10/9/25 TASK: 6.1

Implement various text file operation.

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.
   - Error objects are known when runtime error occurs.
   - close the file.

2. Read from a file:
   - Define read file (filename) function.
   - open the files specified by filename in read using a with Statement.
   - read the entire content of the file
   - print the content.

3. Execute the Program.
   - Call write file ("write") to write the predefined text to "log txt".
   - Call read file (text) to attempt to read from a file name "text" and print its content.

program :

```
def whilefile (filename):
    f = open ("log txt", "w")
    f.write ("Error objects are thrown when
    run time error occur. The error object
    can also be used as base object
    for user-defined exceptions")
    f.close ()
def readfile (filename):
    withe open (filename, "r") as file:
    content = file read()
    print (content)
write file ("write")
read file ("text")
```

Output:

Error objects are thrown when runtime errors occur. The Error object can also be used as a base object for user-defined exceptions.

10/9/25 Task : 12

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
   - Initialize error-count to 0.
2. open and read file :
   - open the file spacified by filename in read mode using a with statement.
3. Check Each line for "ERROR".
   - loop through line the file
   - If line contains the word "ERROR" increment by 1.
4. Return Error count :
   - After reading all the lines return the valve of error-count.
5. Execute the program.
   - Call count-error-lines ("log.txt") to count the number with the word "ERROR" in the file "log.txt".
   - print the result with the message: "Number of lines the "ERROR" : {error-lines}"

program :-

```
def count-error-lines (file name):
  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-line("log.txt")
    print(f"Number of lines with ERROR: {error
        lines}").
```

log.txt

"Error objects are thrown when runtime Error occur.

The error object can also be used as base object for user defined executed.

Output:

RESTART: C:/Users/91979/Desktop/5125/6a Py

Number of lines with 'ERROR' is 2

10/9/25

TASK : 6.3

Aim : To write 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
   - Create a list employee containing dictionaries each with 'name" and development keys for Individual employee.
2. open file for writing
   - open the file specified by filename in write mode using a with statement
3. write Employee Data to file
   - Loop through each employee in the employee list
   - for each employee format a string as Name
   - write the formatted string to the file followed by a newline character (\n)
4. Execute the program
   - Call write-employee-report ("employee_report txt") : to write the employee data to the file "employee_report-txt".

Program :

```
def write-employee-report (filename):
employees = (
{"name": "Alica", "department": "HR"},
{"name": "Charlie", "department": "Engineering"},
{"name": "charlie", "department": "finance"},
}
while open (filename, "w") as file
for employee in employees:
line f"Name: {employee [iname]} Departm
-ment
{employee ("department]} \n"
file write (line)
# Example usage
write employee_report("employee-report.txt")
```

EX NO.	6
PERFORMANCE (5)	5
RESULT AND ANALYSIS (3)	3
VIVA VOCE (3)	3
RECORD (4)	4
TOTAL (15)	
SIGN WITH DATE	15

Result: Thus the python program implemented various text file operation was successfully executed and the output was verified.

Output :

Name : Alice, Department : HR
Name : Bob, Department : Engineering
Name : charlie, Department : Finance.