**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 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 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.

**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:

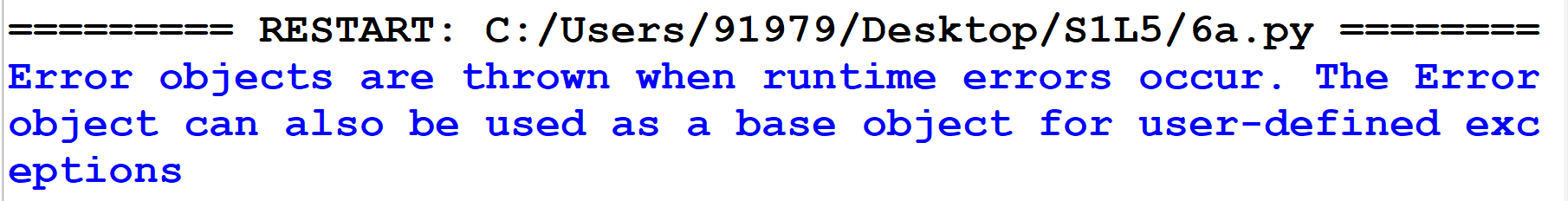
content = file.read()

print(content)

writefile("write")

readfile("text")

Output:



**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:**
   * 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 lines with 'ERROR': {error\_lines}".

**Program 6.3:**

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")

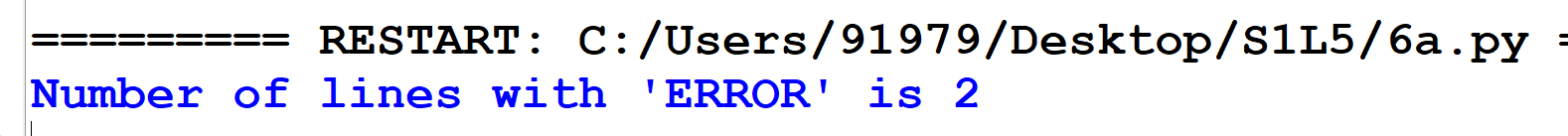
print(f"Number of lines with 'ERROR': {error\_lines}")

**log.txt**

“**Error** objects are thrown when runtime **Error** occur.

The **Error** object can also be used as a base object for user-defined exceptions.”

**Output:**



**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:
   * 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 employee 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:

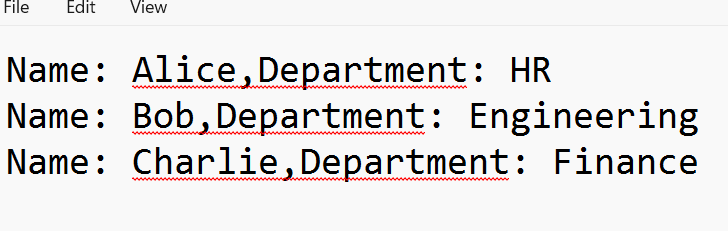
line = f"Name: {employee['name']}, Department: {employee['department']}\n"

file.write(line)

# Example usage:

write\_employee\_report("employee\_report.txt")

**output:**



**Result**:

Thus, the python program Implement various text file operations was successfully executed and the output was verified.