*Program 6.1:-

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

Date:- 1/9/25

TASK-6:- Implement various text file operation

Aim:- To write a python programming ~~z~~ to implement various text file operations.

Problem:- 6.1:-

You need to write the sentence "Error objects are thrown when runtime.errors occur. The Error Object Can also be used as a base object for user-defined exceptions.

Algorithm:-

```
1. write to a File:-
  o Define writefile (filename) function:
    • open a file name "log.text" 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 form file:-
  • Define readfile (filenam) function:
    • open the file specified by filename in read mode
      using a with statement.
    • Read the entire Content of file.
    • print the Content.
3. Execute the program:-
  • Call writefile ("write") to write the predefined text to
    "log.tet".
  • Call readfile ("text") to attempt to read from a file
    named "text" and print its Content.
```

* Program 6.3:-

```
def Count_error_lines (filename):
    error _Count =0
    with open (filename,"r")as file:
    for line in life;
    If" ERROR" in line;
        error _Count +=1
     return error_count
error_lines= Count_error_lines ("log.tet")
print (f"Number of lines with'ERROR' :{error_lines}").
```

log.txt

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

output:-

```
Number of lines with Error" is 2
```

Program 6.2

You have a text file log.tet Containing logs of a system. write a function that Counts the number of lines Containing word "ERROR".

Algorithm:-

```
1. Initialize Error Counter :
    o Define the function Count_error_lines (filename):
        ■ Initialize error_Count to 0.
2. Open and Read File:
    o open the file specified by filename in read
      mode using a with statement.
3. Check Each line for " ERROR":
    o loop through each line in the file
        ■ If the Contains the word "ERROR", increment
          error_Count by 1.
4. Return Error Count:
    o After reading all the lines, return the value of
      error_count.
5. Execute the program:
    o call Count_eerror_lines ("log.txt") to Count the
      number of lines with the word "ERROR" in the
      file "log.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']} ,Department :
            {employee ['department']} \n".
       file .write (line).

#Example usage:-
write_employee_report ("employee_report.txt").
```

Output:-

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

VEL TECH	
EX No.	6
PERFORMANCE (5)	[illegible]
RESULT AND ANALYSIS (5)	[illegible]
VIVA VOCE (5)	[illegible]
RECORD (5)	[illegible]
TOTAL (20)	[illegible]
SIGN WITH DATE	

VEL TECH	
EX No.	
PERFORMANCE (5)	
RESULT AND ANA	
VIVA VOCE (5)	
RECORD (4)	
TOTAL (15)	
SIGN WITH DATE	

Program 6.3:-

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):
      ■ Create a list employees containing dictionaries
        each with "name" and "department" key for
        individual employees.
2. Open file for working:
    • 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:
      {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 employee data to the file "employee_
      report.txt".
```

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