Task-6:- Implement various text file operation

Aim:- TO write a python program Implement various text file operations.

Problem 6.1:- You need to write the sentence "Error objects are thrown when runtime errors occur. The Error 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 writefile(filename) function:
     - Open a file named "log.txt" in write mode
     - write the following text to the file.
     - "Error objects are thrown when runtime errors occur. The Error object can also be used as a base object for user-defined exceptions"
     - close the file.

2. **Read from a File:-**
   * Define readfile(filename) function:
     - Open the file specified by filename in read mode using a with statement.
     - Read the entire content of the file.
     - print the content.

3. **Execute the program:-**
   - call writefile("write") to write the predefined text to "log.txt".
   - call readfile("text") to attempt to read from a file named "text" and print its content.

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

Program 6.1 :-

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

Problem 6.2 :- 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(filename):
     - 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 the file:
     - If the line contains the word "ERROR", increment error_count by 1.
4. Return Error count :

output :-

Number of lines with 'ERROR' is 2

- After reading all the lines, return the value of error_count.

5. Execute the program:
   - call count_error_lines("log.txt") to count the number of lines with the word "ERROR" in the file "log.txt".
   - print the result with the message: "Number of line with 'ERROR': {error_lines}".

Program 6.2 :-

```
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(f"Number of lines with 'ERROR': {error_lines}")
```

problem 6.3 :- You need to write a report containing the details (Name, departments) 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):
     - create a list employees containing dictionaries, each with "name" and "department" keys for individual employees.

2. Open File for Writing:

Output :-

Name : Alice, Department : HR
Name : Bob, Department : Engineering
Name : Charlie, Department : Finance

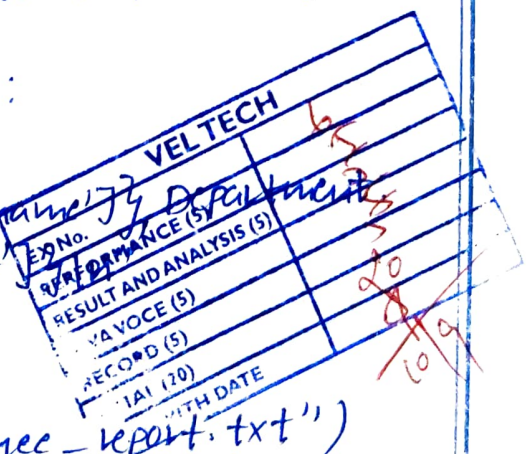
- open the file specified by filename in write mode using a with statement.

3. Write Employee Data to File:
   - Loop through each employee in the employees list:
     - For each employee, format a string as "Name: {employee['name']}, Department: {employee['department']}".
     - 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 employment data to the file "employee_report.txt".

Program 6.3 :-

```
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"
            file.write(line)
# Example usage:
write_employee_report("employee_report.txt")
```



Result :- Thus, the python program implementations text file operations was successfully executed and the output was verified.