Task 8. Implement various text file Operation.

Aim: To write a Python Program implement various text file operations.

Problem 8.1:
You need to write the sequence "Error objects are thrown when runtime errors occur. The error object can also be used as a base object for user-defined exceptions" into a text file named log.txt. Implement a function that Performs this task.

Algorithm:

1. write to a File:
   → Define write file function:
   → open a file named "log.txt" in write mode.
   → close the file.
2. Read from a file:
   → Define read file function.
   → Read the entire content of file.
   → Print the content.
3. Execute the Program:
   → call write file to write Predefined text to "log.txt".
   → Call read file to attempt to read from a file named "text" and Print its content.

Program:

```
def writefile(filename):
    f=open("log.txt","w")
    f.write("Error objects are thrown when runtime error occur")
    f.close()
def-readfile(filename):
    with open(filename,"r") as file:
        content = file.read()
        Print(content)
write file("write")
read file("text")
```

Output:

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

Q.2 You have a text file log.txt containing logs of a system write a function that counts the number of lines containing the word "ERROR".

Algorithm:

1. Initialize Error counter:
→ Define the function count_error_lines.
→ Initialize error_count to 0.

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 file.
→ If the line contains word "ERROR", increment error_count by 1.

4. Return ERROR count.
→ After reading all the lines, return the value of err_count

Program:

```
def count_error_lines(filename):
error_count = 0
with open(filename, "r") as file:
 for line in file:
 if ("ERROR" in line):
error_coont += 1
return error_count
error_lines = coont_error_lines("log.txt")
Print(f"number of lines with 'ERROR': {error_lines}")
```

log.txt.

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

output:

Number of lines with 'ERROR' is 2

8.3 You need to write a report containing the details of the employee in list. write a Python function that writes this report to a file named 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 Employee Data to File:
→ loop through each employee in the employees list
4. Execute the Program:
→ call write_employee_report ("employee_report.txt") to write the employee data to file "employee_report.txt".

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']}.
```

Output:

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

VEL TECH : CSE	
EX NO.	8
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	
TOTAL (20)	
SIGN WITH DATE	15

Result:

Thus, the Python Program implement various text file operations was successfully executed and output was verified.