

Date: 10/5/25

Task - 6.1

Implement various test file operations.

AIM :- To write a python program implement various Text file operations.

ALGORITHM :-

1. Write to a file:

→ Define write file (filename) function:

- Open a file named "log.txt" in write mode.
- Write the following text to the file
- Close the file.

2. Read from a file:

→ Define read file (filename) functions:

- Read the entire content of the file
- Print the content.

3. Execute the Program:

→ Call write file ("write") to write pre-defined text to "log.txt".

→ Call read file ("text") to attempt to read from a file named "text" and print its content.

PROGRAM :-

def write_file (filename):

F = open ("log.txt", "w")

F.write ("Error objects are thrown when runtime errors occurs. The error object can also be used as a base object for user-defined exceptions")

F.close()

def read_file (filename):

With open (filename, "r") as file:

Output

Error objects are thrown when runtime error occurs.

1. Write a file:
→ Define write file (filename) function.
• Open a file named "my.txt" in write mode.
• Write the following text to the file.
• Close the file.

2. Read from a file:
→ Define read file (filename) function.
• Read the entire content of the file.
• Print the content.

3. Execute the program:
→ Call write file ("my.txt") to write the defined text to "my.txt".
→ Call read file ("my.txt") to attempt to read the file content "my.txt" and print the content.

Blockcode:
def write_file(filename):
 f = open(filename, "w")
 f.write("I am writing to a file.")
 f.close()
def read_file(filename):
 f = open(filename, "r")
 content = f.read()
 print(content)
f = open("my.txt", "w")
f.write("I am writing to a file.")
f.close()
read_file("my.txt")

Content = file.read()

Print (Content)

write file ("write")

read file ("text")

19/9/25

Task - 6.2

AIM :- To write a python function that counts the number of lines containing the word 'Error' in a log file.

ALGORITHM :-

1. Initialize Error Counter:
→ Define the function count Error lines (file-name):
2. Open and Read file:
→ Open the file specified by filename in read mode using a with statement.
3. Check Each line for "Error".
→ Loop through Each line in the file.
4. Return Error Count:
→ After reading all the lines, return the value of Error count.
5. Execute the program:
→ Call count Error lines ("log.txt")
→ Print the result with the message.

PROGRAM :-

```
def count_error_lines(filename):  
    error_count = 0  
    with open(filename, 'r') as file:  
        for line in file:  
            if 'ERROR' in line:  
                error_count += 1  
    return error_count  
  
error_lines = count_error_lines("log.txt")  
print("Number of lines with 'ERROR':", error_lines)
```

OUTPUT:-

Number of lines with error = 2

Date :- 10/9/25

Topic 6.3

AIM :- To write a python function that generates an Employee report with names and departments and save it 'Employee-report.txt'.

ALGORITHM :-

1. Create Employee Data.
→ Define the function write_employee_report (filename);
2. Open file for writing :
→ open the file specified by filename in write mode using a with statement.
3. Write an employee data to file :
→ Loop through each Employee in the Employee list.
4. Execute the program.

PROGRAM :-

```
def write_employee_report(filename):  
    employees = [  
        {"name": "Alice", "department": "HR"},  
        {"name": "Bob", "department": "Engineering"},  
        {"name": "Charlie", "department": "Finance"}  
    ]  
    with open(filename, "w") as file:  
        for Employee in employees:  
            line = f"Name : {Employee['name']},  
                    Department : {Employee['department']}\n"
```

OUTPUT :-

Name : Alice , Department : HR.

Name : Bob , Department : Engineering

Name : Charlie , Department : Finance

file.write(line)

write - employee - report ("employee - report - txt")

VEL TECH - CSE	
EX NO.	6
PERFORMANCE (5)	5
RESULT AND ANALYSIS (3)	5
VIVA VOCE (3)	5
RECORD (4)	5
TOTAL (15)	15
DATE	

RESULT :- Thus, the Python program implement various text file operations was successfully executed and the output was verified.