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.

1. write to a file:-

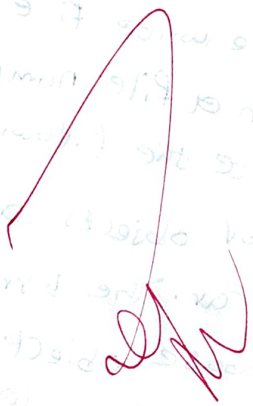
* Define write file (file name) function:
- open a file named "log.txt" in write mode
- write the following text to the file.
- Error objects are thrown when runtime errors occur. The Error object can also be used as a base object for user-defined exceptions"
- Close the file.

2. Read from a file:-

* Define read file (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 write file ("write") to write the predefined text to "log.txt".
- Call read file ("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 write file (filenames);
  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)
      write file ("write")
      read file ("text")
```

Problem 6-2 :- You have a text file log.txt containing logs of a system write a function that connects the number of lines containing the word "ERROR".

Algorithm :-

1 - Initialize Error_Count :-

- 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 the file :
- If the line contains the world "ERROR", increment error_count by 1.

4. Return Error Count :

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

Output:-

Number of lines with 'ERROR' is 0

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 lines with 'ERROR':"

Program 6.2:

```
def Count_error_lines(filename):
    error_count = 0
with open(file name, "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)
```

Problem 6.3: you need to write a report containing the details (name, department) 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 (file name):
   - Create a list employees containing dictionaries each with "name" and department keys for individual employees.
2. Open file for writing:
   - Open the file specified by filename in write mode using a with statement

Output:

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

3. Write Employee Data to File

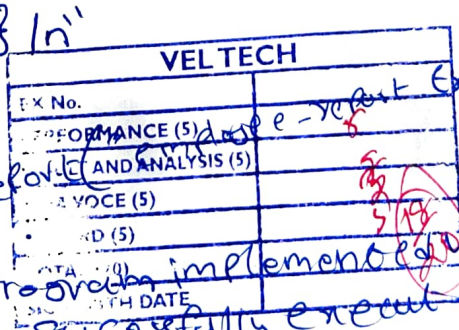
- Loop through each employee in the employees list.
- For each employee, format a string as "name: {employee['name']} Department: {employee['department']}".
- Write the formatted string to the file, followed by a newline character (\n).

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']}, Departments
{employee['department']}\n"
        file.write(ine)
# Example usage:
write_employee_report("employee_report.txt")
```



Result: Thus, the python program for implementing file operations was successfully executed and the output was verified.