

## ASSIGNMENT - 4.3

2303A51355

Batch-10

### Task-1

Prompt: Give me a program to zero-short prompting to check a leap year, and give instructions without providing examples

code :

```
def is_leap_year(year):    if (year % 4 == 0 and year % 100
!= 0) or (year % 400 == 0):
```

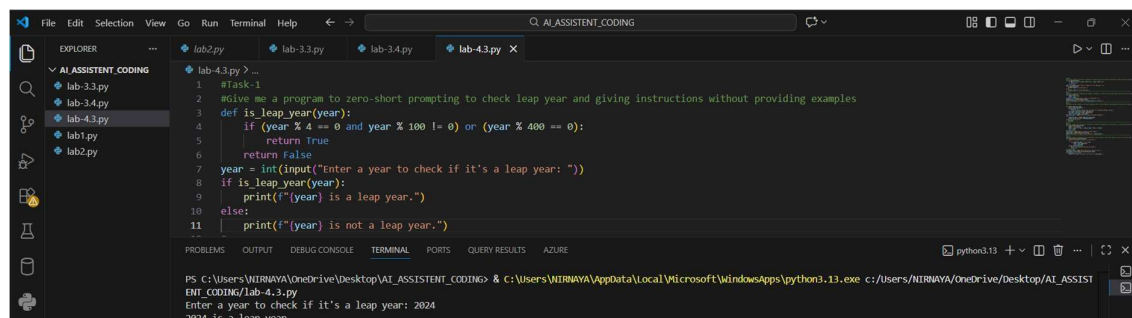
```
        return True    return False    year = int(input("Enter a
year to check if it's a leap year: ")) if is_leap_year(year):
```

```
    print(f'{year} is a leap year.')
```

else:

```
    print(f'{year} is not a leap year.') Output
```

:



```
1 #Task-1
2 #Give me a program to zero-short prompting to check leap year and giving instructions without providing examples
3 def is_leap_year(year):
4     if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
5         return True
6     return False
7 year = int(input("Enter a year to check if it's a leap year: "))
8 if is_leap_year(year):
9     print(f'{year} is a leap year.')
10 else:
11     print(f'{year} is not a leap year.')
```

PS C:\Users\NIRNAYA\OneDrive\Desktop\AI\_ASSISTENT\_CODING> & C:\Users\NIRNAYA\AppData\Local\Microsoft\WindowsApps\python3.13.exe c:/Users/NIRNAYA/OneDrive/Desktop/AI\_ASSISTENT\_CODING/lab-4.3.py

Enter a year to check if it's a leap year: 2024

2024 is a leap year.

Code Analysis:

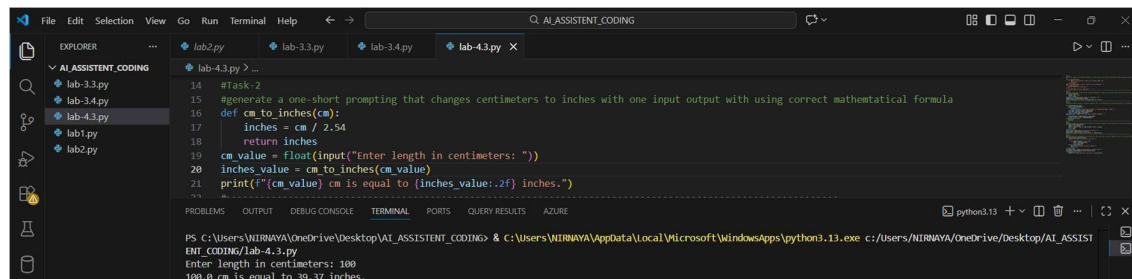
- ☐ This program determines whether a given year is a leap year using a function.
- ☐ The function applies standard leap year rules and returns True or False.
- ☐ The user inputs a year, which is checked by the function.
- ☐ The result is printed as either a leap year or not.

### Task-2

Prompt: generate a one-short prompt that changes cen metres to inches with one input and output using the correct mathema cal formula

Code :

```
def cm_to_inches(cm):  
    inches = cm / 2.54  
  
    return inches  
  
cm_value = float(input("Enter length in cen meters: "))  
  
inches_value = cm_to_inches(cm_value)  
  
print(f'{cm_value} cm is equal to {inches_value:.2f}  
inches.") Output :
```



Code Analysis:

- ☐ This program converts a length from centimetres to inches using the correct mathematical formula.
- ☐ A function performs the conversion by dividing the value by 2.54.
- ☐ The user enters a value in centimetres, which is passed to the function.
- ☐ The converted result is displayed in inches.

Task-3

Prompt: generate a python program is few-short promp ng that is name forma ng like accep ng fullname as firstname,lastname.

Code :

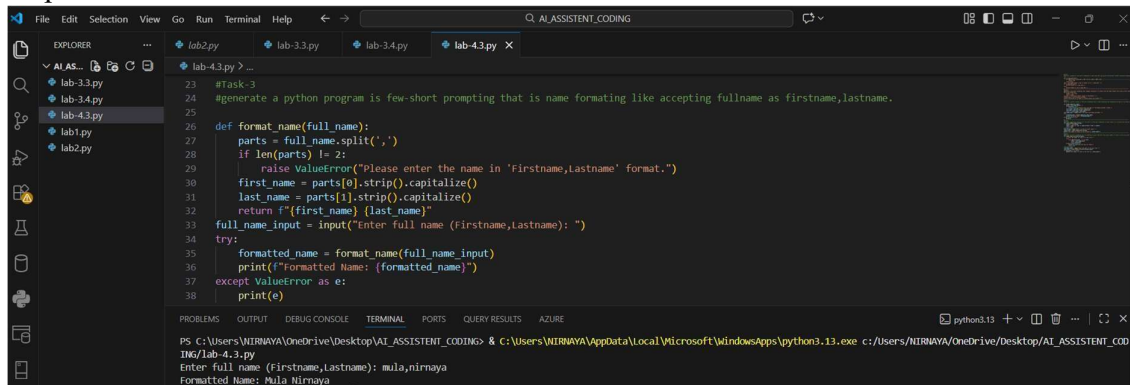
```
def format_name(full_name):  
    parts = full_name.split(',')  
  
    if len(parts) != 2:  
        raise ValueError("Please enter the name in 'Firstname,Lastname' format.")  
  
    first_name = parts[0].strip().capitalize()
```

```

    last_name = parts[1].strip().capitalize()    return f'{first_name}
{last_name}'    full_name_input  =  input("Enter  full  name
(Firstname,Lastname): ")
try:
    formated_name = format_name(full_name_input)
print(f'Formated Name: {formated_name}')
except ValueError as e:
    print(e)

```

Output:



The screenshot shows a VS Code editor with a file named 'lab-4.3.py' open. The code in the file is as follows:

```

23 #Task-3
24 #generate a python program is few-short prompting that is name formatting like accepting fullname as firstname,lastname.
25
26 def format_name(full_name):
27     parts = full_name.split(',')
28     if len(parts) != 2:
29         raise ValueError("Please enter the name in 'Firstname,Lastname' format.")
30     first_name = parts[0].strip().capitalize()
31     last_name = parts[1].strip().capitalize()
32     return f'{first_name} {last_name}'
33 full_name_input = input("Enter full name (Firstname,Lastname): ")
34 try:
35     formatted_name = format_name(full_name_input)
36     print(f"Formatted Name: {formatted_name}")
37 except ValueError as e:
38     print(e)

```

The terminal output at the bottom shows the execution of the script:

```

PS C:\Users\NIRNAYA\OneDrive\Desktop\AI_ASSISTENT_CODING> & C:\Users\NIRNAYA\AppData\Local\Microsoft\WindowsApps\python3.13.exe c:/Users/NIRNAYA/OneDrive/Desktop/AI_ASSISTENT_COD
IMG/lab-4.3.py
Enter full name (Firstname,Lastname): mula,nirnaya
Formatted Name: Mula Nirnaya

```

Code Analysis :

- ☐ This program formats a full name entered as first name and last name.
- ☐ The input is split and validated to ensure the correct format.
- ☐ Each part of the name is cleaned and capitalised. ☐ The formated full name is then displayed.

Task-4

Prompt: [generate a comparative analysis for zero-shot vs few-shot prompting to count vowels in a string using a function:](#)

Code :

```

def count_vowels(input_string):
    vowels = 'aeiouAEIOU'    count = sum(1 for char in
input_string if char in vowels)    return count
# Example usage

```

```
test_string = input("Enter a string to count vowels: ")

vowel_count = count_vowels(test_string) print(f"The
number of vowels in the string is: {vowel_count}") Output:
```

```

40 #Task-4
41 #generate comparative analysis for zero-shot vs few-shot prompting to count vowels in a string using function
42 def count_vowels(input_string):
43     vowels = 'aeiouAEIOU'
44     count = sum(1 for char in input_string if char in vowels)
45     return count
46 # Example usage
47 test_string = input("Enter a string to count vowels: ")
48 vowel_count = count_vowels(test_string)
49 print(f"The number of vowels in the string is: {vowel_count}")

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS QUERY RESULTS AZURE
python3.13
PS C:\Users\NIRNAYA\OneDrive\Desktop\AI_ASSISTENT_CODING> & c:\Users\NIRNAYA\AppData\Local\Microsoft\WindowsApps\python3.13.exe c:\Users\NIRNAYA\OneDrive\Desktop\AI_ASSISTENT_CODING\lab-4.3.py
Enter a string to count vowels: Nirnaya
The number of vowels in the string is: 3

```

### Code Analysis :

- ☐ The function counts vowels in a given string using a direct logic approach.
- ☐ Zero-shot prompting applies the logic without examples.
- ☐ Few-shot prompting helps by showing patterns before execution.
- ☐ The function returns the total number of vowels in the input string.

### Task-5

Prompt: [generate a few short prompts for file handling to give a read text file, count the number of lines in the file, and line count by function](#)

```
def count_lines_in_file(file_path):

    try:

        with open(file_path, 'r') as file:

            lines = file.readlines()

    return len(lines)

    except FileNotFoundError:

        print("The specified file was not found.")

    return None

# Example usage

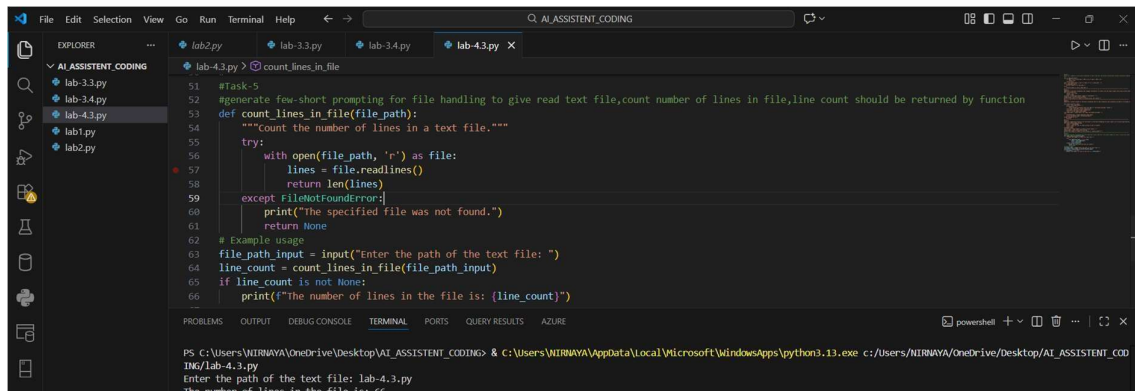
file_path_input = input("Enter the path of the text file:

") line_count = count_lines_in_file(file_path_input)

if line_count is not None:
```

```
print(f"The number of lines in the file is: {line_count}")
```

Output :



The screenshot shows a Visual Studio Code editor window with a file explorer on the left and a code editor in the center. The file explorer shows a folder named 'AI ASSISTENT CODING' containing several Python files. The code editor displays a Python script named 'count\_lines\_in\_file.py'. The script defines a function 'count\_lines\_in\_file' that takes a file path as input and returns the number of lines in the file. It uses a try-except block to handle file not found errors. The script also includes a main block that prompts the user for a file path and prints the line count. The terminal at the bottom shows the command to run the script and the resulting output: 'The number of lines in the file is: 66'.

```
51 #task-5
52 #generate few short prompting for file handling to give read text file,count number of lines in file,line count should be returned by function
53 def count_lines_in_file(file_path):
54     """count the number of lines in a text file."""
55     try:
56         with open(file_path, 'r') as file:
57             lines = file.readlines()
58             return len(lines)
59     except FileNotFoundError:
60         print("The specified file was not found.")
61         return None
62 # Example usage
63 file_path_input = input("Enter the path of the text file: ")
64 line_count = count_lines_in_file(file_path_input)
65 if line_count is not None:
66     print(f"The number of lines in the file is: {line_count}")
```

PS C:\Users\NIRNAYA\OneDrive\Desktop\AI\_ASSISTENT\_CODING> & C:\Users\NIRNAYA\AppData\Local\Microsoft\WindowsApps\python3.13.exe c:/Users/NIRNAYA/OneDrive/Desktop/AI\_ASSISTENT\_CODING/lab-4.3.py  
Enter the path of the text file: lab-4.3.py  
The number of lines in the file is: 66

Code Analysis :

- ☐ This program reads a text file and counts the number of lines.
- ☐ A function opens the file safely and calculates the line count.
- ☐ Error handling is used if the file does not exist.
- ☐ The final line count is returned and displayed.