

## Task-6: Implement various text file operation

Aim: To write a python programming Implement various text file operation

6.1 :- You need to write the sentence "Error object 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 lag.txt. Implement a function that perform this task.

Algorithm:-

1. Write to a file:-

\* Define write file function

• open a file named "lag.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

2. Read from a file

\* Define read file function

• open the file specified filename in read mode

• Read the entire content of file

• Print the content

3. Execute the program



## Program

```
def write_file (file_name):  
    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")  
    f.close()
```

```
def read_file (file_name):  
    with open (file_name, "r") as file:  
        content = file.read()  
        print (content)  
write_file ("write")  
read_file ("txt")
```

Problem 6.2: You have a txt file log.txt containing logs of a system. Write a function that counts the no. of lines the word "error".

### Algorithm:

1. Initialize error counter:
  - Define the func count\_error\_lines (file\_name):
  - Initialize error\_count to 0.
2. Open and read file:
  - Open the file by filename in read mode.
3. Check each line for "Error":
  - Loop through each line in the file
  - If the line contains the word "error", increment count by 1

## Output

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

At runtime, when an error occurs, the program throws an error object. The error object contains information about the error, such as the type of error, the location of the error, and the message of the error.

The error object is a subclass of the `Exception` class. It is created by the runtime system and passed to the user-defined exception handler.

The user-defined exception handler can then use the error object to display an error message or to take corrective action.

For example, if a program throws a `FileNotFoundException`, the user-defined exception handler can catch the exception and display a message like "File not found".

The error object is also used by the runtime system to generate stack traces. A stack trace is a list of the method calls that led to the error. It is useful for debugging the program.

The error object is also used by the runtime system to generate error reports. An error report is a document that contains information about the error, such as the type of error, the location of the error, and the message of the error. It is useful for reporting errors to the user or to a support team.

4. Execute the programme

- call `count - error - lines ("log.txt")` to count the no. of lines with the word "Error" in the file "log.txt".

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 ("no. of lines with 'Error': {error - lines}")
```

Problem 6.3:- You need to write a report containing the details (Name, dept) 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 func `write - Employee - report`
- create a list of Employees containing dictionaries
- Each with "name" and "dept" keys

2. open file for writing.

- open file by file name in write mode using `with` statement

Output:-

output:-

Name: Alice, Dept: HR

Name: Bob, Dept: Engineer

Name: Charlie, Dept: finance



3. write emp Data to file:

- Loop through each Employee in the employees list
- For each Employee, format a string as  
"Name: {Employee['name']}, {Employee['dept']}."

4. Execute the program

- call write - Employee - report ("Employee - report .txt") to write the Employment data.

Program 6.3:-

```
def write - Employee - report (filename):
```

```
    Employees = [
```

```
        {"name": Alice, "dept": "HR"},
```

```
        {"name": Bob, "dept": "Engineer"},
```

```
        {"name": Charlie, "dept": "Finance"}]
```

```
with open (filename, "w") as file:
```

```
    for employee in Employees:
```

```
        line = f"Name: {employee['name']}, Dept:
```

```
        {employee['dept']} \n"
```

```
        file.write (line)
```

```
# Example
```

```
write - Employee - report ("Employee - report .txt")
```

Result: Thus, the python program implement

various text file operations was successfully

executed and the output

VELTECH	
EX No.	was verified.
PERFORMANCE (5)	
RESULT AND ANALYSIS (5)	
VIVA VOCE (5)	
RECORD (5)	
TOTAL (20)	
WITH DATE	