python-programming-lab-9

March 13, 2025

Python Programming - 2301CS404

Gohel Neel

Enrollnment No. : 23010101089

Roll No. 340

Date: 27-01-2025

Lab - 9

1 File I/O

1.0.1 01) WAP to read and display the contents of a text file. (also try to open the file in some other directory)

- in the form of a string

- line by line

- in the form of a list

[1]:
```
file_path = 'file1.txt'

with open(file_path, 'r') as file:
    content = file.read()
    print(content)

with open(file_path, 'r') as file:
    for line in file:
        print(line, end='')

with open(file_path, 'r') as file:
    lines = file.readlines()
    print(lines)
```

```
hello from the file
hello from the file['hello from the file']
```

1.0.2 02) WAP to create file named “new.txt” only if it doesn’t exist.

[2]:
```
fp = open("new.txt","w")
s1 = "Hello form file write method"
fp.write(s1)
fp.close()
```

1.0.3 03) WAP to read first 5 lines from the text file.

[3]:
```
with open('file1.txt', 'r') as file:
    for i in range(5):
        line = file.readline()
        print(line, end='')
```

```
hello from 1st line
hello from 2nd line
hello from 3rd line
hello from 4th line
hello from 5th line
```

1.0.4 04) WAP to find the longest word(s) in a file

[4]:
```
with open('file1.txt', 'r') as file:
    words = file.read().split()
    max_length = max(len(word) for word in words)
    longest_words = [word for word in words if len(word) == max_length]

print(longest_words)
```

```
['bsjbebrjbgjebtrjbwrjbjbtbj']
```

1.0.5 05) WAP to count the no. of lines, words and characters in a given text file.

[5]:
```
lines_count = 0
words_count = 0
chars_count = 0

with open('file1.txt', 'r') as file:
    for line in file:
        lines_count += 1
        words_count += len(line.split())
        chars_count += len(line)

print(f"Lines: {lines_count}")
print(f"Words: {words_count}")
print(f"Characters: {chars_count}")
```

```
Lines: 7
Words: 28
Characters: 160
```

1.0.6 06) WAP to copy the content of a file to the another file.

[6]:
```
with open('file1.txt', 'r') as source_file:
    content = source_file.read()

with open('newCopyFile.txt', 'w') as destination_file:
    destination_file.write(content)
```

1.0.7 07) WAP to find the size of the text file.

[10]:
```
with open('file1.txt', 'r') as file:
    data = file.read()
    size = len(data)

print(size)
```

```
160
```

1.0.8 08) WAP to create an UDF named frequency to count occurances of the specific word in a given text file.

[14]:
```
def frequency(file_path, word_to_find):
    count = 0
    with open(file_path, 'r') as file:
        for line in file:
            count += line.lower().split().count(word_to_find.lower())
    return count

file_path = 'file1.txt'
word_to_find = 'hello'

word_count = frequency(file_path, word_to_find)
print(f"The word '{word_to_find}' appears {word_count} times in the file.")
```

```
The word 'hello' appears 5 times in the file.
```

1.0.9 09) WAP to get the score of five subjects from the user, store them in a file. Fetch those marks and find the highest score.

[15]:
```
file_path = 'marks.txt'

scores = []
for i in range(1, 6):
```

```
    score = int(input(f"Enter marks subject {i}: "))
    scores.append(score)

with open(file_path, 'w') as file:
    for score in scores:
        file.write(f"{score}\n")

with open(file_path, 'r') as file:
    scores_from_file = [int(line.strip()) for line in file]

highest_score = max(scores_from_file)
print(f"The highest score is: {highest_score}")
```

```
Enter marks subject 1:  12
Enter marks subject 2:  6526
Enter marks subject 3:  261
Enter marks subject 4:  6261
Enter marks subject 5:  66161

The highest score is: 66161
```

1.0.10 10) WAP to write first 100 prime numbers to a file named primenumbers.txt

(Note: each number should be in new line)

[18]:
```
def is_prime(num):
    if num <= 1:
        return False
    for i in range(2, int(num ** 0.5) + 1):
        if num % i == 0:
            return False
    return True

file_path = 'primenumbers.txt'
prime_numbers = []
num = 2

while len(prime_numbers) < 100:
    if is_prime(num):
        prime_numbers.append(num)
    num += 1

with open(file_path, 'w') as file:
    for prime in prime_numbers:
        file.write(f"{prime}\n")
```

1.0.11 11) WAP to merge two files and write it in a new file.

[19]:
```
file1_path = 'new.txt'
file2_path = 'newCopyFile.txt'
merged_file_path = 'merged_file.txt'

with open(file1_path, 'r') as file1, open(file2_path, 'r') as file2:
    content1 = file1.read()
    content2 = file2.read()

with open(merged_file_path, 'w') as merged_file:
    merged_file.write(content1)
    merged_file.write(content2)
```

1.0.12 12) WAP to replace word1 by word2 of a text file. Write the updated data to new file.

[ ]:
```
file_path = 'new.txt'
new_file_path = 'updated_file.txt'
word1 = ''
word2 = 'newword'

with open(file_path, 'r') as file:
    content = file.read()

updated_content = content.replace(word1, word2)

with open(new_file_path, 'w') as new_file:
    new_file.write(updated_content)
```

1.0.13 13) Demonstrate tell() and seek() for all the cases(seek from beginning-end-current position) taking a suitable example of your choice.

[ ]:
```

```