceptions using try and except
a = [1, 2, 3]
try:
    print ("2nd element = {}".format(a[1]))

    print ("4th element = {}".format(a[3]))
```

```
except:
    print ("An error occurred\n")
```

2nd element = 2  
An error occurred

In [127...

```
try:
    a = int(input("Enter a:"))
    b = int(input("Enter b:"))
    c = a/b
    print("a/b = {}".format(c))
# Using Exception with except statement. If we print(Exception) it will return exceptio
except Exception:
    print("can't divide by zero")
    print(Exception)
else:
    print("Hi I am else block")
try:
    a = int(input("Enter a:"))
    b = int(input("Enter b:"))
    c = a/b
    print("a/b = {}".format(c))
# Using Exception with except statement. If we print(Exception) it will return exceptio
except Exception:
    print("can't divide by zero")
    print(Exception)
else:
    print("Hi I am else block")
```

can't divide by zero  
<class 'Exception'>  
a/b = 1.0  
Hi I am else block

In [128...

```
# using a keyboard
try:
    a=int(input('enter a number'))
    print(a)
except Exception as e:
    print("!!error!!\n",e)
```

!!error!!  
invalid literal for int() with base 10: 'r'

In [130...

```
# multiple exceptions
try:
    a=10/0;
except(ArithmeticError, IOError):
    print("Arithmetic Exception")
else:
    print("Successfully Done")
```

Arithmetic Exception

In [131...

```
# multiple exceptions
try:
    a=10/0;
```

```
except(ArithmeticError):  
    print("Arithmetic Exception")  
except(Exception):  
    print("Arithmetic Exception")  
else:  
    print("Successfully Done")
```

Arithmetic Exception

In [134...

```
# try finally block  
try:  
    fileptr = open("rishab.txt","r")  
    try:  
        fileptr.write("Hi I am good")  
    finally:  
        fileptr.close()  
        print("file closed")  
except Exception as e:  
    print("Error: ",e)
```

file closed  
Error: not writable

In [135...

```
#raising error  
try:  
    age = int(input("Enter the age:"))  
    if(age<18):  
        raise ValueError  
    else:  
        print("the age is valid")  
except ValueError:  
    print("The age is not valid")
```

The age is not valid