AI ASSISTED CODING

Prompt Engineering-Improving Prompts

H.no:2403A510G7 **Name:N.Prudhvi**

Batch:06

TASK-1:

PROMPT:

write a function that checks whether a given year is a leap year.

CODE:

```
def is_leap_year(year: int) -> bool:
    """Check if a given year is a leap year."""
    if (year % 400 == 0) or (year % 4 == 0 and year % 100 != 0):
        return True
    return False
```

TASK-2:

CODE:

```
def cm_to_inches(cm: float) -> float:
    """Convert centimeters to inches."""
    return cm / 2.54   # since 1 inch = 2.54 cm


# Example usage:
print(cm_to_inches(10))   # 3.937007874015748
print(cm_to_inches(100))  # 39.37007874015748
```

OUTPUT:

```
PS C:\Users\lenovo\
3.937007874015748
39.37007874015748
```

TASK-3:

```python
def format_name(first: str, last: str) -> str:
    return f"{last}, {first}"
# Example usage:
print(format_name("John", "Doe"))       # Doe, John
print(format_name("Alice", "Smith"))    # Smith, Alice
```

```
Doe, John
Smith, Alice
```

EXAMPLE:2

```python
def format_full_name(full_name: str) -> str:
    first, last = full_name.split()
    return f"{last}, {first}"
# Example usage:
print(format_full_name("Emma Watson"))     # Watson, Emma
print(format_full_name("Michael Jordan"))  # Jordan, Michael
```

```
Watson, Emma
Jordan, Michael
```

EXAMPLE:3

```python
def format_name_extended(full_name: str) -> str:
    parts = full_name.split()
    if len(parts) == 2:   # First + Last
        first, last = parts
        return f"{last}, {first}"
    elif len(parts) == 3: # First + Middle + Last
        first, middle, last = parts
        return f"{last}, {first} {middle}"
    else:
        return full_name  # Return as-is if unexpected format

# Example usage:
print(format_name_extended("John Paul Jones"))   # Jones, John Paul
print(format_name_extended("Mary Jane"))         # Jane, Mary
```

```
Jones, John Paul
Jane, Mary
```

TASK-4:

PROMPTS:

ZERO SHOT:

write a function that checks whether a given year is a leap year.

FEW SHOT:

: Provide 2–3 examples to generate a function that formats full names as “Last, First”.

Outputs:

```python
def is_leap_year(year: int) -> bool:
    """Check if a given year is a leap year."""
    if (year % 400 == 0) or (year % 4 == 0 and year % 100 != 0):
        return True
    return False
```

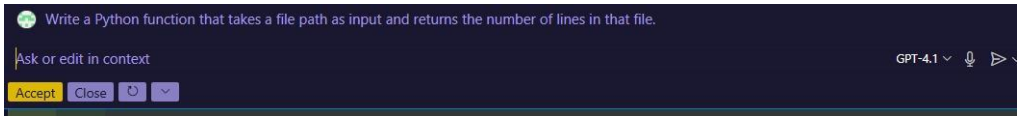
```python
def format_name(first: str, last: str) -> str:
    return f"{last}, {first}"
# Example usage:
print(format_name("John", "Doe"))      # Doe, John
print(format_name("Alice", "Smith"))   # Smith, Alice
```

```python
def format_name_extended(full_name: str) -> str:
    parts = full_name.split()
    if len(parts) == 2:    # First + Last
        first, last = parts
        return f"{last}, {first}"
    elif len(parts) == 3: # First + Middle + Last
        first, middle, last = parts
        return f"{last}, {first} {middle}"
    else:
        return full_name  # Return as-is if unexpected format
# Example usage:
print(format_name_extended("John Paul Jones"))    # Jones, John Paul
print(format_name_extended("Mary Jane"))          # Jane, Mary
```

```python
def format_full_name(full_name: str) -> str:
    first, last = full_name.split()
    return f"{last}, {first}"
# Example usage:
print(format_full_name("Emma Watson"))     # Watson, Emma
print(format_full_name("Michael Jordan"))  # Jordan, Michael
```

TASK-5:

PROMPT:



CODE:

```python
def count_lines(file_path):
    try:
        with open(file_path, "r") as f:
            return len(f.readlines())
    except Exception as e:
        print(f"Error: {e}")
        return None


filename = input("Enter filename: ")
if not filename.endswith(".txt"):
    filename += ".txt"
lines = count_lines(filename)
if lines is not None:
    print(f"Total lines in '{filename}': {lines}")
```

OUTPUT:

```
C:\Users\venub\OneDrive\Desktop\AIAC_Lab\Lab_4>python 4_1.py
Enter filename: corrupted_scroll123
Total lines in 'corrupted_scroll123.txt': 8

C:\Users\venub\OneDrive\Desktop\AIAC_Lab\Lab_4>python 4_1.py
Enter filename: summary
Total lines in 'summary.txt': 7
```