Marwadi University Marwadi Chandarana Group NAAC A+	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Practical based on File Handling using Python	
Experiment No: 13	Date:	Enrollment No:

Aim: Practical based on File Handling using Python

IDE:

File handling in Python is a powerful and versatile tool that can be used to perform a wide range of operations. However, it is important to carefully consider the advantages and disadvantages of file handling when writing Python programs, to ensure that the code is secure, reliable, and performs well.

Python provides various functions to perform different file operations, a process known as File Handling.

- ***open()*** *: Opens a file and returns a file object.*
- ***read()*** *: Reads data from a file.*
- ***write()*** *: Writes data to a file.*
- ***close()*** *: Closes the file, releasing its resources.*

Opening Files in Python

In Python, we need to open a file first to perform any operations on it—we use the open() function

Suppose we have a file named ict.txt

To open this file, we can use the open() function.

file1 = open("C:\\Users\\Mitesh\\OneDrive\\Desktop \\ict.txt")

or

file1 = open(r"C:\Users\Mitesh\OneDrive\Desktop \ict.txt")

Output

Working in Read mode

The open command will open the Python file in the read mode and the for loop will print each line present in the file.

```
f1 = open(r"C:\Users\Mitesh\OneDrive\Desktop \ict.txt")
# This will print every line one by one in the file
for each in f1:
```

```
    print (each)
```

Output:

```
In [1]: runfile('C:/Users/student/Desktop/PWP
Pratical/untitled0.py', wdir='C:/Users/
student/Desktop/PWP Pratical')
ICT ICT ICT

ICT ICT ICT ICT ICT
```

In this example, we will extract a string that contains all characters in the Python file then we can use f1.read().

```
# Python code to illustrate read() mode
f1 = open(r"C:\Users\Mitesh\OneDrive\Desktop \ict.txt")
print (f1.read())
```

Output :

```
In [2]: runfile('C:/Users/student/Desktop/PWP
Pratical/untitled1.py', wdir='C:/Users/
student/Desktop/PWP Pratical')
ICT ICT ICT
ICT ICT ICT ICT ICT
```

Example

In this example, Read a file using the with statement in Python.

```
with open(r"C:\Users\Mitesh\OneDrive\Desktop\ict.txt",'r') as f1:
    data = f1.read()
print(data)
```

Output :

```
In [3]: runfile('C:/Users/student/Desktop/PWP
Pratical/untitled2.py', wdir='C:/Users/
student/Desktop/PWP Pratical')
ICT ICT ICT
ICT ICT ICT ICT ICT
```

Example 4:

Another way to read a file is to call a certain number of characters like in the following code the interpreter will read the first five characters of stored data and return it as a string:

f1 = open(r"C:\Users\Mitesh\OneDrive\Desktop \ict.txt")
print (f1.read(5))
Output

```
In [4]: runfile('C:/Users/student/Desktop/PWP
Pratical/untitled3.py', wdir='C:/Users/
student/Desktop/PWP Pratical')
ICT I
```

Example

The split() function splits the variable when space is encountered. You can also split using any characters as you wish.

```
with open(r"C:\Users\Mitesh\OneDrive\Desktop\ict.txt",'r') as file:
    data = file.readlines()
    for line in data:
        word = line.split()
        print (word)
```

Output

```
In [16]: runfile('C:/Users/student/Desktop/
PWP Pratical/untitled8.py', wdir='C:/Users/
student/Desktop/PWP Pratical')
['ICT', 'ICT', 'ICT']
['ICT', 'ICT', 'ICT', 'ICT', 'ICT']
```

Working in Write Mode

The write() function is used to write in a file. The close() command terminates all the resources in use and frees the system of this particular program.

```
file = open("ict1.txt",'w')

file.write("ICT ICT ICT \n")
file.write("ICT ICT ICT ICT ICT")
file.close()
```

Output :

Marwadi University Marwadi Chandarana Group NAAC A+	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Practical based on File Handling using Python	
Experiment No: 13	Date:	Enrollment No:

Using with() function

```
with open("file.txt", "w") as f:
    f.write("Hello World!!!")
    f.close()
```

Working of Append Mode

Appending text to an existing file.

```
file = open("ict1.txt",'a')
file.write("\n Department Department")
file.close()
```

Output

Reading and Writing Binary Files

Reading and writing binary files, such as images.

Rading files

```
with open(r'C:\Users\Mitesh\OneDrive\Desktop\a.tif', 'rb') as file:
    binary_data = file.read()
```

Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Practical based on File Handling using Python	
Experiment No: 13	Date:	Enrollment No:

Output

```
In [10]: runfile('C:/Users/student/Desktop/
PWP Pratical/untitled6.py', wdir='C:/Users/
student/Desktop/PWP Pratical')
```

Writing binary files

```
with open('c.tif', 'wb') as f:
    f.write(binary_data)
    f.close()
```

Output

```
In [11]: runfile('C:/Users/student/Desktop/
PWP Pratical/untitled6.py', wdir='C:/Users/
student/Desktop/PWP Pratical')
```

Working with CSV Files

```
import csv

# Reading from a CSV file
with open('data.csv', 'r') as file:
    reader = csv.reader(file)
    for row in reader:
        print(row)
```

Output

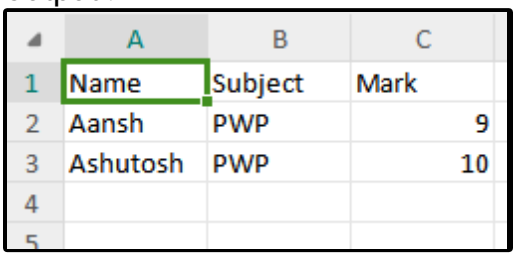
```
In [13]: runfile('C:/Users/student/Desktop/
PWP Pratical/untitled7.py', wdir='C:/Users/
student/Desktop/PWP Pratical')
['Name', 'City', 'Number']
['A', 'M', '1']
['B', 'N', '4']
['C', 'V', '5']
['D', 'B', '7']
['E', 'J', '8']
['F', 'G', '9']
['G', 'F', '7']
['H', 'D', '5']
['I', 'C', '6']
['J', 'X', '7']
['K', 'Z', '3']
['L', 'S', '4']
['M', 'R', '6']
```

```
# Writing to a CSV file
import csv

with open('output.csv', 'w', newline='') as file:
    writer = csv.writer(file)
```

```
writer.writerow(['Name', 'Subject', 'Mark'])
writer.writerow(['Aansh', 'PWP', 9])
writer.writerow(['Ashutosh', 'PWP', 10])
file.close()
```

Output :



	A	B	C
1	Name	Subject	Mark
2	Aansh	PWP	9
3	Ashutosh	PWP	10
4			
5			

Post Lab Exercise:

Write a program that reads a text file example.txt and counts the number of lines, words, and characters in the file. Print these counts.

```
def My_Self(file_name):
    lines, words, characters = 0, 0, 0

    with open(r"C:\Users\Desktop\PWP_Praticals\python files\file.txt", 'r') as file:
        for line in file:
            lines += 1
            words += len(line.split())
            characters += len(line)

    print(f"Lines: {lines}")
    print(f"Words: {words}")
    print(f"Characters: {characters}")

My_Self('file.txt')
```

Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology		
Subject: Programming With Python (01CT1309)	Aim: Practical based on File Handling using Python	
Experiment No: 13	Date:	Enrollment No:

```
In [12]: def My_Self(file_name):
    ...:     lines, words, characters = 0, 0, 0
    ...:
    ...:     with open(r"C:\Users\ambat\OneDrive\Desktop\python files\file.txt", 'r') as file:
    ...:         for line in file:
    ...:             lines += 1
    ...:             words += len(line.split())
    ...:             characters += len(line)
    ...:
    ...:     print(f"Lines: {lines}")
    ...:     print(f"Words: {words}")
    ...:     print(f"Characters: {characters}")
    ...:
    ...: My_Self('file.txt')
Lines: 3
Words: 15
Characters: 68
```

Write a Python program to read a text file line by line and store each line in a list. Print the list after reading the entire file.

```
def read_file(file_name):
    lines = []

    with open(r"C:\Users\Desktop\PWP_Praticals\python files\file.txt", 'r') as file:
        for line in file:
            lines.append(line.strip())

    print("File content as a list:")
    print(lines)
    return lines

read_file('file.txt')
```

Marwadi University Marwadi Chandarana Group NAAC A+	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Practical based on File Handling using Python	
Experiment No: 13	Date:	Enrollment No:

```
In [35]: def read_file(file_name):
    ...:     lines = []
    ...:
    ...:     with open(r"C:\Users\ambat\OneDrive\Desktop\python files\ict1.txt", 'r') as file:
    ...:         for line in file:
    ...:             lines.append(line.strip())
    ...:
    ...:     print("File content as a list:")
    ...:     print(lines)
    ...:     return lines
    ...:
    ...: read_file('ict1.txt')
File content as a list:
['Hi everyone', 'Today is very nice day for me', 'I learnt python', 'which will very useful for me in
future']
Out[35]:
['Hi everyone',
 'Today is very nice day for me',
 'I learnt python',
 'which will very useful for me in future']
```

Write a Python program to read data from a CSV file data.csv and print each row to the console.

```
import csv
def read_csv(file_name):
    with open(r"C:\Users\Desktop\PWP_Praticals\python files\PYTHON.csv", newline='') as csvfile:
        csv_reader = csv.reader(csvfile)

        for row in csv_reader:
            print(row)

read_csv('PYTHON.csv')
```

Marwadi University Marwadi Chandarana Group NAAC A+	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Practical based on File Handling using Python	
Experiment No: 13	Date:	Enrollment No:

```
In [14]: import csv
    ...: def read_csv(file_name):
    ...:     with open(r"C:\Users\ambat\OneDrive\Desktop\python files\PYTHON.csv", newline='') as
csvfile:
    ...:         csv_reader = csv.reader(csvfile)
    ...:
    ...:         for row in csv_reader:
    ...:             print(row)
    ...:
    ...: read_csv('PYTHON.csv')
['Name', 'Subject', 'Mark']
['SHIVANI', 'COA', '9']
['TANVI', 'DMGT', '10']
['payal', 'c++', '8']
['aditya', 'PWP', '8']
['mihir', 'SS', '9']
```

Write a Python program that merges the contents of two text files file1.txt and file2.txt into a third file merged.txt. Ensure that the contents of file1.txt come first.

```
def merge_files(file1, file2, output_file):
   with open(r"C:\Users\Desktop\PWP_Praticals\Merged file.txt", 'w') as merged_file:

      with open(r"C:\Users\Desktop\PWP_Praticals\python files\file.txt", 'r') as f1:
         merged_file.write(f1.read())

      with open(r"C:\Users\Desktop\PWP_Praticals\python files\Hi everyone .txt", 'r') as f2:
         merged_file.write(f2.read())

   print(f"Contents of {file1} and {file2} merged into {output_file}")

merge_files('file.txt', 'Hi everyone .txt', 'Merged file')
```

```
In [15]:
    ...: def merge_files(file1, file2, output_file):
    ...:     with open(r"C:\Users\ambat\OneDrive\Desktop\Merged file.txt", 'w') as merged_file:
    ...:
    ...:         with open(r"C:\Users\ambat\OneDrive\Desktop\python files\file.txt", 'r') as f1:
    ...:             merged_file.write(f1.read())
    ...:
    ...:         with open(r"C:\Users\ambat\OneDrive\Desktop\python files\Hi everyone .txt", 'r') as f2:
    ...:             merged_file.write(f2.read())
    ...:
    ...:     print(f"Contents of {file1} and {file2} merged into {output_file}")
    ...:
    ...: merge_files('file.txt', 'Hi everyone .txt', 'Merged file')
Contents of file.txt and Hi everyone .txt merged into Merged file
```